

Cross-domain and Cross-dimension Learning for Image-to-Graph Transformers

Alexander H. Berger¹, Laurin Lux^{1,3}, Suprosanna Shit^{1,5}, Ivan Ezhov¹,
 Georgios Kaissis^{1,3,4}, Martin J. Menten^{1,2,3}, Daniel Rueckert^{1,2,3}, Johannes C. Paetzold^{2,6}
¹ School of Computation, Information and Technology, Technical University of Munich, Germany
² Department of Computing, Imperial College London, UK
³ Munich Center for Machine Learning (MCML), Munich, Germany
⁴ Institute for Machine Learning in Biomedical Imaging, Helmholtz Munich, Germany
⁵ Department of Quantitative Biomedicine, University of Zurich, Switzerland
⁶ Weill Cornell Medicine, Cornell University, New York City, USA

a.berger@tum.de, jpaetzold@med.cornell.edu

Abstract

Direct image-to-graph transformation is a challenging task that involves solving object detection and relationship prediction in a single model. Due to this task’s complexity, large training datasets are rare in many domains, making the training of deep-learning methods challenging. This data sparsity necessitates transfer learning strategies akin to the state-of-the-art in general computer vision. In this work, we introduce a set of methods enabling cross-domain and cross-dimension learning for image-to-graph transformers. We propose (1) a regularized edge sampling loss to effectively learn object relations in multiple domains with different numbers of edges, (2) a domain adaptation framework for image-to-graph transformers aligning image- and graph-level features from different domains, and (3) a projection function that allows using 2D data for training 3D transformers. We demonstrate our method’s utility in cross-domain and cross-dimension experiments, where we utilize labeled data from 2D road networks for simultaneous learning in vastly different target domains. Our method consistently outperforms standard transfer learning and self-supervised pretraining on challenging benchmarks, such as retinal or whole-brain vessel graph extraction.¹

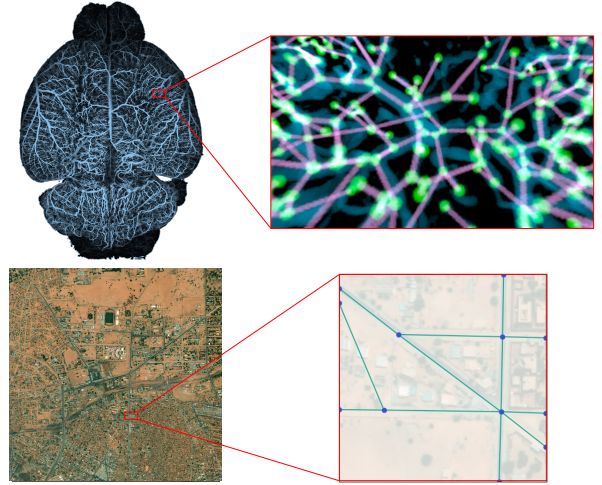


Figure 1. Direct image-to-graph transformation. Whole brain vessel (top) and Agadez road dataset (bottom). The predicted graph is visualized as an overlay on the real image.

disregard the physical structure’s semantic content. Hence, constructing graph representations from images (image-to-graph, see Fig. 1) is a critical challenge for unlocking the full potential in many real-world applications [27].

Traditionally, image-to-graph transformation involves a complex multi-stage process of segmentation, identification of physical structures and their relations, and iteratively pruning the constructed graph, which leads to inaccuracies and information loss at each step [14]. These disadvantages negatively impact prediction accuracy and limit the application to downstream tasks that require more information from the underlying image (e.g., [43, 44]). Hence, there is a clear need for machine learning solutions that facilitate accurate image-to-graph transformation in a single

1. Introduction

Representing physical relationships via graph representations has proven to be an efficient and versatile concept with vast utility in machine learning. Prominent examples are road network graphs [2], neuron representations and connections in the brain [45], blood vessels [8], and cell interactions [55]. Here, typically used voxelized images

¹Code: github.com/AlexanderHBerger/cross-dim_i2g

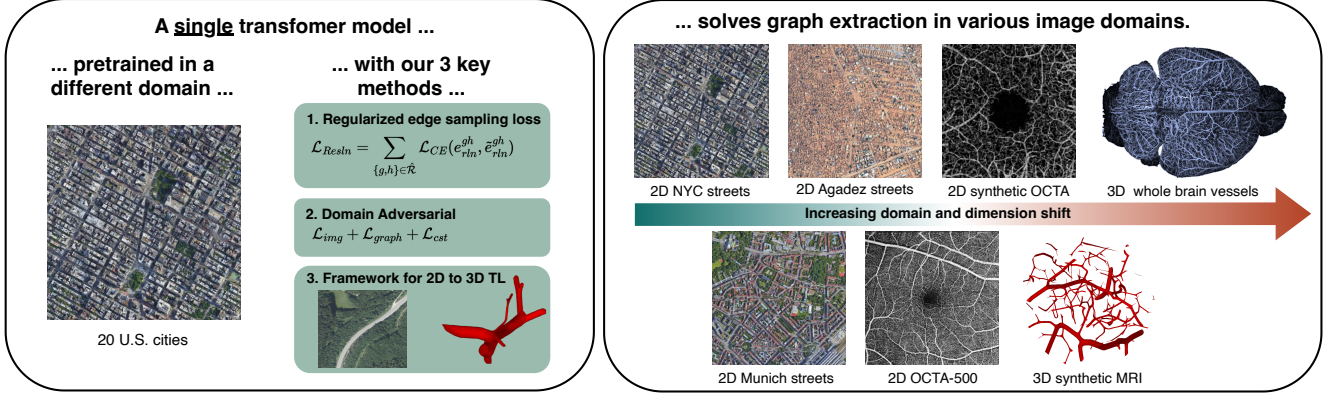


Figure 2. Conceptual overview of our framework. We use a transformer for single-stage image-to-graph transformation. Our three methodological contributions enable knowledge transfer between vastly different domains in 2D and 3D.

step [4, 22]. Recently, vision transformers have been proposed for this task and showed superior performance to traditional multi-stage approaches [42, 54]. However, these approaches require large sets of annotated 2D data and have not been shown to generalize to diverse 3D datasets where graph-level annotations are not widely available.

To address this challenge, we adopt and extend concepts from the field of inductive transfer learning (TL) (see taxonomy in [38]), which have not been explored for image-to-graph transformation. Inductive TL simultaneously uses large annotated datasets in the source domain (e.g., 2D satellite images) and a small set of labeled data in the target domain (e.g., 3D microscopy images of vascular networks).

Our contribution. Guided by the hypothesis that the *underlying graph representations of physical networks are similar across domains*, we introduce a set of methodological innovations for image-to-graph transformation:

1. We introduce a loss formulation that regularizes the number of sampled edges during training, allowing simultaneous learning in domains that differ in terms of the number of objects and relations (Sec. 3.1).
2. We propose a supervised domain adaptation framework for image-to-graph transformers that aligns features from different domains (Sec. 3.2).
3. We propose a framework for training 3D image-to-graph models with 2D data. Our framework introduces a projection function from the source representation to a space similar to the target domain (Sec. 3.4).

In extensive experiments on six datasets, we demonstrate our method’s utility and outperform all baselines, validating our initial hypothesis. Our method leads to *four main results*: 1) we show compelling improvements in image-to-graph transformation on existing datasets; 2) our method

enables image-to-graph transformation in previously unsolvable sparse data scenarios; 3) our method outperforms self-supervised pretraining by a margin and 4) our method bridges dimensions during training, i.e. we solve the previously unsolved direct image-to-graph inference for complex 3D vessel images by utilizing 2D road data in an additional training step. Further ablation studies highlight that each of our methodological contributions addresses a specific weakness in the state-of-the-art.

2. Related Works

Image-to-graph transformation. Image-to-graph transformation is an increasingly important field in computer vision with application to various domains, including road network extraction from satellite images [51] or 3D vessel graph extraction from microscopy images [1, 3]. Traditional methods solve this task through a multi-step approach involving input segmentation [2, 7, 11], skeletonization, and pruning to generate the graph [14, 51]. Deep learning-based approaches frequently use an object detector followed by relation prediction [1, 3] or require a segmentation map as input [54]. These approaches’ performances are determined by the performance of the pipeline’s intermediate stages. Furthermore, information loss at each stage limits performance and applicability to downstream tasks. Lastly, these methods are tailored to specific domains, rendering them unsuitable for cross-domain applications, including TL.

We select the *Relationformer* as our base concept because it is the most general single-stage transformer concept that can directly predict the graph from diverse image data in various domains [39, 42]. This generalizability makes it especially suitable for cross-domain TL.

Transfer learning for transformers. Recent studies showed effective pretraining for transformers on natural images. Prevailing architectures involve supervised pretrain-

ing of the model’s backbone on, e.g., ImageNet, coupled with random initialization of the encoder-decoder component [7, 33, 42]. Dai et al. [12] extend DETR [7] with a specific pretext task for object detection. Li et al. [28] compared self- and supervised pretraining methods with random initialization of a ViT-backbone [13]. They showed how self-supervised pretraining improves downstream object detection without a specific pretext task. Ma et al. [33] pretrained a transformer architecture on synthetic data and outperformed self-supervised methods for object detection. However, generating synthetic data, especially for specialized tasks, is typically domain-specific or requires expert knowledge [25]. Ma et al. [33] find that random weight initialization remains a robust baseline, often achieving competitive downstream performance. To this date, no study has explored TL for image-to-graph transformers.

Cross-domain transfer learning. Existing cross-domain TL approaches are generative or discriminative. Generative approaches translate images from source to target domain on a pixel-level using a generative network [56]. In restricted settings, they have shown promising results, e.g., object detection for day-/nighttime shifts in road scenes [23] or MRI to CT translation [21]. However, generative approaches require additional training of the translation network, which is computationally expensive and suffers from training instability [31]. Discriminative approaches train a single model to learn a general representation that is transferable between domains [50]. Some utilize a domain adversarial network that distinguishes whether a sample is from the source or target domain based on its feature representation. Combined with a gradient reversal layer (GRL) [17], this approach has proven effective in classification [17], segmentation [49], and object detection [10]. Chen et al. [10] introduced a domain adaptation (DA) framework for object detection using the GRL concept with two domain classifiers at the image and instance levels. Both are based on \mathcal{H} -divergence theory [5]. To reduce the bias towards the source domain, a consistency regularization penalizes the L_2 -distance between the domain classifiers [10]. To this day, all existing approaches are limited to a relatively small domain shift (e.g., from day to night scenes or synthetic to real images) instead of a *fundamental* domain shift, such as from satellite images to medical scans. Also, none of the existing approaches have been applied to the image-to-graph transformation problem.

Cross-dimension transfer learning. 2D to 3D TL is a highly challenging but promising research direction because of the abundance of labeled 2D data compared to the scarcity of 3D data. The few existing approaches [29, 41, 52, 53] address the challenge, either data-based or model-based. Data-based approaches augment and project

2D data into 3D space, while model-based approaches aim to adjust the model to work with multi-dimensional input. Shen et al. [41] introduced a method that projects 3D point clouds to pseudo-2D RGB images. Liu et al. [29] introduced a pixel-to-point knowledge transfer that pretrains a 3D model by generating 3D point cloud data from 2D images with a learned projection function. A model-based approach by Xie et al. [53] used dimension-specific feature extractors and a dimension-independent Transformer. Similarly, Wang et al. [52] proposed a special tokenizer that creates 2D-shaped patch embeddings in a standard 2D ViT-model. All existing approaches either require changes to the target model, which may limit performance, a specifically crafted projection function, or additional training (e.g., for learning an explicit projection function). In our approach, we seek simplicity in implementation and training as well as generalizability to new domains and tasks.

3. Methodology

In this section, we describe our three key contributions to efficiently transfer knowledge from a source domain with image and graph space denoted as a $(\mathcal{I}^S, \mathcal{G}^S)$ to a target domain with image and graph space denoted as $(\mathcal{I}^T, \mathcal{G}^T)$ for an image-to-graph transformer.

3.1. Regularized Edge Sampling Loss

A leading difference between the source and target domain in our cross-domain TL setting is the different node and edge distribution, which poses a challenge because they dictate the relation loss calculation. Previous works arbitrarily pick the number of ground truth edges and fix it for the relation loss calculation (\mathcal{L}_{rln}) for a specific dataset [42], which evidently generalizes poorly across varying domains. Generally, such a loss originates from object detection, where pair-wise relations are classified with a cross-entropy (CE) loss over a fixed number of edges m . This includes all active edges \mathcal{A} and an irregular number of randomly sampled background edges $\tilde{\mathcal{B}} \subseteq \mathcal{B}$. An active edge is defined as a pair of nodes g, h that is connected by an edge, i.e., $e_{\text{rln}}^{gh} = 1$. Similarly, for a background edge, $e_{\text{rln}}^{gh} = 0$ holds. Then, \mathcal{L}_{rln} is the CE loss on the set of sampled edges $\mathcal{R} = \mathcal{A} \cup \tilde{\mathcal{B}}$ with $|\mathcal{A}| + |\tilde{\mathcal{B}}| \leq m$:

$$\mathcal{L}_{\text{rln}} = \sum_{\{g,h\} \in \mathcal{R}} \mathcal{L}_{\text{CE}}(e_{\text{rln}}^{gh}, \tilde{e}_{\text{rln}}^{gh}) \quad (1)$$

where $\tilde{e}_{\text{rln}}^{gh} \in \{0, 1\}$ is the model’s relation prediction for the respective pair of nodes. Crucially, this formulation ignores the subset $\mathcal{B} \setminus \tilde{\mathcal{B}}$ when calculating the loss.

In previous works [42], m is a manually chosen global hyperparameter, strongly affecting the model’s performance. If m is too small, not enough background edges are sampled; hence, the loss does not penalize over-prediction.

If m is too large, background edges dominate the edge loss calculation because the edge space is sparse. In that case, the network under-predicts edges. Furthermore, with only a small subset of all edges being sampled, the loss gives a noisy signal regarding edge prediction, which worsens the learning process overall.

To address these limitations, we introduce our regularized edge sampling loss for transformers, short $\mathcal{L}_{\text{Reslt}}$. In simple terms, $\mathcal{L}_{\text{Reslt}}$ adaptively chooses the number of sampled edges. If necessary, $\mathcal{L}_{\text{Reslt}}$ upsamples the edges up to a fixed ratio between active and background edges. With our novel approach, we achieve a consistent loss across samples from different domains. Formally, we introduce our regularized edge sampling below:

$\hat{\mathcal{R}} = \hat{\mathcal{A}} \cup \hat{\mathcal{B}}$ is the set of a batch’s upsampled edges. The number of elements in the upsampled multisets $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$ have the pre-defined ratio $r = \frac{|\hat{\mathcal{A}}|}{|\hat{\mathcal{B}}|}$ where $r \in [0, 1]$. Multisets are necessary because the ratio is achieved by duplicating random edges in \mathcal{A} or \mathcal{B} .

A batch’s labels consist of a ground-truth graph \mathcal{G}_n for each sample n . \mathcal{G}_n is defined as a tuple of the sample’s nodes and edges, $\mathcal{G}_n = (\mathcal{V}_n, \mathcal{E}_n)$. The set of a batch’s nodes \mathcal{V} and edges \mathcal{E} is thus defined as:

$$\mathcal{V} = \bigcup_{n=0}^N \mathcal{V}_n \quad \text{and} \quad \mathcal{E} = \bigcup_{n=0}^N \mathcal{E}_n \quad (2)$$

Each active edge $a \in \mathcal{A} = \mathcal{E}$ is a tuple of two nodes (g, h) that are connected by the respective edge:

$$a = (g, h) \in \mathcal{E} \quad \text{where} \quad g, h \in \mathcal{V} \quad (3)$$

Each background edge $b \in \mathcal{B}$ is a tuple of two nodes (j, k) that are not connected by an edge:

$$b = (j, k) \in (\mathcal{V} \times \mathcal{V}) \setminus \mathcal{E} = \mathcal{B} \quad (4)$$

Then, the upsampled multiset of active edges $\hat{\mathcal{A}}$ is:

$$\hat{\mathcal{A}} = \mathcal{E} \cup [x_i | x_i = a_{i \bmod |\mathcal{E}|}] \quad (5)$$

with $|\mathcal{E}| \leq i < |\mathcal{B}| * r$ and $i \in \mathbb{N}$. Similarly, we define the upsampled multiset of background edges $\hat{\mathcal{B}}$ as:

$$\hat{\mathcal{B}} = \mathcal{B} \cup [x_i | x_i = b_{i \bmod |\mathcal{B}|}] \quad (6)$$

with: $|\mathcal{B}| \leq i < \frac{|\mathcal{E}|}{r}$. Notably, only one of the sets is upsampled while the other stays the same because only one of the conditions in Eq. (5) or Eq. (6) can produce a valid i .

Hence, we define our regularized edge sampling loss as:

$$\mathcal{L}_{\text{Reslt}} = \sum_{\{g, h\} \in \hat{\mathcal{R}}} \mathcal{L}_{\text{CE}}(e_{\text{rln}}^{gh}, \tilde{e}_{\text{rln}}^{gh}) \quad (7)$$

Although the edge ratio r is a hyperparameter, the model’s performance is relatively insensitive to its value. A

default value of 0.15 showed good results across all datasets (see Sec. 4 and Supplement), which makes $\mathcal{L}_{\text{Reslt}}$ beneficial compared to the previous works’ stochastic edge losses. Furthermore, $\mathcal{L}_{\text{Reslt}}$ gives a precise signal regarding edge prediction and increases convergence speed.

3.2. Supervised Domain Adaptation

In our setting, the stark differences between the source and target domain in image and graph features further amplify the TL challenge. Specifically, source and target domains significantly differ in image characteristics such as background and foreground intensities, signal-to-noise ratio, and background noise, as well as graph characteristics such as the structures’ radii or edge regularity. Edge regularity refers to the geometrical straightness of the underlying structure. While roads in the U.S. (e.g., highways) typically have a high edge regularity, vessels in microscopic images are highly irregular (i.e., the vessel does not follow a straight line and has high intra-edge curvature). To address this challenge, we utilize a domain adversarial network on the image and graph level, respectively. These adversarial networks are used when jointly pretraining in both domains. Similar to previous methods [17, 30, 48], the image-level adversarial network is a small neural network that classifies the domain based on a sample’s feature representation after the feature extractor. We treat each image patch $p^{u,v}$ at position u, v as an individual sample and compute the CE loss

$$\mathcal{L}_{\text{img}} = - \sum_{u,v} \left[D \log p^{u,v} + (1 - D) \log (1 - p^{u,v}) \right] \quad (8)$$

where $D \in \{0, 1\}$ denotes whether the respective sample is from the source or target domain [10].

To align graph-level features, we view the concatenated tokenized transformer output $T \in \mathbb{R}^{(\#o + \#r) \times d}$ as a sample’s abstract graph representation where $\#o$ and $\#r$ are the numbers of object and relation tokens, respectively, and d is the amount of the tokens’ hidden channels. We train a domain classifier on this abstract graph representation using the CE loss:

$$\mathcal{L}_{\text{graph}} = - \left[D \log T + (1 - D) \log (1 - T) \right] \quad (9)$$

Both domain classifiers are preceded by a GRL [17] reversing the gradient such that the main network is learning to maximize the domain loss. This framework forces the network to learn domain invariant representations and, thus, aligns the source and target domain. Furthermore, we apply a consistency regularization between both domain classifications to reduce the bias towards the source domain, as shown in [10]. This consistency regularization is expressed by an objective function minimizing the L_2 -distance between the domain classifiers’ predictions (i.e., the output

of the classification functions d_{img} and d_{graph}):

$$\mathcal{L}_{\text{cst}} = \left\| \frac{1}{|I|} \sum_{u,v} d_{\text{img}}(p^{u,v}) - d_{\text{graph}}(D) \right\|_2 \quad (10)$$

where we take the average over the image-level patch classifications of a sample consisting of $|I|$ patches.

3.3. Combined Training Loss

Our new regularized edge loss is combined with the other essential loss components to a final optimization function. The final loss consists of the L_1 regression loss (\mathcal{L}_{reg}), the scale-invariant, generalized intersection over union loss ($\mathcal{L}_{\text{gIoU}}$) for the box predictions (with predicted boxes \tilde{v}_{box} and ground truth v_{box}), and a CE classification loss (\mathcal{L}_{cls}) for object classification [42]. Further, our new regularized edge sampling loss $\mathcal{L}_{\text{Reslt}}$ and our three DA losses (\mathcal{L}_{img} , $\mathcal{L}_{\text{graph}}$, and \mathcal{L}_{cst}) are included. Furthermore, in order to achieve unique predictions, we compute a bipartite matching between the ground truth and predicted objects utilizing the Hungarian algorithm [42]. \mathcal{L}_{reg} , $\mathcal{L}_{\text{gIoU}}$, and $\mathcal{L}_{\text{Reslt}}$ are calculated over all object predictions v that are matched to a ground truth, i.e. where $v_{\text{cls}}^i = 1$, whereas \mathcal{L}_{cls} is calculated over all object predictions. The combined loss term for our N object tokens in a batch is defined as:

$$\begin{aligned} \mathcal{L} = & \sum_{i=1, [v_{\text{cls}}^i=1]}^N \left[\lambda_{\text{reg}} \mathcal{L}_{\text{reg}}(v_{\text{box}}^i, \tilde{v}_{\text{box}}^i) \right. \\ & \left. + \lambda_{\text{gIoU}} \mathcal{L}_{\text{gIoU}}(v_{\text{box}}^i, \tilde{v}_{\text{box}}^i) \right] \\ & + \lambda_{\text{cls}} \sum_{i=1}^N \mathcal{L}_{\text{cls}}(v_{\text{box}}^i, \tilde{v}_{\text{box}}^i) \\ & + \lambda_{\text{DA}} (\mathcal{L}_{\text{img}} + \mathcal{L}_{\text{graph}} + \mathcal{L}_{\text{cst}}) \\ & + \lambda_{\text{Reslt}} \underbrace{\sum_{\{g,h\} \in \tilde{\mathcal{R}}} \mathcal{L}_{\text{CE}}(e_{\text{rIn}}^{gh}, \tilde{e}_{\text{rIn}}^{gh})}_{\mathcal{L}_{\text{Reslt}}} \end{aligned} \quad (11)$$

with λ_{reg} , λ_{gIoU} , λ_{cls} , λ_{Reslt} , and λ_{DA} as weights.

3.4. Framework for 2D-to-3D Transfer Learning

This section describes our framework for the challenging setting of a 2D source domain and a 3D target domain. This setting is especially relevant given the scarcity of completely annotated 3D image datasets. At the core of our framework is a simple projection function Π that transforms source instances into a space similar to the target space, i.e., $\Pi : (\mathcal{I}^S, \mathcal{G}^S) \rightarrow (\tilde{\mathcal{I}}, \tilde{\mathcal{G}})$. Since our regularized edge sampling loss (Sec. 3.1) and domain adaptation framework (Sec. 3.2) automatically optimize the alignment of source and target domain characteristics, we do not need to engineer our projection to resemble the target domain characteristics (e.g., signal-to-noise ratio or the structures' radiuses).

Thus, we can design our projection function in the most simple and generalizable form. Intuitively, Π projects 2D data to a 3D space by simply creating an empty 3D volume, placing the 2D image as a frame in it, and randomly rotating the entire volume. Formally, Π is described by:

1. Resize \mathcal{I}^S from $(H^{\mathcal{I}^S} \times W^{\mathcal{I}^S})$ to the target domain's spatial patch size $(H^T \times W^T)$ by a linear down-sampling operator $D : \mathcal{I}^S \rightarrow \tilde{\mathcal{I}}^{S'}$, where $D \in \mathbb{R}^{H^T W^T \times H^{\mathcal{I}^S} W^{\mathcal{I}^S}}$. G remains unchanged as we use normalized coordinates.
2. We initialize I in 3D with $I = \mathbf{0}^{H^T W^T D^T}$ and place $\tilde{\mathcal{I}}^{S'}$ in I at slice location $z = 0.5$. We also augment the node coordinates of $G := V, E$ by $V' = \{[v, 0.5] : v \in V\}$. New graph $G' := (V', E)$.
3. We apply a random three dimensional rotation matrix R on I and obtain $\tilde{I} \in \tilde{\mathcal{I}}$. We apply the same R on the nodes of G' and obtain $V'' = \{Rv : v \in V'\}$. New graph $\tilde{G} := (V'', E) \in \tilde{\mathcal{G}}$.

Notably, our approach works out of the box without requiring segmentation masks, handcrafted augmentations, specifically engineered projections, or changes to the target model. Furthermore, it naturally extends to new domains and is trainable end-to-end together with the target task.

4. Experiments and Results

Datasets. We validate our method on a diverse set of six public image datasets capturing physical networks. We choose two 2D road datasets, namely a dataset from Munich (a European city with green vegetation-dominated land cover) and from Agadez (a historic Tuareg city in Niger in the Sahara desert). The appearance in satellite images of these cities and their network structure substantially differ from the pretraining set as well as from each other; see Fig. 2. Accurately extracting road graphs is a highly important task for traffic forecasting and traffic flow modeling [15, 34]. Next, we choose a synthetic OCTA retina dataset [35] and a real OCTA dataset [26]. Additionally, we present experiments on two 3D datasets, namely a synthetic vessel dataset [40] and a real whole-brain microscopy vessel dataset [47]. Details on the datasets and data generation can be found in the Supplement.

Training. We pretrain our method on the 20 U.S. cities dataset [22] jointly with the target dataset. We crop the source images to overlapping patches, in which we eliminate redundant nodes (i.e., nodes of degree 2 with a curvature of fewer than 160 degrees) to train our model on meaningful nodes [4]. After pretraining, we finetune the model on the target dataset for 100 epochs. For more details, please refer to the Supplement and our repository.

Table 1. Main results. Quantitative Results for our cross-domain and cross-dimensional image-to-graph transfer learning framework. The domain shift increases from top to bottom. We outperform the baselines across all datasets. The best scores per respective metric across all models for a dataset are highlighted in bold. \star Results for supervised pretraining on D) and E) (3D data) are not reported because it is technically not possible. Additional metrics and standard deviations are given in the Supplement.

Fine Tuning Training Set	(Pre-)Training Strategy	Node-mAP \uparrow	Node-mAR \uparrow	Edge-mAP \uparrow	Edge-mAR \uparrow	SMD \downarrow	Topo-Prec. \uparrow	Topo-Rec. \uparrow
A) TL from roads (2D) to roads (2D)								
Agadez [20]	No Pretraining [18]	0.067	0.122	0.021	0.043	0.062	0.369	0.261
	Self-supervised [9]	0.083	0.156	0.030	0.071	0.030	0.471	0.459
	Supervised	0.161	0.237	0.115	0.177	0.023	0.783	0.711
	Ours	0.163	0.244	0.116	0.172	0.022	0.816	0.614
Munich [20]	No Pretraining [18]	0.083	0.120	0.034	0.054	0.235	0.260	0.247
	Self-supervised [9]	0.088	0.145	0.060	0.097	0.155	0.339	0.384
	Supervised	0.277	0.336	0.207	0.272	0.091	0.682	0.660
	Ours	0.285	0.344	0.224	0.277	0.090	0.726	0.655
B) TL from roads (2D) to synthetic retinal vessels (2D)								
Synthetic OCTA [35]	No Pretraining [18]	0.273	0.375	0.140	0.339	0.005	0.181	0.948
	Self-supervised [9]	0.136	0.260	0.069	0.223	0.031	0.093	0.927
	Supervised	0.291	0.384	0.170	0.338	0.004	0.211	0.957
	Ours	0.415	0.493	0.250	0.415	0.002	0.401	0.890
C) TL from roads (2D) to real retinal vessels (2D)								
OCTA-500 [26]	No Pretraining [18]	0.189	0.282	0.108	0.169	0.017	0.737	0.634
	Self-supervised [9]	0.214	0.305	0.135	0.213	0.016	0.763	0.706
	Supervised	0.366	0.447	0.276	0.354	0.014	0.862	0.775
	Ours	0.491	0.571	0.366	0.489	0.012	0.877	0.817
D) TL from roads (2D) to brain vessels (3D)								
Synthetic MRI [40]	No Pretraining [18]	0.162	0.250	0.125	0.201	0.013	-	-
	Self-supervised [9]	0.162	0.252	0.120	0.193	0.014	-	-
	Supervised	\star	\star	\star	\star	\star	\star	\star
	Ours	0.356	0.450	0.221	0.322	0.013	-	-
E) TL from roads (2D) to real whole-brain vessel data (3D)								
Microscopic images [47]	No Pretraining [18]	0.231	0.308	0.249	0.329	0.017	-	-
	Self-supervised [9]	0.344	0.404	0.363	0.425	0.017	-	-
	Supervised	\star	\star	\star	\star	\star	\star	\star
	Ours	0.483	0.535	0.523	0.566	0.017	-	-

Metrics. We evaluate our method on six metrics from object detection and graph similarity tasks. For graph similarity, we report the 2D TOPO-score [6] and the street mover distance (SMD), which approximates the Wasserstein distance of the graph [4]. From object detection, we report mean average recall (mAR) and mean average precision (mAP) for node- and edge-detection. For more implementation details, please refer to the Supplement.

Baselines. No prior work has developed transfer learning techniques for the structural image-to-graph transformation problem. To evaluate the significance of our proposed methods, we compare the downstream task performance against three competing approaches with varying pretraining and initialization methods. Our first baseline, *no pretraining*, is random weight initialization [18], which is considered standard practice for model initialization when no suitable

pretraining is available. Second, we benchmark against a state-of-the-art method for *self-supervised pretraining*, MoCo v3 [9], which even outperformed supervised pretraining in some tasks [9]. Self-supervised pretraining is typically used when the amount of unlabeled data significantly exceeds that of labeled data in the same (or very similar) domain. Hence, we pretrain on a large set of unlabeled data from the same domain in each experiment for the self-supervised baseline. For more details regarding the unlabeled dataset, please refer to the Supplement. Third, *supervised pretraining*, where we pretrain the target model on the source data without using our methodological contributions. This approach has been successfully applied for various vision transformers, including the relationformer [42]. Note that the *supervised pretraining* baseline is impossible in 3D scenarios; only a projection function enables the use of 2D data for pretraining 3D models.

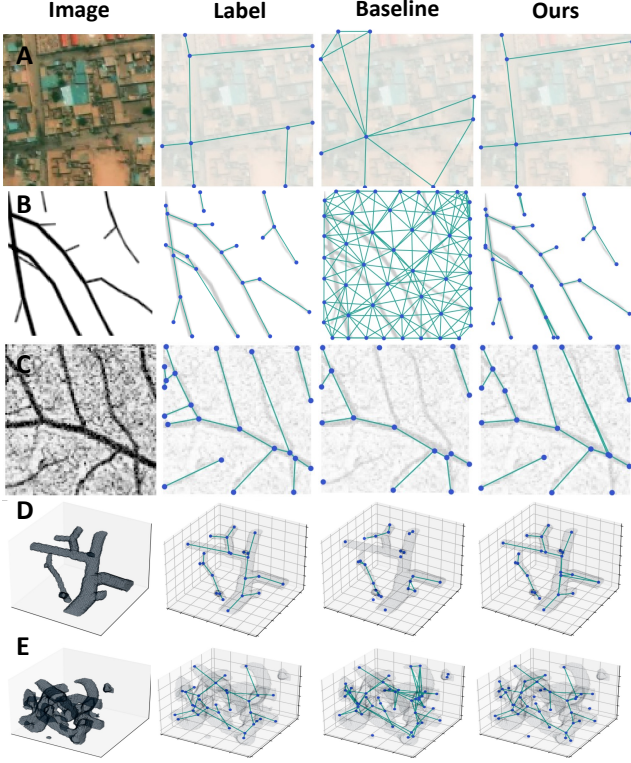


Figure 3. Qualitative results. From left to right: Image, ground truth graph, *no pretraining*-baseline, and our method. Datasets in each row are indicated by the letters for the datasets as in Tab. 1. Our method consistently outperforms the *no pretraining* baseline, which overpredicts the edges and nodes in all datasets but the OCTA-500, where the fine-tuning set is uncharacteristically large.

4.1. Results on Cross-domain TL (2D)

Our proposed transfer learning strategy shows excellent results across 2D datasets. We outperform the baseline without pretraining and self-supervised pretraining on all datasets across all object detection and graph similarity metrics; see Tab. 1. As the domain shift increases, we significantly outperform naive pretraining.

Roads in diverse locations. First, we show that we can learn to extract road graphs in topographically diverse locations with vastly variant land cover via TL. On both datasets (see Tab. 1 A), we tripled the performance across almost all metrics compared to our baseline. Our results show that edge detection fails without any form of transfer learning. Although the self-supervised method improves the baseline across all metrics, the performance does not reach the level of supervised pretraining. While our approach yields the best performance, the difference to the baseline with naive pretraining is small. We attribute this small difference to the small domain shift between the source (U.S. roads) and the

target domains, which eases knowledge transfer.

Retinal blood vessels. In the next experiment, we introduce a larger domain shift in our TL. Our target sets are two retinal blood vessel datasets (see Tab. 1 B and C). Our method doubles the node and edge detection performance on the OCTA-500 dataset [26] compared to “no pretraining.” Furthermore, we significantly increase object detection and graph similarity metrics across both datasets compared to all baselines. The qualitative examples (Fig. 3) indicate that this improvement is associated with identifying more correct nodes and edges. We observe that the self-supervised method does not improve performance on the synthetic dataset but still achieves minor improvements for OCTA-500. We attribute this to the differences between the target sets: in the OCTA-500, only the arterioles and venules are annotated, leading to an easy topological structure. The main difficulty here lies in differentiating foreground from background, a task in which contrastive self-supervised training shows excellent performance. While this differentiation is easy in the synthetic OCTA dataset, the main difficulty is learning the topological structure with many (often overlapping) edges (see Fig. 3 and the Supplement). Learning this complex structure requires label information, as visible in the superior performance of the supervised pretraining methods. Naive pretraining still improves performance, but we observe a large (compared to A) performance difference between naive pretraining and our method. We attribute this to the larger domain shift, which is better addressed by our proposed methodology.

4.2. Results on Cross-dimension TL (2D to 3D)

Finally, we explore dimensional shifts in addition to a stark domain shift. Leveraging our new proposed loss (Sec. 3.1), our DA framework (Sec. 3.2), and our 2D-3D projection function (Sec. 3.4), we pretrain models on raw satellite images for the challenging task of 3D vessel graph extraction on a synthetic and a real dataset. Our experiments on the VesSAP dataset [47] show strong improvements in all graph similarity and object detection scores (see Tab. 1 E). The self-supervised method also displays improvements, which, however, do not reach our method’s performance. Similarly, our method roughly doubles the object detection metrics on the 3D MRI dataset compared to the baselines (see Tab. 1 D).

We do not report results for naive pretraining because, in contrast to our method, it simply does not allow for 2D-to-3D TL. Our ablation in Sec. 4.3 shows results for applying our projection function only, without any use of our other proposed contributions. When studying the lower-performing baselines, we observe that self-supervised pretraining with MoCo v3 leads to higher improvements in the real microscopic vessel data compared to the synthetic

Table 2. Ablation study on our proposed Loss ($\mathcal{L}_{\text{Reslt}}$) and DA framework in a transfer learning setting (experiments congruent to Tab. 1). Performance improvements are associated with both our loss and DA. Both components combined lead to the best results.

Experiment	$\mathcal{L}_{\text{Reslt}}$	DA	Node mAP	Node mAR	Edge mAP	Edge mAR
C	✗	✗	0.389	0.475	0.294	0.383
	✗	✓	0.456	0.538	0.351	0.464
	✓	✓	0.491	0.571	0.366	0.489
D	✗	✗	0.190	0.285	0.122	0.210
	✗	✓	0.349	0.443	0.219	0.320
	✓	✓	0.356	0.450	0.221	0.322

Table 3. Ablation study on our proposed Loss ($\mathcal{L}_{\text{Reslt}}$) with and without pretraining (experiments congruent to Tab. 1). We find that our loss improves performance in an arbitrary image-to-graph learning setting. Here, the DA is employed in all experiments.

Experiment	Pre-training	$\mathcal{L}_{\text{Reslt}}$	Node mAP	Node mAR	Edge mAP	Edge mAR
E	✗	✗	0.231	0.308	0.249	0.329
	✗	✓	0.410	0.464	0.468	0.512
	✓	✗	0.424	0.488	0.449	0.510
	✓	✓	0.483	0.535	0.523	0.566

MRI dataset. Both datasets have complex topologies that require supervised training (see Sec. 4.1), but only the real dataset has high-intensity variations, which can be efficiently learned in a self-supervised setting. The qualitative results in Fig. 3 and the Supplement indicate that the 3D tasks were often unsolvable without our contributions.

4.3. Ablations on our Methods

In Tab. 2 and 3, we present ablations on the regularized edge sampling loss (3.1) and DA framework (3.2) for the 2D OCTA-500 [26] (Tab. 2 C), the 3D MRI [40] (Tab. 2 D), and the 3D brain vessel dataset [47] (Tab. 3 E). Expectedly, we observe that our DA alone leads to compelling performance gains for the 3D setting (almost double the performance) and 2D setting across all metrics; see Tab. 2. This is expected since the domain shift between the source dataset of satellite images and our medical images is large. Note that our projection function is always employed for the 3D dataset since pretraining is otherwise impossible. We further observe that employing the projection function alone diminishes the performance because of the domain gap, which is only alleviated by our other contributions.

Next, we ablate on our loss. When applying the DA, adding our $\mathcal{L}_{\text{Reslt}}$ loss further improves the performance (Tab. 2), indicating its strength in stabilizing and improving the loss landscape to train better networks. Additionally, we ablate our proposed $\mathcal{L}_{\text{Reslt}}$ in an experiment with and without TL. Importantly, our experiments show that $\mathcal{L}_{\text{Reslt}}$ is

a general contribution that improves image-to-graph transformation for TL as well as for general network training (Tab. 3). In a TL setting, $\mathcal{L}_{\text{Reslt}}$ is particularly useful as it reduces the data-specific hyperparameter search. Interestingly, our loss improves not only edge detection but also node detection metrics across our ablations. These improvements can be attributed to the transformer’s cross-attention modules, which treat node and edge detection as joint prediction tasks instead of separate problems. Consequently, both metrics improve jointly. For further ablation studies, e.g., experiments without the domain adversarial networks or an alternative over-sampling of the parameter r , please refer to the Supplement. In conclusion, we note that each individual contribution enhances the overall performance of the graph prediction task.

5. Discussion and Conclusion

In this work, we propose a framework for cross-domain and cross-dimension transfer learning for image-to-graph transformers. At the core of this work are our strong empirical results, which show that our proposed inductive transfer learning method outperforms competing approaches across six benchmark datasets that contain 2D and 3D images by a margin. We achieve these results through our three methodological contributions, which we ablate individually. We conclude that transfer learning has the potential to substantially reduce data requirements for highly complex geometric deep learning tasks, such as transformer-based image-to-graph inference, see Supplement. Our work shows that this holds especially when the targeted graph representations are defined by a similar physical principle or physical network. In the presented work, this shared principle is the transport of physical units (cars and blood) in a physical network.

Limitations and future work. Our problem setting is specific to image-to-graph tasks and our learning scenario in which we have some labeled data in both domains. Future work should investigate how our solution translates to different settings. Furthermore, we use a dimensionality-dependent feature extractor, which might limit generalizability to other dimensions. Future work should explore the development of strong dimension-invariant graph extractors to allow further generalization.

Acknowledgments. AB is supported by the Stiftung der Deutschen Wirtschaft. MM is funded by the German Research Foundation under project 532139938. GK received support from the German Ministry of Education and Research and the Medical Informatics Initiative as part of the PrivateAIM Project and from the Bavarian Collaborative Research Project PRIPREKI of the Free State of Bavaria Funding Programme "Artificial Intelligence – Data Science". This work was partially funded by the Konrad Zuse School of Excellence in Reliable AI (relAI).

References

- [1] Sepideh Almasi, Xiaoyin Xu, Ayal Ben-Zvi, Baptiste Lacoste, Chenghua Gu, and Eric L Miller. A novel method for identifying a graph-based representation of 3-d microvascular networks from fluorescence microscopy image stacks. *Medical image analysis*, 20(1):208–223, 2015. [2](#)
- [2] Favyen Bastani, Songtao He, Sofiane Abbar, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Sam Madden, and David DeWitt. Roadtracer: Automatic extraction of road networks from aerial images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4720–4728, 2018. [1](#), [2](#)
- [3] James Batten, Matthew Sinclair, Ben Glocker, and Michiel Schaap. Image to tree with recursive prompting, 2023. [2](#)
- [4] Davide Belli and Thomas Kipf. Image-conditioned graph generation for road network extraction, 2019. [2](#), [5](#), [6](#), [3](#)
- [5] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010. [3](#)
- [6] James Biagioni and Jakob Eriksson. Inferring road maps from global positioning system traces: Survey and comparative evaluation. *Transportation Research Record*, 2291(1):61–71, 2012. [6](#), [5](#)
- [7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 213–229. Springer, 2020. [2](#), [3](#)
- [8] Chunhui Chen, Joon Huang Chuah, Raza Ali, and Yizhou Wang. Retinal vessel segmentation using deep learning: a review. *IEEE Access*, 9:111985–112004, 2021. [1](#)
- [9] X. Chen, S. Xie, and K. He. An empirical study of training self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9620–9629, Los Alamitos, CA, USA, oct 2021. IEEE Computer Society. [6](#), [2](#), [7](#)
- [10] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3339–3348, 2018. [3](#), [4](#), [1](#)
- [11] Ziyi Chen, Liai Deng, Yuhua Luo, Dilong Li, José Marcato Junior, Wesley Nunes Gonçalves, Abdul Awal Md Nurunnabi, Jonathan Li, Cheng Wang, and Deren Li. Road extraction in remote sensing data: A survey. *International Journal of Applied Earth Observation and Geoinformation*, 112:102833, 2022. [2](#)
- [12] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1601–1610, June 2021. [3](#)
- [13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. [3](#)
- [14] Dominik Drees, Aaron Scherzinger, René Hägerling, Friedemann Kiefer, and Xiaoyi Jiang. Scalable robust graph and feature extraction for arbitrary vessel networks in large volumetric datasets. *BMC bioinformatics*, 22(1):1–28, 2021. [1](#), [2](#), [4](#)
- [15] Alireza Ermagun and David Levinson. Spatiotemporal traffic forecasting: review and proposed directions. *Transport Reviews*, 38(6):786–814, 2018. [5](#)
- [16] Ali Ertürk, Klaus Becker, Nina Jährling, Christoph P Mauch, Caroline D Hojer, Jackson G Egen, Farida Hellal, Frank Bradke, Morgan Sheng, and Hans-Ulrich Dodt. Three-dimensional imaging of solvent-cleared organs using 3disco. *Nature protocols*, 7(11):1983–1995, 2012. [5](#), [6](#)
- [17] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015. [3](#), [4](#)
- [18] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010. [6](#), [2](#), [7](#)
- [19] Google. Google static maps api. [3](#)
- [20] Mordechai Haklay and Patrick Weber. Openstreetmap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008. [6](#), [3](#), [7](#)
- [21] Maryam Hammami, Denis Friboulet, and Razmig Kéchichian. Cycle gan-based data augmentation for multi-organ detection in ct images via yolo. In *2020 IEEE international conference on image processing (ICIP)*, pages 390–393. IEEE, 2020. [3](#)
- [22] Songtao He, Favyen Bastani, Satvat Jagwani, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Mohamed M Elsharif, Samuel Madden, and Mohammad Amin Sadeghi. Sat2graph: Road graph extraction through graph-tensor encoding. In *European Conference on Computer Vision*, pages 51–67. Springer, 2020. [2](#), [5](#), [1](#), [3](#), [6](#)
- [23] Sheng-Wei Huang, Che-Tsung Lin, Shu-Ping Chen, Yen-Yi Wu, Po-Hao Hsu, and Shang-Hong Lai. Auggan: Cross domain adaptation with gan-based data augmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 718–731, 2018. [3](#)
- [24] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR, 2019. [3](#)
- [25] Linus Kreitner, Johannes C Paetzold, Nikolaus Rauch, Chen Chen, Ahmed M Hagag, Alaa E Fayed, Sobha Sivaprasad, Sebastian Rausch, Julian Weichsel, Bjoern H Menze, et al. Detailed retinal vessel segmentation without human annotations using simulated optical coherence tomography angiographs. *arXiv preprint arXiv:2306.10941*, 2023. [3](#)
- [26] Mingchao Li, Yuhua Zhang, Zexuan Ji, Keren Xie, Songtao Yuan, Qinghuai Liu, and Qiang Chen. Ipn-v2 and octa-

- 500: Methodology and dataset for retinal image segmentation. *arXiv preprint arXiv:2012.07261*, 2020. 5, 6, 7, 8, 4
- [27] Michelle M Li, Kexin Huang, and Marinka Zitnik. Graph representation learning in biomedicine and healthcare. *Nature Biomedical Engineering*, pages 1–17, 2022. 1
- [28] Yanghao Li, Saining Xie, Xinlei Chen, Piotr Dollár, Kaiming He, and Ross B. Girshick. Benchmarking detection transfer learning with vision transformers. *ArXiv*, abs/2111.11429, 2021. 3
- [29] Yueh-Cheng Liu, Yu-Kai Huang, Hung-Yueh Chiang, Hung-Ting Su, Zhe-Yu Liu, Chin-Tang Chen, Ching-Yu Tseng, and Winston H. Hsu. Learning from 2d: Contrastive pixel-to-point knowledge transfer for 3d pretraining, 2021. 3
- [30] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018. 4
- [31] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. *Advances in neural information processing systems*, 31, 2018. 3
- [32] Yuhui Ma, Huaying Hao, Jianyang Xie, Huazhu Fu, Jiong Zhang, Jianlong Yang, Zhen Wang, Jiang Liu, Yalin Zheng, and Yitian Zhao. Rose: a retinal oct-angiography vessel segmentation dataset and new model. *IEEE Transactions on Medical Imaging*, 40(3):928–939, 2021. 4, 6
- [33] Yan Ma, Weicong Liang, Yiduo Hao, Bohan Chen, Xiangyu Yue, Chao Zhang, and Yuhui Yuan. Revisiting detr pre-training for object detection. *arXiv preprint arXiv:2308.01300*, 2023. 3
- [34] Henry Martin, Dominik Bucher, Ye Hong, René Buffat, Christian Rupprecht, and Martin Raubal. Graph-resnets for short-term traffic forecasts in almost unknown cities. In *NeurIPS 2019 Competition and Demonstration Track*, pages 153–163. PMLR, 2020. 5
- [35] Martin J Menten, Johannes C Paetzold, Alina Dima, Björn H Menze, Benjamin Knier, and Daniel Rueckert. Physiology-based simulation of the retinal vasculature enables annotation-free segmentation of oct angiographs. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 330–340. Springer, 2022. 5, 6, 1, 4, 7
- [36] Rafael Padilla, Sergio L. Netto, and Eduardo A. B. da Silva. A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 237–242, 2020. 5
- [37] Johannes C Paetzold, Julian McGinnis, Suprosanna Shit, Ivan Ezhov, Paul Büschl, Chinmay Prabhakar, Anjany Sekuboyina, Mihail Todorov, Georgios Kaissis, Ali Ertürk, et al. Whole brain vessel graphs: A dataset and benchmark for graph learning and neuroscience. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. 5
- [38] Sinno Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge, data engineering*, 2009. 2
- [39] Chinmay Prabhakar, Suprosanna Shit, Johannes C Paetzold, Ivan Ezhov, Rajat Koner, Hongwei Li, Florian Sebastian Kofler, et al. Vesselformer: Towards complete 3d vessel graph generation from images. In *Medical Imaging with Deep Learning*, 2023. 2
- [40] Matthias Schneider et al. Tissue metabolism driven arterial tree generation. *Med Image Anal.*, 16(7):1397–1414, 2012. 5, 6, 8, 4, 7
- [41] Xiaoke Shen and Ioannis Stamos. simcrosstrans: A simple cross-modality transfer learning for object detection with convnets or vision transformers, 2022. 3
- [42] Suprosanna Shit, Rajat Koner, Bastian Wittmann, Johannes Paetzold, Ivan Ezhov, Hongwei Li, Jiazhen Pan, Sahand Sharifzadeh, Georgios Kaissis, Volker Tresp, et al. Relation-former: A unified framework for image-to-graph generation. In *ECCV 2022: 17th European Conference on Computer Vision, October 2022*, 2022. 2, 3, 5, 6, 1
- [43] Suprosanna Shit, Chinmay Prabhakar, Johannes C Paetzold, Martin J Menten, Bastian Wittmann, Ivan Ezhov, et al. Graflow: Neural blood flow solver for vascular graph. In *Geometric Deep Learning in Medical Image Analysis (Extended abstracts)*, 2022. 1
- [44] Jannik Sobisch, Žiga Bizjak, Aichi Chien, and Žiga Špiclin. Automated intracranial vessel labeling with learning boosted by vessel connectivity, radii and spatial context. In *Geometric Deep Learning in Medical Image Analysis*, pages 34–44. PMLR, 2022. 1
- [45] Tzu-An Song, Samadrita Roy Chowdhury, Fan Yang, Heidi Jacobs, Georges El Fakhri, Quanzheng Li, Keith Johnson, and Joyita Dutta. Graph convolutional neural networks for alzheimer’s disease classification. In *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*, pages 414–417. IEEE, 2019. 1
- [46] Giles Tetteh et al. Deepvesselnet: Vessel segmentation, centerline prediction, and bifurcation detection in 3-d angiographic volumes. *arXiv preprint arXiv:1803.09340*, 2018. 4
- [47] Mihail Ivilinov Todorov*, Johannes Paetzold*, Oliver Schoppe, Giles Tetteh, Suprosanna Shit, Velizar Efremov, Katalin Todorov-Völgyi, Marco Düring, Martin Dichgans, Marie Piraud, et al. Machine learning analysis of whole mouse brain vasculature. *Nature Methods*, 17(4):442–449, 2020. 5, 6, 7, 8, 2
- [48] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017. 4
- [49] Haoran Wang, Tong Shen, Wei Zhang, Ling-Yu Duan, and Tao Mei. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *European conference on computer vision*, pages 642–659. Springer, 2020. 3
- [50] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018. 3
- [51] Weixing Wang, Nan Yang, Yi Zhang, Fengping Wang, Ting Cao, and Patrik Eklund. A review of road extraction from remote sensing images. *Journal of traffic and transportation engineering (english edition)*, 3(3):271–282, 2016. 2

- [52] Yi Wang, Zhiwen Fan, Tianlong Chen, Hehe Fan, and Zhangyang Wang. Can we solve 3d vision tasks starting from a 2d vision transformer?, 2022. 3
- [53] Yutong Xie, Jianpeng Zhang, Yong Xia, and Qi Wu. Unified 2d and 3d pre-training for medical image classification and segmentation. *arXiv preprint arXiv:2112.09356*, 2021. 3
- [54] Zhenhua Xu, Yuxuan Liu, Yuxiang Sun, Ming Liu, and Lujia Wang. Rngdet++: Road network graph detection by transformer with instance segmentation and multi-scale features enhancement. *arXiv preprint arXiv:2209.10150*, 2022. 2
- [55] Yanning Zhou, Simon Graham, Navid Alemi Koohbanani, Muhammad Shaban, Pheng-Ann Heng, and Nasir Rajpoot. Cgc-net: Cell graph convolutional network for grading of colorectal cancer histology images. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, 2019. 1
- [56] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 3

Supplementary Material

Cross-domain and Cross-dimension Learning for Image-to-Graph Transformers

6. Additional ablation studies

Additionally to our main experiments, we present ablation studies on further aspects of our proposed framework. These ablation studies give a deeper understanding of the components' dynamics and guide future reimplementations and adaptations.

6.1. Generalizability

In Table 4, we present the results of our experiment E) (see Table 1) with one additional configuration. In this configuration, we pretrain the model on the synthetic OCTA dataset [35] instead of the U.S. cities dataset [22]. This pretraining strategy outperforms all baselines and increases the performance compared to our main experimental setting (see Section 4). We attribute this improvement to the smaller domain gap between the new source domain (i.e., retinal blood vessels) and the target domain (i.e., a mouse's cerebrovasculature). These results show how our method generalizes seamlessly to new domains. Furthermore, they substantiate the rationale behind our experimental design: by showcasing the utility of our method in a challenging setting, focused on the most intricate transfer learning scenarios, we establish its effectiveness in more straightforward transfer learning situations (as presented in Table 4 as well).

6.2. Regularized edge sampling loss

Next, we conduct an ablation study on the effect of the regularized edge sampling loss. As explained in Section 4, the new loss stabilizes training and increases convergence speed. This effect is shown in Figure 4, where the training loss decreases faster from the beginning on and converges towards a lower level compared to the baseline loss formulation. This effect can be observed across all datasets and training strategies. Also, we experiment with different foreground-to-background-edge ratios r (see Section 3.1). Table 5 shows that the performance stays stable across a large range of r -values. These results underline our hypothesis from Section 3.1 that \mathcal{L}_{ResIn} reduces the hyperparameter space because it does not require careful optimization.

Furthermore, we study the effect of different edge-sampling strategies on our loss formulation in Table 6. Specifically, we compare our fixed-ratio upsampling strategy with a varying- r upsampling (i.e., for each batch, we randomly choose r with a uniform distribution in $(, 1]$), and a fixed-ratio subsampling strategy. The decreased performance with a varying- r upsampling strategy shows that a fixed r is important for our loss formulation. We further find that subsampling is a valid alternative in scenarios where data is extremely scarce (e.g., Experiment A) but performs worse when more data is available (e.g., Experiment E). Notably, Shit et al. [42] proposed a one-sided subsampling strategy, i.e., subsampling only the background edges if the ratio is **above** a certain ratio. This strategy is problematic when the target dataset contains dense graphs, in which our loss formulation upsamples the background edges (see Table 7 for dataset statistics). Furthermore, the official relationformer repository does use a dynamic subsampling

strategy but selects background edges up to an absolute threshold m , which introduces strong hyperparameter sensitivity. Table 7 shows that up- or sub-sampling only one edge type (e.g., the background edges) would not be sufficient.

6.3. Domain adaptation framework

Table 8 shows an ablation study of our domain adaptation framework's components. \mathcal{L}_{img} , \mathcal{L}_{graph} , and \mathcal{L}_{cst} refer to the optimization terms from Section 3.2. Using the image-level alignment alone already yields a performance increase of around 30 % compared to not using our framework at all. We attribute this observation to the large image-level differences between the source and target domain, which hinders knowledge transfer in the feature extractor if an adversarial does not mitigate it. The graph-level adversarial slightly decreases the performance when being applied without consistency regularization (i.e., \mathcal{L}_{cst}). This decrease is likely caused by the abstraction level of the transformer's tokenized graph representation. Without any further guidance (e.g., by the image-level domain classifier through consistency regularization), the graph-level classifier does not provide a precise gradient toward a domain-invariant representation. Combining all three components yields the best results, supporting our hypothesis that the graph-level adversarial needs regularization by the image-level adversarial.

Furthermore, we study the impact of our projection function and loss formulation without applying our domain adaptation framework. Table 9 shows that our other contributions alone enable transfer learning across dimensions. This enables transfer learning without access to the target domain during pretraining. However, even in these cases, our DA framework yields the best performance. Table 2 shows a similar trend in cases without dimension shift.

6.4. Adversarial learning coefficient

In Table 10, we ablate on the domain adversarial learning coefficient α . α is the factor with which the gradient in the GRL is multiplied before passing it to the respective model component, i.e., the feature extractor for the image-level adversarial and the encoder-decoder for the graph-level adversarial (see Section 3.2). We use the α schedule proposed by Chen et al. [10], which increases α during the training until reaching a fixed maximum. Table 10 shows that the right choice of α is crucial and that a sub-optimal value can decrease downstream performance. We attribute this observation to the model's tradeoff between learning to produce domain-invariant features (i.e., domain confusion) and task learning (i.e., graph extraction). If α is too large, the adversarial loss dominates the task loss, and the network does not learn how to produce meaningful features. If it is too small, the domain gap between the source and target domain stays too large, and knowledge transfer is impeded. Figure 5 shows how a small α (e.g., $\alpha = 0.3$) is not sufficient to learn domain-invariant features while an α -value that is too large does not increase domain confusion but obstructs learning the core task. Note that the specific α value must be optimized for the used datasets and is not domain-invariant.

6.5. Target dataset size

Lastly, Figure 6 shows the results of an ablation study on the target dataset size. We plot the harmonic mean of node and edge

Table 4. Ablation study on the pretraining dataset. As the domain gap between the source and target domain decreases, the downstream performance increases. We show that our method is generalizable across different pretraining datasets.

Fine Tuning Training Set	(Pre-)Training Strategy	Node-mAP \uparrow	Node-mAR \uparrow	Edge-mAP \uparrow	Edge-mAR \uparrow	SMD \downarrow
Microscopic images [47]	No Pretraining [18]	0.231	0.308	0.249	0.329	0.017
	Self-supervised [9]	0.344	0.404	0.363	0.425	0.017
	Supervised	*	*	*	*	*
	Ours, pretr. on cities	0.483	0.535	0.523	0.566	0.017
	Ours, pretr. on OCTA	0.548	0.583	0.588	0.615	0.016

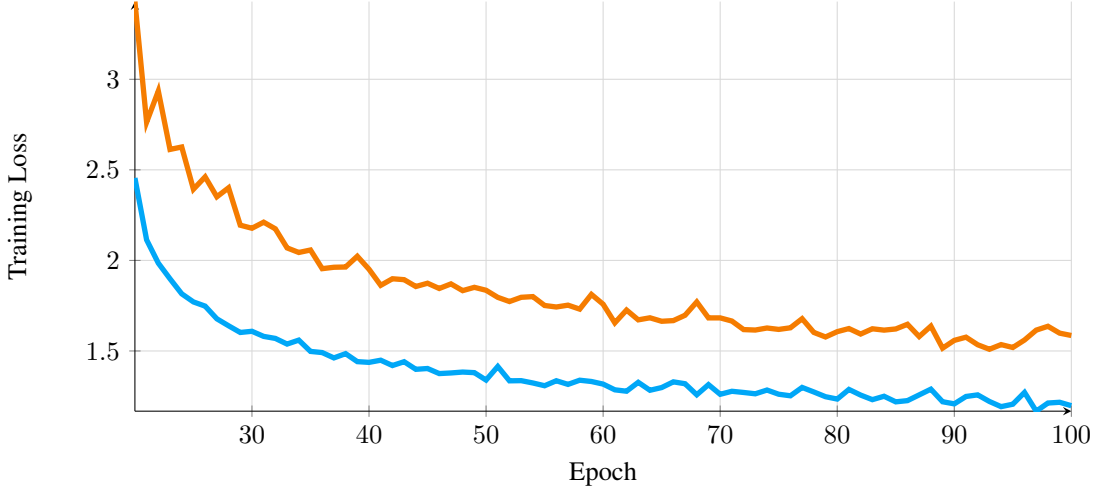


Figure 4. Training loss curves. The orange line depicts the training loss without our regularized edge sampling loss \mathcal{L}_{Resln} and the blue line with \mathcal{L}_{Resln} , respectively. \mathcal{L}_{Resln} shows faster convergence from the beginning on.

Table 5. Ablation study on the loss ratio r for \mathcal{L}_{Resln} as described in Section 3.1 (experiments congruent to Table 1). We observe that \mathcal{L}_{Resln} is stable across varying loss ratios and does not require sensitive hyperparameter tuning.

Experiment	r	Node mAP	Node mAR	Edge mAP	Edge mAR
D	0.05	0.3539	0.4446	0.2166	0.3102
	0.1	0.3564	0.4498	0.2209	0.3218
	0.15	0.3470	0.4380	0.2164	0.3122
	0.2	0.3532	0.4449	0.2203	0.3193
	0.3	0.3451	0.4351	0.2183	0.3153
	0.5	0.3470	0.4407	0.2231	0.3242
	0.8	0.3462	0.4394	0.2253	0.3288

mAP (see Section 4) of our method and the *no-pretraining* baseline against the size of the target dataset. We observe that our method consistently outperforms the baseline across all dataset sizes. However, as the number of samples increases, the performance difference between the two methods decreases. This observation is expected because transfer learning becomes less effective (and is also less required) when enough target domain samples are available. Our framework is especially useful if target data is

scarce.

7. Model & training details

To find the optimal hyperparameters, we follow a three-step approach. First, we optimize the model architecture hyperparameters (e.g., model size) with a random weight initialization (i.e., no transfer learning) on the target task. Then, we fix these hyperparameters for the remainder of the optimization process. An overview of the model hyperparameters for each experiment can be found in Table 11. Second, we optimize the training hyperparameters (e.g., learning rate or batch size) for pretraining on the source task with the fixed model architecture hyperparameters from step 1. Third, we use the pretrained model with the best performance on the source task and optimize the training parameters on the target task for each training strategy separately on the validation set. We follow this approach because optimizing the whole pipeline (including pretraining and fine-tuning) in a brute-force manner would require too many resources in terms of computational power and energy consumption. Table 12 depicts the training hyperparameters for the target task for all the experiments listed in Section 4.

Table 6. Ablation study on different edge sampling strategies.

Strategy	Experiment A		Experiment E	
	node-mAP	edge-mAP	node-mAP	edge-mAP
subsampling	0.172	0.125	0.237	0.277
varying-r	0.156	0.115	0.218	0.134
oversampling (ours)	0.173	0.129	0.267	0.323

Table 7. Edge statistics for the used datasets. The varying ratios between active and background edges underline the utility of our dynamic loss formulation. Different datasets require upsampling for active and background edges.

Dataset	Avg. Edges	Avg. edge ratio	Upsampling background	Upsampling active
20 U.S. Cities	6.37	0.53	92.5%	7.5%
Agadez	5.05	0.75	96.2%	3.8%
Munich	4.69	0.77	97.1%	2.9%
Synth. OCTA	32.57	0.05	1.4%	98.6%
OCTA-500	11.07	0.18	47.5%	52.5%
Synth. MRI	22.37	0.07	0.3%	99.7%
Microscopy	33.31	0.05	0.3%	99.7%

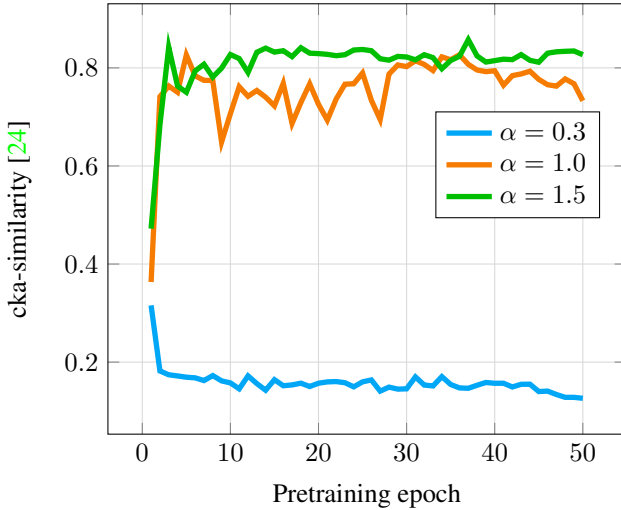


Figure 5. cka-similarity [24] (y-axis) between the feature representations of source and target domain during pretraining. α must be sufficiently large such that the similarity increases during training. From a certain threshold on, the similarity does not increase further. We associate a high similarity between both domains with the model learning domain-invariant features.

8. Datasets

In the following, we describe the properties and sampling of our six diverse image datasets and the unlabeled datasets we used for the self-supervised baseline.

8.1. Training set - 20 U.S. Cities

[22] is a city-scale dataset consisting of satellite remote sensing (SRS) images from 20 U.S. cities and their road graphs covering

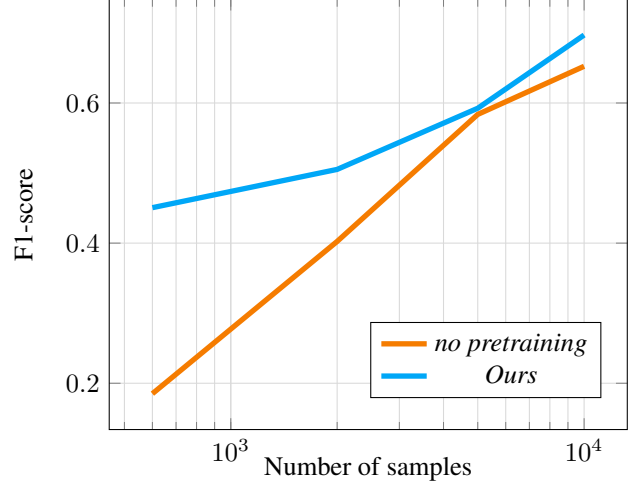


Figure 6. F1-scores (y-axis) over different target dataset sizes (x-axis). The F1-score is calculated between the node and edge mAP as described in Section 9. The orange line depicts the F1-scores of the *no-pretraining* baseline, and the blue line with our contributions, as described in Section 4. The x-axis is in logarithmic scale. We observe that our contributions are significantly reducing data requirements, especially when data is scarce.

a total area of 720 km². The satellite images are retrieved in the RGB format via the Google Maps API [19]. The corresponding road network graphs are extracted from OpenStreetMap [20]. We cut the resulting images and labels into overlapping patches of 128x128 pixels with a spatial resolution of one meter per pixel. In these patches, we eliminate redundant nodes (i.e., nodes of degree 2 with a curvature of fewer than 160 degrees) to simplify the prediction task [4].

8.2. Agadez and Munich, cities around the globe

We create our own image dataset from OpenStreetMap² covering areas that differ from those covered by the 20 U.S. cities dataset in terms of geographical and structural characteristics. Geographical characteristics refer to the area’s natural features (e.g., vegetation), while structural characteristics relate to anthropogenic (human-made) structures that affect an area’s surface (e.g., street type) or layout (e.g., city type). The complete dataset contains a 4 km² area of 11 cities with different characteristics in different parts of the world. Both source images and labels were obtained in the same manner as for the 20 U.S. cities dataset [22]. Our dataset is

²<https://www.openstreetmap.org>

Table 8. Ablation study on our domain adaptation framework in a transfer learning setting (experiments congruent to Table 1). \mathcal{L}_{img} , \mathcal{L}_{graph} , and \mathcal{L}_{cst} refer to the optimization terms from Section 3.2. We find that performance improvements are associated with all adaptation components. Using the complete optimization term as presented in Section 3.3 yields the best results.

Experiment	\mathcal{L}_{img}	\mathcal{L}_{graph}	\mathcal{L}_{cst}	Node mAP	Node mAR	Edge mAP	Edge mAR
C	\times	\times	\times	0.3423	0.4341	0.2581	0.3414
	\checkmark	\times	\times	0.4286	0.5073	0.3293	0.4264
	\times	\checkmark	\times	0.3264	0.4136	0.2355	0.3117
	\checkmark	\checkmark	\times	0.4071	0.4980	0.2685	0.3831
	\checkmark	\checkmark	\checkmark	0.4909	0.5712	0.3656	0.4887

Table 9. The performance of our contributions with and without our DA framework in the additional experiment from Table 4. We show that smaller domain gaps (here, from OCTA to microscopy images) can be bridged without our DA framework even with dimension shift.

Method	node-mAP	node-mAR	edge-mAP	edge-mAR
No Pretraining	0.231	0.308	0.249	0.329
No DA	0.508	0.549	0.551	0.584
Ours	0.548	0.583	0.588	0.615

Table 10. Ablation study on the domain adversarial learning coefficient α (experiments congruent to Table 1). α must be optimized such that the adversarial loss balances with the graph extraction loss. An α value of 0 is equivalent to not using the domain adaptation framework.

Experiment	α	Node mAP	Node mAR	Edge mAP	Edge mAR
C	0.0	0.3884	0.4755	0.2947	0.3767
	0.1	0.3123	0.4076	0.2258	0.3044
	0.3	0.3381	0.4287	0.2439	0.3240
	0.5	0.4563	0.5395	0.3614	0.4618
	0.8	0.4623	0.5458	0.3464	0.4687
	1.0	0.4909	0.5712	0.3656	0.4887
	1.5	0.3914	0.4726	0.2854	0.3897
	2.0	0.0530	0.1719	0.0214	0.0451

accessible in our GitHub repository ³.

For our experiments, we choose two cities, Agadez and Munich, whose characteristics differ from the 20 U.S. cities dataset in different aspects as displayed in Table 13. We strategically choose those cities to investigate how differences in specific characteristics between the source and target domain affect knowledge transfer and how transfer learning strategies should be adapted to these differences. We especially test our hypothesis that surface-level characteristics are captured by different components than layout-level characteristics. These new datasets enable the verification because Agadez differs from 20 U.S. cities in surface-level characteristics (e.g., vegetation, street type, and buildings) but shares a similar city layout (i.e., grid plan). Note that although Agadez

has a historical city center, we chose a part of the city that follows the typical grid layout. Contrary to this, Munich is similar to U.S. cities in surface-level characteristics while following a different city layout (i.e., a historical European city layout). We test each city dataset separately.

8.3. Synthetic OCTA

The synthetic Optical Coherence Tomography Angiography (OCTA) dataset [35] consists of synthetic OCTA scans with intrinsically matching ground truth labels, namely the corresponding segmentation map and the vessel graphs. The images were created using a simulation based on the physiological principles of angiogenesis to replicate the intricate retinal vascular plexuses [40], followed by incorporating physics-based modifications to emulate the image acquisition process of OCTA along with the usual artifacts. We project the 3D OCTA images along the main axis, split them scan-wise between training and testing sets, and extract 2600 overlapping samples of 128×128 pixels. For training our self-supervised baseline, we use the same procedure on 200 additional synthetic OCTA scans to extract almost 100,000 patches.

8.4. OCTA-500

The OCTA-500 dataset [26] includes 300 OCTA scans with a $6 \text{ mm} \times 6 \text{ mm}$ field of view. The 400×400 large *en-face* projection images were manually annotated with sparse vessel labels. We extract the graphs from these segmentation maps using the method presented by Drees et al. [14]. We split the scans patient-wise between training and testing sets and create around 3000 overlapping patches with a spatial size of 128×128 . Furthermore, we combine the OCTA scans from *OCTA-500* with the scans of the *ROSE* dataset [32] to obtain around 40,000 patches for training our self-supervised baseline. This combination is necessary for an unlabeled dataset large enough for self-supervised pretraining.

8.5. Synthetic MRI

The Synthetic MRI dataset [46] is a synthetic 3D dataset that simulates the characteristics of clinical vessel datasets. The original dataset provides ground truth labels for vessel segmentation, centerlines, and bifurcation points. The ground truth graphs are obtained with the method described by Drees et al. [14]. We cut the volumes and their graphs in overlapping patches of $64 \times 64 \times 64$ voxels. We use the same dataset with all 80,000 patches for our self-supervised pretraining.

³GitHub repository will be made publicly available upon acceptance

Table 11. Model details for each experiment. The hyperparameters are the same for each model of the respective experiment. The latent space resolution is controlled via the CNN backbone’s stride. It determines the feature size between the backbone and the transformer.

Target Set	Backbone		Latent Space	Transformer				FFN
	Type	Hid. Dim.	Resolution	Hid. Dim.	# Lay.	# Obj Token	# RLN Token	Hid. Dim.
Agadez	ResNet101	512	Multi-Level	512	3	80	2	1024
Munich	ResNet101	512	Multi-Level	512	3	80	2	1024
Synthetic OCTA	ResNet101	512	Multi-Level	512	3	80	5	1024
OCTA-500	ResNet101	512	Multi-Level	512	3	80	2	1024
Synthetic MRI	SeresNet	256	$2 \times 2 \times 2$	552	4	120	2	1280
Whole Brain Vessels	SeresNet	256	$2 \times 2 \times 2$	552	4	120	2	1280

Table 12. Training details and hyperparameters for each trained model.

Experiment	Batch Size	Epochs	Learning Rate		Loss Coeff.				
			Backbone	Transformer	λ_{gIoU}	λ_{cls}	λ_{reg}	λ_{Resln}	λ_{reg}
A	32	100	0.00002	0.0002	2	3	5	5	1.0
B	32	100	0.00002	0.0002	2	3	5	5	1.0
C	32	100	0.00002	0.0002	2	3	5	5	1.0
D	32	100	0.00007	0.00007	3	4	2	6	0.8
E	32	100	0.00007	0.00007	3	4	2	6	0.8

8.6. Whole Brain Vessels

The Whole Brain Vessel dataset [37] is a publicly available open graph benchmark dataset for link prediction (ogbl-vessel⁴). It consists of a graph representing the entirety of the mouse brain’s vascular structure down to the capillary level. Todorov et al. [47] obtained the raw vessel scans using tissue-clearing methods and fluorescent microscopy and then segmented the brain vasculature using CNNs. The dataset has an image, segmentation, and graph representation. We create overlapping patches with a spatial size of $50 \times 50 \times 50$ voxels and remove artifactual patches (e.g., patches containing only noise). We extract 43,500 image patches from an unlabeled whole-brain mouse scan obtained with the vDISCO pipeline [16] for training our self-supervised model.

9. Evaluation metrics

We choose to evaluate our models’ performance using three different evaluation metric types: 1) topological metrics, 2) graph distance metrics, and 3) object detection metrics.

Topological metrics The TOPO-score [6] samples multiple sub-graphs starting from different seed locations from the ground truth and measures its similarity to the inferred graph from the predicted graph with the same seed location. The similarity is measured by matching a fixed amount of points between the two graphs. Two points from two graphs are matched if the distance between their spatial coordinates is below a threshold. The result

of this matching across all sampled subgraphs is used for calculating precision and recall. This method accurately quantifies a prediction’s geometrical (i.e., the roads’ geographical position) and topological (i.e., the roads’ interconnections) quality. We use the implementation and parameters from Biagioni et al. [6]. These metrics are not implemented in 3D.

Graph distance metrics The street mover distance (SMD) approximates the Wasserstein distance between a fixed number of uniformly sampled points along the ground truth graph and the predicted graph. Intuitively, it represents the minimal distance by which the predicted graph must be moved to match the ground truth [4].

Object detection metrics Further, we resort to widely-used object detection metrics: mean average precision (mAP) and mean average recall (mAR) [36]. To calculate each detection’s intersection over union (IoU), we create a hypothetical bounding box of fixed size around each node. Similarly, we create bounding boxes around the edges with a minimum spatial size m in all dimensions. This minimum holds for edges that connect two nodes a and b where the difference between the coordinates in one dimension is lower than m (e.g. if $|a_x - b_x| < m$). We calculate the mean AP and AR between the values of different IoU thresholds (i.e., 0.5 and 0.95).

10. Additional quantitative results

In Table 14, we present our main results from Table 1 in addition to the results’ standard deviation across five mutually exclusive folds of the test set.

⁴<https://ogb.stanford.edu/docs/linkprop/#ogbl-vessel>

Table 13. Overview of the used datasets, selected characteristics, and respective training, validation, and test set sizes.

Dataset	Road Description				Vessel Description		Train	Split	
	Street Type	Vegetation	Layout	Continent	Dimension	Spatial Size		Val	Test
20 U.S. Cities [22]	Sealed	Rich	Grid-plan	N. America	2D	128×128	99.2k	24.8k	25k
Global Diverse Cities									
Agadez	Unsealed	Arid	Grid-plan	Africa	2D	128×128	480	120	290
Munich	Sealed	Rich	Historical	Europe	2D	128×128	440	110	220
Synth. OCTA [35]	-	-	-	-	2D	128×128	480	120	2k
OCTA-500 [26]	-	-	-	-	2D	128×128	1.6k	400	2.2k
Synth. MRI [40]	-	-	-	-	3D	64×64×64	4k	1k	5k
Microscopy [47]	-	-	-	-	3D	50×50×50	4k	1k	1.2k
Unlabeled datasets									
20 U.S. Cities	-	-	-	-	2D	128×128	124k	-	-
Synth. OCTA [35]	-	-	-	-	2D	128×128	96.4k	-	-
Real OCTA [26, 32]	-	-	-	-	2D	128×128	40k	-	-
Synthetic MRI [40]	-	-	-	-	3D	64×64×64	80k	-	-
Microscopy [16]	-	-	-	-	3D	50×50×50	43.5k	-	-

11. Additional qualitative results

We are providing additional qualitative results in the form of multiple figures; please see Figure 7 - 11.

Table 14. Main results with standard deviations. Quantitative Results for our cross-dimensional image-to-graph transfer learning framework. All models are pretrained on the U.S cities road dataset. We outperform the baselines across all datasets. We present the standard deviations in addition to the main results.

Fine Tuning Training Set	(Pre-)Training Strategy	Node-mAP \uparrow	Node-mAR \uparrow	Edge-mAP \uparrow	Edge-mAR \uparrow	SMD \downarrow	Topo-Prec. \uparrow	Topo-Rec. \uparrow
A) TL from roads (2D) to roads (2D)								
Agadez [20]	No Pretr. [18]	0.067 \pm 0.006	0.122 \pm 0.007	0.021 \pm 0.005	0.043 \pm 0.006	0.062 \pm 0.028	0.369 \pm 0.051	0.261 \pm 0.047
	Self-superv. [9]	0.083 \pm 0.010	0.156 \pm 0.011	0.030 \pm 0.005	0.071 \pm 0.007	0.030 \pm 0.005	0.471 \pm 0.0082	0.459 \pm 0.039
	Supervised	0.161 \pm 0.021	0.237 \pm 0.023	0.115 \pm 0.016	0.177\pm0.017	0.023 \pm 0.009	0.783 \pm 0.018	0.711\pm0.039
	Ours	0.163\pm0.017	0.244\pm0.015	0.116\pm0.019	0.172 \pm 0.021	0.022\pm0.003	0.816\pm0.032	0.614 \pm 0.036
Munich [20]	No Pretr. [18]	0.083 \pm 0.012	0.120 \pm 0.011	0.034 \pm 0.013	0.054 \pm 0.016	0.235 \pm 0.049	0.260 \pm 0.057	0.247 \pm 0.070
	Self-superv. [9]	0.088 \pm 0.021	0.145 \pm 0.033	0.060 \pm 0.015	0.097 \pm 0.023	0.155 \pm 0.032	0.339 \pm 0.035	0.384 \pm 0.075
	Supervised	0.277 \pm 0.022	0.336 \pm 0.025	0.207 \pm 0.027	0.272 \pm 0.031	0.091 \pm 0.038	0.682 \pm 0.037	0.660\pm0.041
	Ours	0.285\pm0.015	0.344\pm0.011	0.224\pm0.030	0.277\pm0.031	0.090\pm0.043	0.726\pm0.078	0.655 \pm 0.070
B) TL from roads (2D) to synthetic retinal vessels (2D)								
Synthetic OCTA [35]	No Pretr. [18]	0.273 \pm 0.003	0.375 \pm 0.003	0.140 \pm 0.002	0.339 \pm 0.003	0.005 \pm 0.002	0.181 \pm 0.004	0.948 \pm 0.004
	Self-superv. [9]	0.136 \pm 0.002	0.260 \pm 0.003	0.069 \pm 0.002	0.223 \pm 0.004	0.031 \pm 0.006	0.093 \pm 0.005	0.927 \pm 0.010
	Supervised	0.291 \pm 0.003	0.384 \pm 0.003	0.170 \pm 0.002	0.338 \pm 0.005	0.004 \pm 0.001	0.211 \pm 0.005	0.957\pm0.007
	Ours	0.415\pm0.005	0.493\pm0.003	0.250\pm0.004	0.415\pm0.004	0.002\pm0.001	0.401\pm0.003	0.890 \pm 0.007
C) TL from roads (2D) to real retinal vessels (2D)								
OCTA-500 [26]	No Pretr. [18]	0.189 \pm 0.005	0.282 \pm 0.007	0.108 \pm 0.004	0.169 \pm 0.006	0.017 \pm 0.002	0.737 \pm 0.007	0.634 \pm 0.010
	Self-superv. [9]	0.214 \pm 0.004	0.305 \pm 0.004	0.135 \pm 0.001	0.213 \pm 0.002	0.016 \pm 0.002	0.763 \pm 0.012	0.706 \pm 0.005
	Supervised	0.366 \pm 0.004	0.447 \pm 0.004	0.276 \pm 0.006	0.354 \pm 0.007	0.014 \pm 0.001	0.862 \pm 0.010	0.775 \pm 0.011
	Ours	0.491\pm0.006	0.571\pm0.005	0.366\pm0.009	0.489\pm0.007	0.012\pm0.002	0.877\pm0.004	0.817\pm0.011
D) TL from roads (2D) to brain vessels (3D)								
Synthetic MRI [40]	No Pretr. [18]	0.162 \pm 0.003	0.250 \pm 0.003	0.125 \pm 0.004	0.201 \pm 0.004	0.013\pm0.000	-	-
	Self-superv. [9]	0.162 \pm 0.003	0.252 \pm 0.003	0.120 \pm 0.004	0.193 \pm 0.004	0.014 \pm 0.000	-	-
	Supervised	*	*	*	*	*	*	*
	Ours	0.356\pm0.003	0.450\pm0.002	0.221\pm0.003	0.322\pm0.003	0.013\pm0.000	-	-
E) TL from roads (2D) to real whole-brain vessel data (3D)								
Microscopic images [47]	No Pretr. [18]	0.231 \pm 0.016	0.308 \pm 0.021	0.249 \pm 0.017	0.329 \pm 0.023	0.017\pm0.000	-	-
	Self-superv. [9]	0.344 \pm 0.026	0.404 \pm 0.029	0.363 \pm 0.026	0.425 \pm 0.030	0.017\pm0.000	-	-
	Supervised	*	*	*	*	*	*	*
	Ours	0.483\pm0.037	0.535\pm0.039	0.523\pm0.041	0.566\pm0.043	0.017\pm0.000	-	-

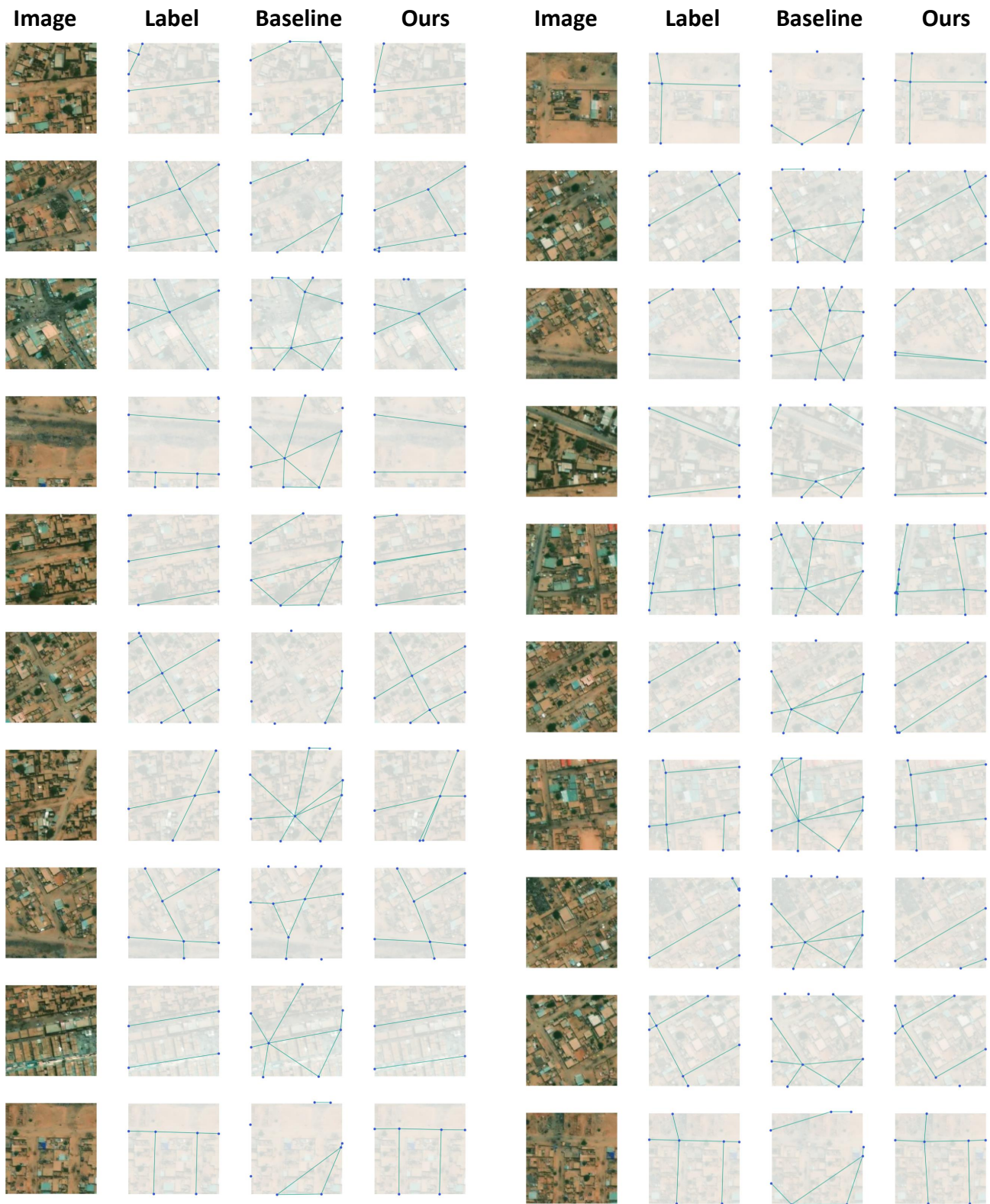


Figure 7. Qualitative results for the Agadez dataset. Two columns, from left to right: Image, ground truth graph, baseline, and our method. Our method consistently outperforms the baselines, which overpredict the edges and nodes for road data.

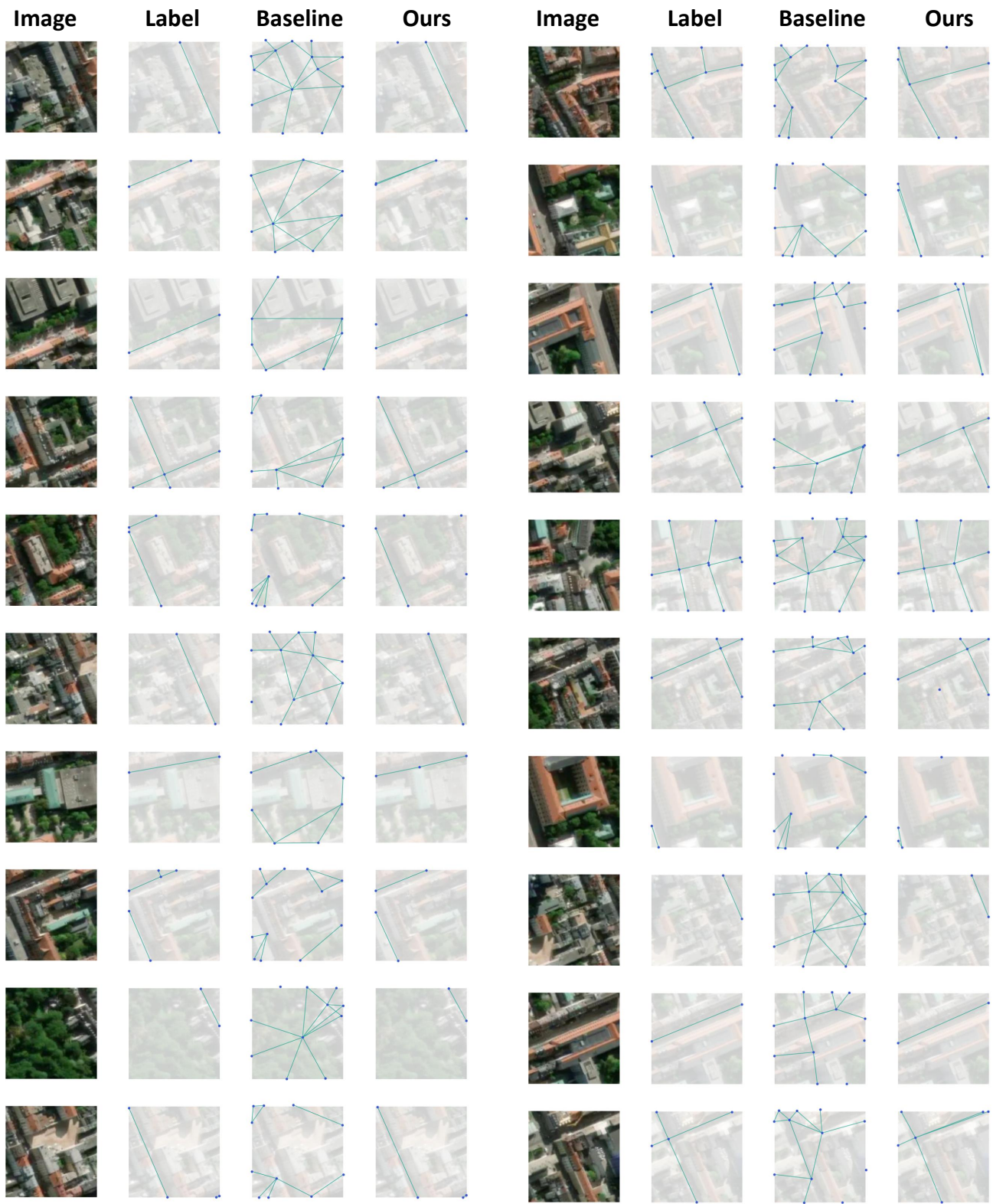


Figure 8. Qualitative results for the Munich dataset. Two columns, from left to right: Image, ground truth graph, baseline, and our method. Our method consistently outperforms the baselines, which overpredict the edges and nodes for road data.

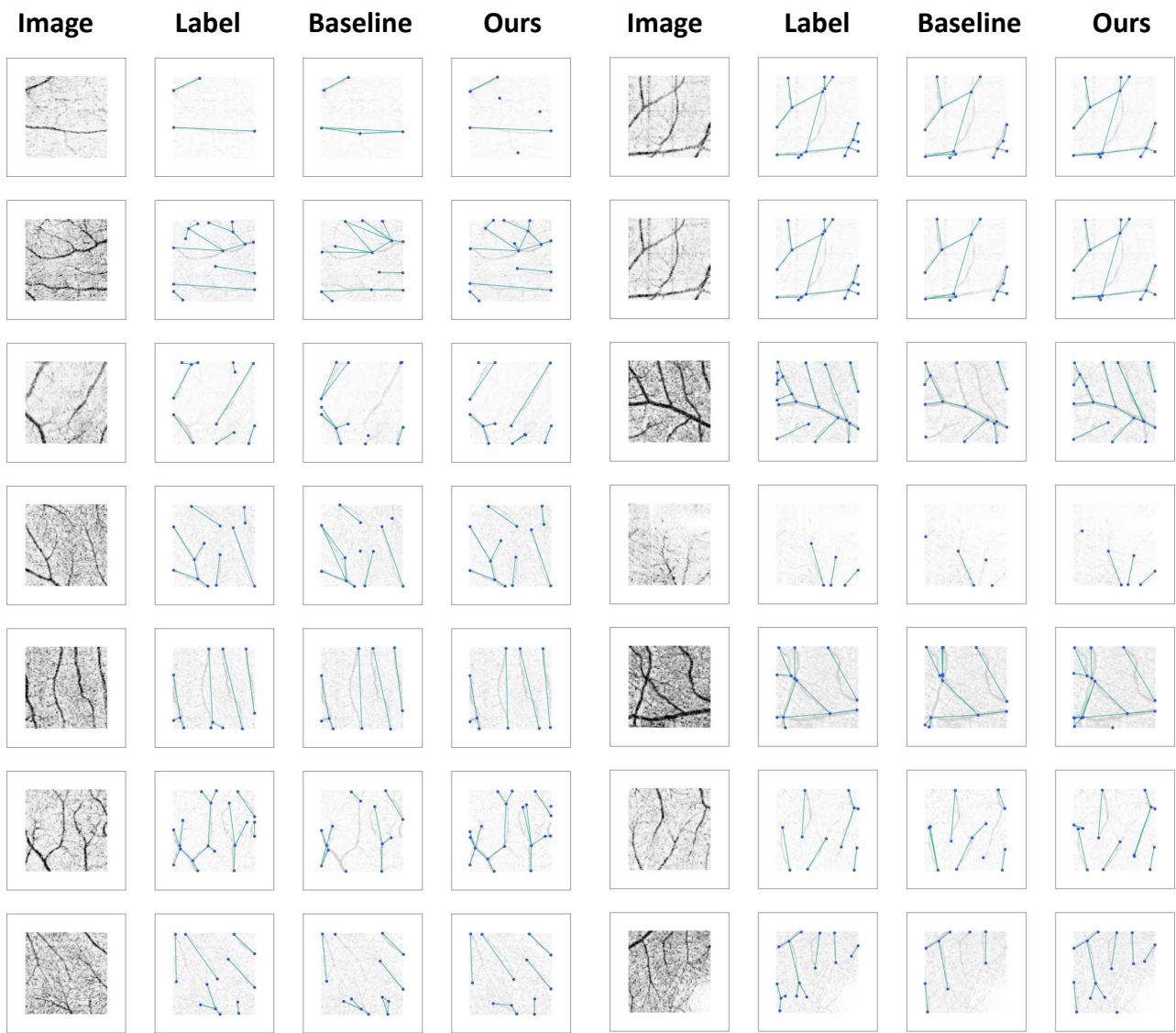


Figure 9. Qualitative results for the OCTA-500 dataset. Two columns, from left to right: Image, ground truth graph, baseline, and our method. Our method consistently outperforms the baselines, which underpredict the edges and nodes for the vessel data. It is important to note that the OCTA-500 dataset labels are on the large vessels. The graph annotations are not provided for all capillaries and are therefore not learned by the models either.

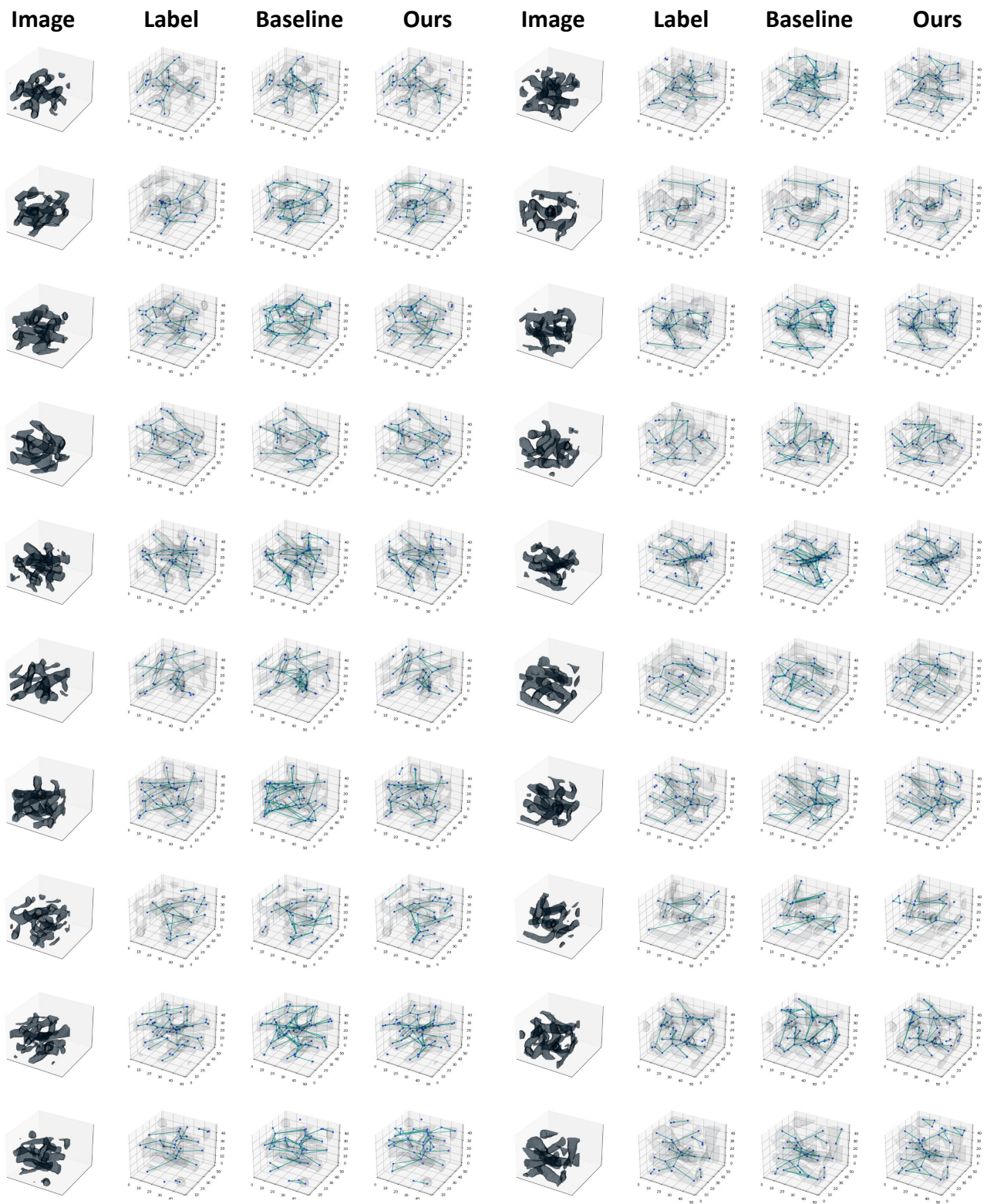


Figure 10. Qualitative results for the 3D whole brain vessel dataset. Two columns, from left to right: Image, ground truth graph, baseline, and our method. Our method consistently outperforms the baselines, which overpredict the edges for the 3D vessel data. Furthermore, the baseline often predicts implausible triangles between three nodes.



Figure 11. Qualitative results for the synthetic 3D vessel MRI dataset. Two columns, from left to right: Image, ground truth graph, baseline, and our method. Our method consistently outperforms the baselines, which overpredict the nodes for the 3D vessel data and underpredict edges.