SideInfNet: A Deep Neural Network for Semi-Automatic Semantic Segmentation with Side Information

Jing Yu Koh[1⋆], Duc Thanh Nguyen[2], Quang-Trung Truong[1], Sai-Kit Yeung[3], and Alexander Binder[1]

[1] Singapore University of Technology and Design, Singapore
[2] Deakin University, Australia
[3] Hong Kong University of Science and Technology, Hong Kong

Abstract. Fully-automatic execution is the ultimate goal for many Computer Vision applications. However, this objective is not always realistic in tasks associated with high failure costs, such as medical applications. For these tasks, semi-automatic methods allowing minimal effort from users to guide computer algorithms are often preferred due to desirable accuracy and performance. Inspired by the practicality and applicability of the semi-automatic approach, this paper proposes a novel deep neural network architecture, namely SideInfNet that effectively integrates features learnt from images with side information extracted from user annotations. To evaluate our method, we applied the proposed network to three semantic segmentation tasks and conducted extensive experiments on benchmark datasets. Experimental results and comparison with prior work have verified the superiority of our model, suggesting the generality and effectiveness of the model in semi-automatic semantic segmentation.

Keywords: semi-automatic semantic segmentation, side information

1 Introduction

Most studies in Computer Vision tackle fully-automatic inference tasks which, ideally, perform automatically without human intervention. To achieve this, machine learning models are often well trained on rich datasets. However, these models may still fail in reality when dealing with unseen samples. A possible solution for this challenge is using assistive information provided by users, e.g., user-provided brush strokes and bounding boxes [16]. Human input is also critical for tasks with high costs of failure. Examples include medical applications where predictions generated by computer algorithms have to be verified by human experts before they can be used in treatment plans. In such cases, a semi-automatic approach that allows incorporation of easy-and-fast side information provided from human annotations may prove more reliable and preferable.

⋆ Currently an AI Resident at Google.

Semantic segmentation is an important Computer Vision problem aiming to associate each pixel in an image with a semantic class label. Recent semantic segmentation methods have been built upon deep neural networks [11,8,4,5]. However, these methods are not flexible to be extended with additional information from various sources, such as human annotations or multi-modal data. In addition, human interactions are not allowed seamlessly and conveniently.

In this paper, we propose SideInfNet, a general model that is capable of integrating domain knowledge learnt from domain data (e.g., images) with side information from user annotations or other modalities in an end-to-end fashion. SideInfNet is built upon a combination of advanced deep learning techniques. In particular, the backbone of SideInfNet is constructed from state-of-the-art convolutional neural network (CNN) based semantic segmentation models. To effectively calibrate the dense domain-dependent information against the spatially sparse side information, fractionally strided convolutions are added to the model. To speed up the inference process and reduce the computational cost while maintaining the quality of segmentation, adaptive inference gates are proposed to make the network's topology flexible and optimal. To the best of our knowledge, this combination presents a novel architecture for semi-automatic segmentation.

A key challenge in designing such a model is in making it generalize to different sparsity and modalities of side information. Existing work focuses on sparse pixel-wise side information, such as user-defined keypoints [19,13], and geotagged photos [22,7]. However, these methods may not perform optimally when the side information is non-uniformly distributed and/or poorly provided, e.g., brush strokes which can be drawn dense and intertwined. In [22], street-level panorama information is used as a source of side information. However, such knowledge is not available in tasks other than remote sensing, e.g., in tasks where the side information is provided as brush strokes. Furthermore, expensive nearest neighbor search is used for the kernel regression in [22], which we replace by efficient trainable fractionally strided convolutions. The Higher-Order Markov Random Field model proposed in [7] can be adapted to various side information types but is not end-to-end trainable. Compared with these works, SideInfNet provides superior performance in various tasks and on different datasets. Importantly, our model provides a principled compromise between fully-automatic and manual segmentation. The benefit gained by the model is well shown in tasks where there exists a mismatch between training and test distribution. A few brush strokes can drastically improve the performance on these tasks. We show the versatility of our proposed model on three tasks:

- **Zone segmentation** of low-resolution satellite imagery [7]. Geotagged street-level photographs from social media are used as side information.
- **BreAst Cancer Histology (BACH) segmentation** [2]. Whole-slide images are augmented with expert-created brush strokes to segment the slides into *normal*, *benign*, *in situ carcinoma* and *invasive carcinoma* regions.
- **Urban segmentation** of very high-resolution (VHR) overhead crops taken of the city of Zurich [21]. Brush annotations indicate geographic features and

are augmented with imagery features to identify eight different urban and peri-urban classes from the Zurich Summer dataset [21].

2 Related Work

2.1 Interactive Segmentation

GrabCut [16] is a seminal work of interactive segmentation that operates in an unsupervised manner. The method allows users to provide interactions in the form of brush strokes and bounding boxes demarcating objects. Several methods have extended the GrabCut framework for both semantic segmentation and instance segmentation, e.g., [9,23]. However, these methods only support bounding box annotations and thus cannot be used in datasets containing irregular object shapes, e.g., non-rectangular zones in the Zurich Summer dataset [21].

Users can also provide prior and reliable cues to guide the segmentation process on-the-fly [14,3,17,10]. For instance, Perazzi et al. [14] proposed a CNN-based guidance method for segmenting user-defined objects from video data. In this work, users provide object bounding boxes or regions. It is also shown that increasing the number of user annotations led to improved segmentation quality. In a similar manner, Nagaraja et al. [17] tackled the task of object segmentation from video by combining motion cues and user annotations. In their work, users make scribbles to delineate the objects of interests. Experimental results verified the cooperation of sparse user annotations and motion cues, filling the gap between fully automatic and manual object segmentation. However, in the above methods, user annotations play a role as auxiliary cues but are not effectively incorporated (as features) into the segmentation process.

2.2 Semantic Segmentation with Side Information

One form of side information used in several segmentation problems is keypoint annotations. The effectiveness of oracle keypoints in human segmentation is illustrated in [19]. Similarly, in [13], a method for automatically learning keypoints was proposed. The keypoints are grouped into pose instances and used for instance segmentation of human subjects. The spatial layout of keypoints is important to represent meaningful human structures, but such constraint are not always held for other object types, such as cell masses in histopathology.

Literature has also demonstrated the advantages of using ground-level imagery as side information in remote sensing. For instance, in [12], multi-view imagery data, including aerial and ground images, were fused into a Markov Random Field (MRF) model to enhance the quality of fine-grained road segmentation. In [7], domain-dependent features from satellite images were learnt using CNNs while street-level photos were classified and considered as higher-orders in a Higher-Order MRF model. These methods are flexible to various CNN architectures but are not trainable in an end-to-end fashion.

Workman et al. [22] proposed a model for fusing multi-view imagery data into a deep neural network for estimating geospatial functions land cover and land

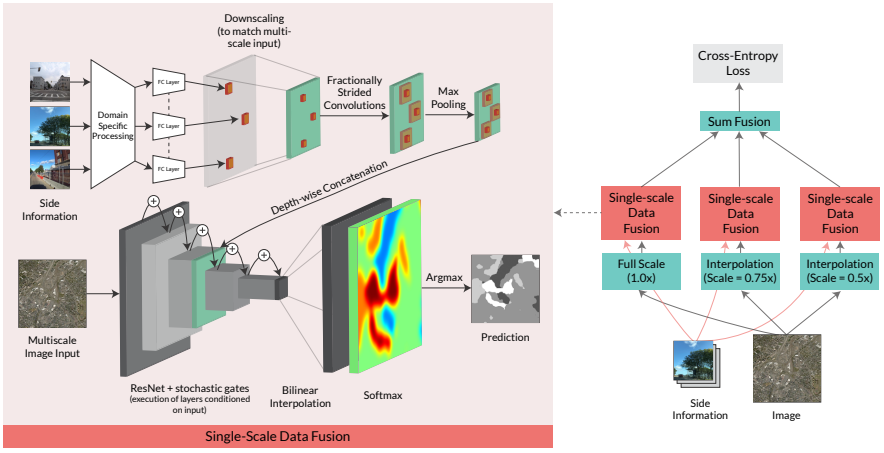

Fig. 1. Our proposed network architecture. A feature map of annotations is constructed based on the task. Our architecture for semantic segmentation is built on top of Deeplab-ResNet [5].

use. While this model is end-to-end, it has heavy computational requirements for its operation, e.g., for calculating and storing k nearest annotations, and thus may not be tractable for tasks with high density annotations. In addition, the model requires panorama knowledge to infer street-view photography.

3 SideInfNet

We propose SideInfNet, a novel neural network that fuses domain knowledge and user-provided side information in an end-to-end trainable architecture. SideInfNet allows the incorporation of multi-modal data, and is flexible with different annotation types and adaptive to various segmentation models. SideInfNet is built upon state-of-the-art semantic segmentation [11,5,15] and recent advances in adaptive neural networks [20,18]. This combination makes our model optimal while maintaining high quality segmentation results. For the sake of ease in presentation, we describe our method in the view of zone segmentation, a case study. However, our method is general and can be applied in different scenarios.

Zone segmentation aims to provide a zoning map for an aerial image, i.e., to identify the zone type for every pixel on the aerial image. Side information in this case includes street-level photos. These photos are captured by users and associated with geocodes that refer to their locations on the aerial image. Domain-dependent features are extracted from the input aerial image using some CNN-based semantic segmentation model (see Section 3.1). Side information features are then constructed from user-provided street-level photos (see Section 3.2 and Section 3.3). Associated geocodes in the street-level photos help to identify their locations in the receptive fields in the SideInfNet architecture where both

domain-dependent and side information features are fused. To reduce the computational cost of the model while not sacrificing the quality of segmentation, adaptive inference gates are proposed to skip layers conditioned on input (see Section 3.4). Fig. 1 illustrates the workflow of SideInfNet whose components are described in detail in the following subsections.

3.1 CNN-based Semantic Segmentation

To extract domain-dependent features, we adopt the Deeplab-ResNet [5], a state-of-the-art CNN-based semantic segmentation. Deeplab-ResNet makes use of a series of dilated convolutional layers, with increasing rates to aggregate multi-scale features. To adapt Deeplab-ResNet into our framework, we retain the same architecture but extend the *conv2_3* layer with side information (see Section 3.2).

Specifically, the side information feature map is concatenated to the output of the *conv2_3* layer (see Fig. 1). As the original *conv2_3* layer outputs a feature map with 256 channels, concatenating the side information feature map results in a $\frac{H}{4} \times \frac{W}{4} \times (256 + d)$ dimensional feature map where H and W are the height and width of the input image, and d is the number of channels of the side information feature map. This extended feature map is the input to the next convolutional layer, *conv3_1*. We provide an ablation study on varying the dimension d in our supplementary material.

3.2 Side Information Feature Map Construction

Depending on applications, domain specific preprocessing may need to be applied to the side information. For instance, in the zone segmentation problem, we use the Places365-CNN in [24] to create vector representations for street-level photos (see details in Section 4.1). These vectors are then passed through a fully-connected layer returning d-dimensional vectors. Suppose that the input aerial image is of size $H \times W$. A side information feature map $\boldsymbol{x}^l$ of size $H \times W \times d$ can be created by initializing the d-dimensional vector at every location in $H \times W$ with the feature vector of the corresponding street-level photo, if one exists there. The feature vectors at locations that are not associated to any street-level photos are padded with zeros. Mapping image locations to street-level photos can be done using the associated geocodes of the street-level photos. Nearest neighbor interpolation is applied on the side information feature map to create multi-scale features. Features that fall in the same image locations (on the aerial image) due to downscaling are averaged. To make feature vectors consistent across scales and data samples, all feature vectors are normalized to the unit length.

There may exist misalignment in associating the side information features with their corresponding locations on the side information feature map. For instance, a brush stroke provided by a user may not well align with a true region. In the application of zoning, a street-level photo may not record the scene at the exact location where the photo is captured. Therefore, a direct reference of a street-level photo to a location on the feature map via the photo's geocode

may not be a perfect association. However, one could expect that the side information could be propagated from nearby locations. To address this issue, we apply a series of fractionally-strided convolutions to the normalized feature map $\boldsymbol{x}^l$ to distribute the side information spatially. In our implementation, we use 3×3 kernels of ones, with stride length of 1 and padding of 1. After a single fractionally-strided convolution, side information features are distributed onto neighbouring 3×3 regions. We repeat this operation (denoted as f_c) n times and sum up all the feature maps to create the features for the next layer as follows,

$$\boldsymbol{x}^{l+1} = F(\boldsymbol{x}^l) = \sum_{i=1}^{n} w_i f_c^i(\boldsymbol{x}^l) \tag{1}$$

where w_i are learnable parameters and f_c^i is the i-th functional power of f_c, i.e.,

$$f_c^i(\boldsymbol{x}^l) = \begin{cases} f_c(\boldsymbol{x}^l), & i = 1 \\ f_c(f_c^{i-1}(\boldsymbol{x}^l)), & \text{otherwise} \end{cases} \tag{2}$$

The parameters w_i in (1) allow our model to learn the importance of spatial extent. We observe a decreasing pattern in w_i (i.e., $w_1 > w_2 > \cdots$) after training. This matches our intuition that information is likely to become less relevant with increased distances. The resulting feature map $\boldsymbol{x}^{l+1}$ represents a weighted sum of nearby feature vectors. We also normalize the feature vector at each location in the feature map by the number of the fractionally-strided convolutions used at that location. This has the effect of averaging overlapping features.

Lastly, we perform maxpooling to further downsample the side information feature map to fit with the counterpart domain-dependent feature map for feature fusion. We choose to perform feature fusion before the second convolutional block of Deeplab-ResNet, with the output of the *conv2_3* layer. We empirically found that this provided a good balance between computational complexity and segmentation quality. The output of the maxpooling layer is concatenated in the channels dimension to the output of the original layer (see Fig. 1). It is important to note that our proposed side information feature map construction method is general and can be applied alongside any CNN-based semantic segmentation architectures.

3.3 Fusion Weight Learning

As defined in (1), the output for each pixel (p, q) in the feature map $f_c^{i+1}(x^l)$ (after applying 3×3 fractionally-strided convolution of 1s) can be described as:

$$f_c^{i+1}(x^l)_{p,q} = \sum_{j=1}^{3} \sum_{k=1}^{3} w_i x^l_{p-2+j,q-2+k}. \tag{3}$$

Gradient of the fusion weight w_i for each layer can be computed as,

$$\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial f_c^{i+1}(x^l)} \frac{\partial f_c^{i+1}(x^l)}{\partial w_i} = \sum_p \sum_q \frac{\partial L}{\partial f_c^{i+1}(x^l)_{p,q}} \sum_{j=1}^{3} \sum_{k=1}^{3} x^l_{p-2+n,q-2+n} \tag{4}$$

where $\frac{\partial L}{\partial f_c^{i+1}(x^l)}$ is back-propagated from the *conv2_3* layer.

For the fully-connected layers used for domain-specific processing (see Fig. 1), the layers are shared for each side-information instance. The shared weights w_{fc} can be learnt through standard back-propagation of a fully-connected layer:

$$\frac{\partial L}{\partial w_{\text{fc}}} = \frac{\partial L}{\partial f_c^1} \frac{\partial f_c^1}{\partial w_{\text{fc}}} \tag{5}$$

where $\frac{\partial L}{\partial f_c^1}$ is back-propagated from the first fusion layer (see (4)).

3.4 Adaptive Architecture

Inspired by advances in adaptive neural networks [20,18], we adopt adaptive inference graphs in SideInfNet. Adaptive inference graphs decide skip-connections in the network architecture using adaptive gates $\boldsymbol{z}^l$. Specifically, we define,

$$\boldsymbol{x}^{l+1} = \boldsymbol{x}^l + \boldsymbol{z}^l(h(\boldsymbol{x}^l)) \cdot F(\boldsymbol{x}^l) \tag{6}$$

where $\boldsymbol{z}^l(h(\boldsymbol{x}^l)) \in \{0, 1\}$ and h is some function that maps $\boldsymbol{x}^l \in H \times W \times d$ into a lower-dimensional space of $1 \times 1 \times d$. The gate $\boldsymbol{z}^l$ is conditioned on $\boldsymbol{x}^l$ and takes a binary decision (1 for "on" and 0 for "off").

Like [20], we set the early layers and the final classification layer of our model to always be executed, as these layers are critical for maintaining the accuracy. The gates are included in every other layer. We define the function h as,

$$h(\boldsymbol{x}^l) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \boldsymbol{x}_{i,j}^l \tag{7}$$

The feature map $h(\boldsymbol{x}^l)$ is passed into a multi-layer perceptron (MLP), which computes a relevance score to determine whether the layer l is executed. We also use a gate target rate t, that determines what fraction of layers should be activated. This is implemented as a mean squared error (MSE) loss and jointly optimized with the cross entropy loss. Each separate MLP determines whether its corresponding layer should be executed (contributing 1 to the total count), or not (contributing 0). Thus, the MSE loss encourages the overall learnt execution rate to be close to t. This is dynamic, i.e., more important layers would be executed more frequently and vice versa. For instance, a target rate $t = 0.8$ imposes a penalty on the loss function when the proportion of layers executed is greater or less than 80%. Our experimental results on this adaptive model are presented in Section 4, where we find that allowing a proportion of layers to be skipped helps improve segmentation quality.

4 Experiments and Results

In this section, we extensively evaluate our proposed SideInfNet in three different case studies. In each case study, we compare our method with its baseline and other existing works. We also evaluate our method under various levels of side information usage and with another CNN backbone.

Table 2. Segmentation performance on zoning. Best performances are highlighted.

Approach	Accuracy				mIOU			
	BOS	**NYC**	**SFO**	**Mean**	**BOS**	**NYC**	**SFO**	**Mean**
Deeplab-ResNet [5]	60.79%	59.58%	72.21%	64.19%	28.85%	23.77%	38.40%	30.34%
HO-MRF* [7]	59.52%	72.25%	73.93%	68.57%	31.92%	34.99%	46.53%	37.81%
Unified* [22]	67.91%	70.92%	75.92%	71.58%	40.51%	39.27%	55.36%	45.05%
SideInfNet	71.33%	71.08%	79.59%	74.00%	41.96%	39.59%	60.31%	47.29%

* Our implementation.

4.1 Zone Segmentation

Experimental Setup Like [7], we conducted experiments on three US cities: Boston (BOS), New York City (NYC), and San Francisco (SFO). Freely available satellite images hosted on Microsoft Bing Maps [6] were used. Ground-truth maps were retrieved at a service level of 12, which corresponds to a resolution of 38.2185 meters per pixel. An example of the satellite imagery is shown in Fig. 2. We retrieved street-level photos from Mapillary [1], a service for sharing crowdsourced geotagged photos. There were four zone types: *Residential*, *Commercial*, *Industrial* and *Others*. Table 1 summarizes the dataset used in this case study.

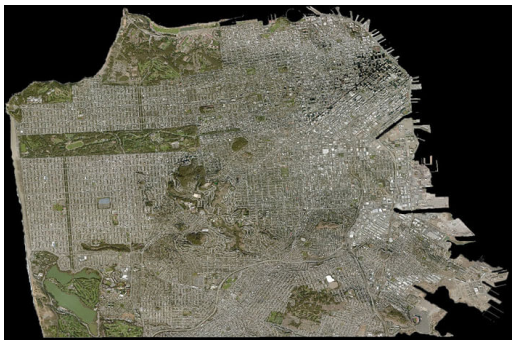

Table 1. Proportion of street-level photos (#photos).

Zone Type	City		
	BOS	**NYC**	**SFO**
Residential	25,607	16,395	50,116
Commercial	13,412	5,556	19,641
Industrial	2,876	9,327	15,219
Others	25,402	15,281	50,214

Fig. 2. Satellite image of San Francisco.

To extract side information features, we utilized the pre-trained model of Places365-CNN [24], which was designed for scene recognition. We fine-tuned the model on our data. During training the model, we froze the weights of the Places365-CNN and used this fine-tuned model to generate side information feature maps. We also applied a series of $n = 5$ fractionally-strided convolutions on feature maps generated from Places365-CNN. This acts as to distribute the side information from each geotagged photo 5 pixels in each cardinal direction.

Results We evaluate our method and compare it with two recent works: Higher-Order Markov Random Field (HO-MRF) [7] and Unified model [22] using 3-fold

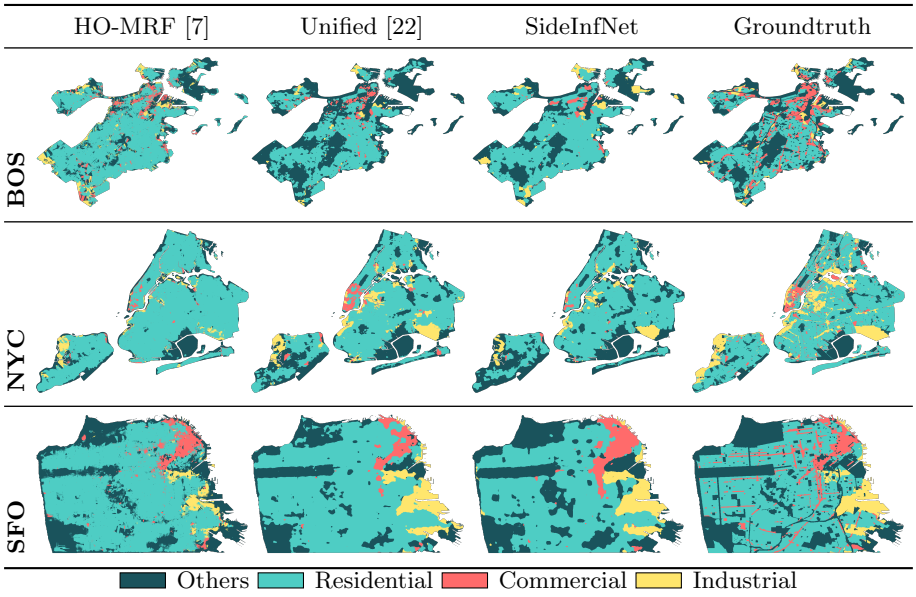

Fig. 3. Comparison of our method and previous works. Best viewed in color.

cross validation, i.e., two cities are used for training and the other one is used for testing. To have a fair comparison, the same Places365-CNN model is used to extract side information in all methods. We also compare our method against the baseline Deeplab-ResNet, which directly performs semantic segmentation of satellite imagery without the use of geotagged photos.

Our results on both pixel accuracy and mean intersection over union (mIOU) are reported in Table 2. As shown in the table, our method significantly improves over its baseline, Deeplab-ResNet, proving the importance of side information. SideInfNet also outperforms prior work, with a relative improvement in pixel accuracy from the Unified model by 3.38% and from the HO-MRF by 7.92%. Improvement on mIOU scores is also significant, e.g., by 4.97% relative to the Unified model, and 25.07% relative to the HO-MRF model.

In addition to improved accuracy, our method offers several advantages over the previous works. First, compared with the HO-MRF [7], our method is trained end-to-end, allowing it to jointly learn optimal parameters for both semantic segmentation and side information feature extraction. Second, our method is efficient in computation. It simply performs a single forward pass through the network to produce segmentation results, opposed to iterative inference in the HO-MRF. Third, by using fractionally-strided convolutions, the complexity of our method is invariant to the side information density. This allows optimal performance on regions with high density of side information. In contrast, the Unified model [22] requires exhaustive searches to determine nearest street-level

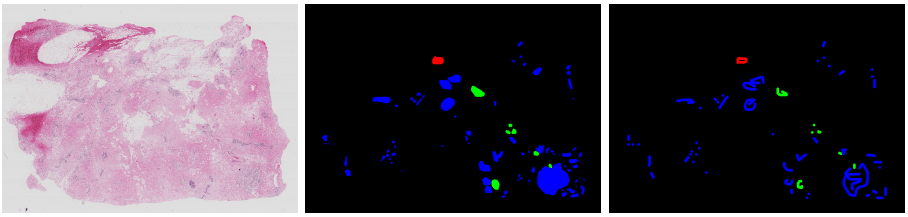

Fig. 4. Left: Whole-slide image. Middle: True labels from the ground-truth. Right: Simulated brush strokes. Best viewed in color.

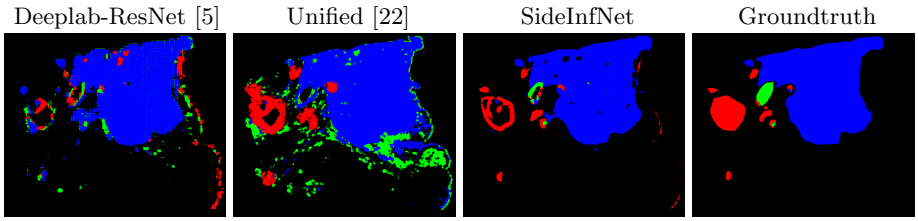

Fig. 5. Comparison of our method and other works on image A05 in the BACH dataset [2]. Best viewed in color.

photos for every pixel on satellite image and thus depend on the density of the street-level photos and the size of the satellite image.

We qualitatively show the segmentation results of our method and other works in Fig. 3. A clear drawback of the HO-MRF is that the results tend to be grainy, likely due to the sparsity of street-level imagery. In contrast, our method generally provides smoother results that form contiguous regions. Moreover, our method better captures fine grained details from street-level imagery.

4.2 BreAst Cancer Histology Segmentation

Experimental Setup BACH (BreAst Cancer Histology) [2] is a dataset for breast cancer histology microscopy segmentation[4]. This dataset consists of high resolution whole-slide images that contain an entire sampled tissue. The whole-slide images were annotated by two medical experts, and images with disagreements were discarded. There are four classes: *normal*, *benign*, *in situ carcinoma* and *invasive carcinoma*. An example of a whole-slide image and its labels is shown in Fig. 4. As the *normal* class is considered background, it is not evaluated. Side information for BACH consists of expert brush stroke annotations,

[4] Data can be found at `https://iciar2018-challenge.grand-challenge.org/`. Due to the unavailability of the actual test set, we used slides A05 and A10 for testing, slide A02 for validation, and all other slides for training. This provides a fair class distribution, as not all slides contained all semantic classes.

indicating the potential presence of each class. In this case study, we use four different brush stroke colors to annotate the four classes.

BACH dataset does not include actual expert-annotated brush strokes. Therefore, to evaluate our method, we simulated expert annotations by using ground-truth labels in the dataset. Since the ground-truth was created by two experts, our brush strokes can be viewed as simulated rough expert input. To simulate situations where users have limited annotation time, we skipped annotating small regions that are likely to be omitted under time constraints. Fig. 4 shows an example of our simulated brush strokes. In our experiments, we used slides A05 and A10 for testing, slide A02 for validation, and all other slides for training.

Table 3. Segmentation performance (mIOU) on BACH dataset. Best performances are highlighted.

Approach	A05	A10	Mean
Deeplab-ResNet [5]	34.08%	21.64%	27.86%
GrabCut [16]	30.20%	25.21%	27.70%
Unified* [22]	41.50%	17.23%	29.37%
SideInfNet	<u>59.03%</u>	<u>35.45%</u>	<u>47.24%</u>

* Our implementation.

Table 4. Segmentation performance on Zurich Summer dataset. Best performances are highlighted.

Approach	Accuracy	mIOU
Deeplab-ResNet [5]	73.20%	42.95%
GrabCut [16]	60.53%	26.89%
Unified* [22]	68.20%	42.09%
SideInfNet	<u>78.97%</u>	<u>58.31%</u>

* Our implementation.

Results We evaluate three different methods: our proposed SideInfNet, Unified model [22], and GrabCut [16]. We were unable to run the HO-MRF model [7] on the BACH dataset due to the large size of the whole-slide images (note that the HO-MRF makes use of fully-connected MRF and thus is not computationally feasible under this context). In addition, since GrabCut is a binary segmentation method, to adapt this work to our case study, we ran the GrabCut model independently for each class. We report the performance of all the methods in Table 3. We also provide some qualitative results in Fig. 5.

Experimental results show that our method greatly outperforms previous works on BACH dataset. Furthermore, the Unified model [22] even performs worse than the baseline Deeplab-ResNet that used only whole-slide imagery. This suggests the limitation of the Unified model [22] in learning from dense annotations. Table 3 also confirms the role played by the side information (i.e., the Deeplab-ResNet vs SideInfNet). This aligns with our intuition, as we would expect that brush strokes provide stronger cues to guide the segmentation.

4.3 Urban Segmentation

Experimental Setup The Zurich Summer v1.0 dataset [21] includes 20 very high resolution (VHR) overview crops taken from the city of Zurich, pansharpened to a PAN resolution of about 0.62 centimeters ground sampling distance (GSD). This is a much higher resolution compared to the low-resolution satellite

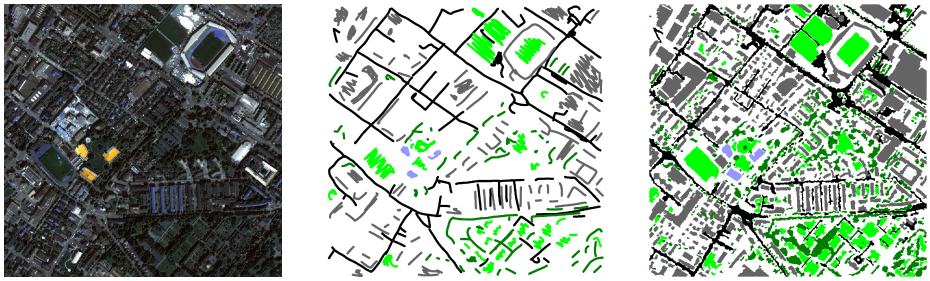

Fig. 6. Example satellite image, brush annotations, and ground-truth map from the Zurich Summer dataset [21]. Best viewed in color.

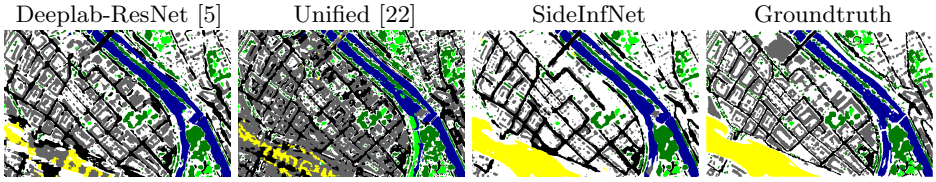

Fig. 7. Qualitative comparison of our method and other works on the Zurich Summer dataset [21]. Best viewed in color.

imagery used in the zoning dataset. The Zurich Summer dataset contains eight different urban and periurban classes: Roads, Buildings, Trees, Grass, Bare Soil, Water, Railways and Swimming pools. Examples of satellite imagery, ground-truth labels, and brush annotations are shown in Fig. 6. Preprocessing steps and feature map construction are performed similarly to that of BACH. We also used rough brush strokes demarcating potential urban classes as side information.

Results Our experimental results on the Zurich Summer dataset are summarized in Table 4. In general, similar trends with the BACH dataset are found, and our proposed method outperforms all prior works. Specifically, by using brush strokes, we are able to gain a relative improvement of 7.88% on accuracy and 35.76% on mIOU over the baseline Deeplab-ResNet. The Zurich dataset contains high-resolution satellite imagery, which suggests the usefulness of including brush annotations even with high fidelity image data. SideInfNet also outperforms the Unified model [22] with a relative improvement of 15.79% on accuracy and 38.53% on mIOU. This result proves the robustness of our method in dealing with dense annotations, which challenge the Unified model. GrabCut also under-performs due to its limitations as an unsupervised binary segmentation method. A qualitative comparison of our method with other works is also shown in Fig. 7.

Table 5. Performance of SideInfNet with varying side information.

Side Information Used	mIOU			Mean Accuracy		
	Zoning [7]	BACH [2]	Zurich [21]	Zoning [7]	BACH [2]	Zurich [21]
100%	47.29%	47.24%	58.31%	74.00%	71.99%	78.97%
80%	40.27%	40.53%	52.32%	72.46%	68.60%	77.58%
60%	39.56%	34.16%	52.14%	72.39%	68.56%	76.33%
40%	37.70%	29.56%	49.49%	71.01%	64.87%	75.83%
20%	34.04%	26.15%	47.72%	68.11%	56.86%	74.29%
0%	28.11%	23.86%	45.98%	58.63%	60.48%	73.36%

4.4 Varying Levels of Side Information

In this experiment, we investigate the performance of our method when varying the availability of side information. To simulate various densities of brush strokes for an input image, we sample the original brush strokes (e.g., from 0% to 100% of the total number) and evaluate the segmentation performance of our method accordingly. The brush strokes could be randomly sampled. However, this approach may bias the spatial distribution of the brush strokes. To maintain the spatial distribution of the brush strokes for every sampling case, we perform k-means clustering on the original set of the brush strokes. For instance, if we wish to utilize a percentage p of the total brush strokes, and n brush strokes are present in total, we apply k-means algorithm with $k = \text{ceil}(np)$ on the centers of the brush strokes to spatially cluster the brush strokes into k groups. For each group, we select the brush stroke whose center is closest to the group's centroid. This step results in k brush strokes. We note that a similar procedure can be applied to sample street-level photos for zone segmentation.

We report the quantitative results of our method w.r.t varying side information in Table 5. In general, we observe a decreasing trend over the accuracy and mIOU as the proportion of side information decreases. This supports our hypothesis that side information is a key signal for improving segmentation accuracy. We also observe a trade off between human effort and segmentation accuracy. For instance, on the zone segmentation dataset [7], improvement over the baseline Deeplab-ResNet is achieved with as little as 20% of the original number of geo-tagged photos. This suggests that our proposed method can provide significant performance gains even with minimal human effort.

4.5 SideInfNet with another CNN Backbone

To show the adaptability of SideInfNet, we experimented SideInfNet built with another CNN backbone. In particular, we adopted the VGG-19 as the backbone in our architecture. Note that VGG was also used in the Unified model [22]. To provide a fair comparison, we re-implemented both SideInfNet and Unified model with the same VGG architecture and evaluated both models using the

Table 6. Performance (mIOU) of SideInfNet with VGG.

Model	Zoning [7]	BACH [2]	Zurich [21]
SideInfNet-VGG	46.12%	49.53%	49.73%
Unified [22]	45.05%	29.37%	42.09%

same training/test split. We also utilized the original hyperparameters proposed in [22] in our implementation. We report the results of this experiment in Table 6.

Experimental results show that SideInfNet outperforms the Unified model [22] on all segmentation tasks when the same VGG backbone is used. These results confirm again the advantages of our method in feature construction and fusion.

5 Conclusion

This paper proposes SideInfNet, a novel end-to-end neural network for semi-automatic semantic segmentation with additional side information. Through extensive experiments on various datasets and modalities, we have shown the advantages of our method across a wide range of applications, including but not limited to remote sensing and medical image segmentation. In addition to being general, our method boasts improved accuracy and computational advantages over prior models. Lastly, our architecture is easily adapted to various semantic segmentation models and side information feature extractors.

The method proposed in this paper acts as a compromise between fully-automatic and manual segmentation. This is essential for many applications with high cost of failure, in which fully-automatic methods may not be widely accepted as of yet. Our model works well with dense brush stroke information, providing a quick and intuitive way for human experts to refine the model's outputs. In addition, our model also outperforms prior work on sparse pixel-wise annotations. By including side information to shape predictions, we are able to achieve an effective ensemble of human expertise and machine efficiency, producing both fast and accurate segmentation results.

6 Acknowledgement

– Duc Thanh Nguyen was partially supported by an internal SEBE 2019 RGS grant from Deakin University.
– Sai-Kit Yeung was partially supported by an internal grant from HKUST (R9429) and HKUST-WeBank Joint Lab.
– Alexander Binder was supported by the MoE Tier2 Grant MOE2016-T2-2-154, Tier1 grant TDMD 2016-2, SUTD grant SGPAIRS1811, TL grant RTDST1907012.

References

1. AB, M.: Mapillary (2019), https://www.mapillary.com, Last accessed on 2019-11-01
2. Aresta, G., Araújo, T., Kwok, S., Chennamsetty, S.S., Safwan, M., Alex, V., Marami, B., Prastawa, M., Chan, M., Donovan, M., et al.: Bach: Grand challenge on breast cancer histology images. Medical image analysis (2019)
3. Caelles, S., Maninis, K.K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L.: One-shot video object segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 221–230 (2017)
4. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. arXiv preprint arXiv:1412.7062 (2014)
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2018)
6. Corporation, M.: Bing maps tile system (2019), https://msdn.microsoft.com/en-us/library/bb259689.aspx, Last accessed on 2019-11-01
7. Feng, T., Truong, Q.T., Thanh Nguyen, D., Yu Koh, J., Yu, L.F., Binder, A., Yeung, S.K.: Urban zoning using higher-order markov random fields on multi-view imagery data. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 614–630 (2018)
8. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 580–587 (2014)
9. Göring, C., Fröhlich, B., Denzler, J.: Semantic segmentation using grabcut. In: VISAPP. pp. 597–602 (2012)
10. Li, S., Seybold, B., Vorobyov, A., Fathi, A., Huang, Q., Jay Kuo, C.C.: Instance embedding transfer to unsupervised video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6526–6535 (2018)
11. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
12. Máttyus, G., Wang, S., Fidler, S., Urtasun, R.: Hd maps: Fine-grained road segmentation by parsing ground and aerial images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3611–3619 (2016)
13. Papandreou, G., Zhu, T., Chen, L.C., Gidaris, S., Tompson, J., Murphy, K.: Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 269–286 (2018)
14. Perazzi, F., Khoreva, A., Benenson, R., Schiele, B., Sorkine-Hornung, A.: Learning video object segmentation from static images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2663–2672 (2017)
15. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
16. Rother, C., Kolmogorov, V., Blake, A.: Grabcut: Interactive foreground extraction using iterated graph cuts. In: ACM transactions on graphics (TOG). vol. 23, pp. 309–314. ACM (2004)

17. Shankar Nagaraja, N., Schmidt, F.R., Brox, T.: Video segmentation with just a few strokes. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3235–3243 (2015)
18. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. arXiv preprint arXiv:1701.06538 (2017)
19. Tripathi, S., Collins, M., Brown, M., Belongie, S.: Pose2instance: Harnessing keypoints for person instance segmentation. arXiv preprint arXiv:1704.01152 (2017)
20. Veit, A., Belongie, S.: Convolutional networks with adaptive inference graphs. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 3–18 (2018)
21. Volpi, M., Ferrari, V.: Semantic segmentation of urban scenes by learning local class interactions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 1–9 (2015)
22. Workman, S., Zhai, M., Crandall, D.J., Jacobs, N.: A unified model for near and remote sensing. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2688–2697 (2017)
23. Xu, N., Price, B., Cohen, S., Yang, J., Huang, T.: Deep grabcut for object selection. arXiv preprint arXiv:1707.00243 (2017)
24. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: Advances in neural information processing systems. pp. 487–495 (2014)