A Continual Learning Framework for Uncertainty-Aware Interactive Image Segmentation

Ervine Zheng, Qi Yu *, Rui Li, Pengcheng Shi, Anne Haake
Rochester Institute of Technology
{mxz5733, qi.yu, rxlics, spcast, arhics}@rit.edu

Abstract

Deep learning models have achieved state-of-the-art performance in semantic image segmentation, but the results provided by fully automatic algorithms are not always guaranteed satisfactory to users. Interactive segmentation offers a solution by accepting user annotations on selective areas of the images to refine the segmentation results. However, most existing models only focus on correcting the current image's misclassified pixels, with no knowledge carried over to other images. In this work, we formulate interactive image segmentation as a continual learning problem and propose a framework to effectively learn from user annotations, aiming to improve the segmentation on both the current image and unseen images in future tasks while avoiding deteriorated performance on previously-seen images. It employs a probabilistic mask to control the neural network's kernel activation and extract the most suitable features for segmenting images in each task. We also apply a task-aware embedding to automatically infer the optimal kernel activation for initial segmentation and subsequent refinement. Interactions with users are guided through multi-source uncertainty estimation so that users can focus on the most important areas to minimize the overall manual annotation effort. Experiments are performed on both medical and natural image datasets to illustrate the proposed framework's effectiveness on basic segmentation performance, forward knowledge transfer, and backward knowledge transfer.

Introduction

Deep learning (DL) based methods have been increasingly applied to semantic image segmentation in recent years with superior performance (Long, Shelhamer, and Darrell 2015; Ronneberger, Fischer, and Brox 2015; Chen et al. 2017; He et al. 2017). However, large-scale annotated images are usually required to properly train a complex DL model to ensure a good segmentation performance. Fully annotating an image is a lengthy and laborious process. In many specialized domains (e.g., medicine and biology), image annotation may only be performed by expert users with special expertise. Thus, the availability of large-scale fully annotated images is usually rather limited in these domains, which poses a key challenge of applying state-of-the-art DL models. Furthermore, segmentation results provided by fully automatic algorithms are not always guaranteed satisfactory to users. For example, in medical image analysis, physicians may form different opinions on the boundary of segments. It is vital to provide an effective means for users to adjust the segmentation results. Finally, domain shift may cause severe deterioration of segmentation results if training images vary significantly from the testing ones due to different imaging devices, environments, and other factors, which may frequently occur in many applications.

Interactive segmentation offers a promising direction by leveraging users' knowledge to refine the segmentation results. It allows users to interact with the system by annotating a few pixels, which are then used by the model to update the segmentation results. However, most existing algorithms process images and user annotations as isolated cases. They mainly focus on correcting the current image's misclassified pixels but ignore carrying over knowledge learned from user annotations to other images. In a practical setting, new images are usually provided to a model in sequential order. The model should effectively learn from user annotations to improve segmentation performance and reduce users' (repetitive) annotation effort in the future. Besides, most existing algorithms leave the decision of selective annotation altogether to the users, i.e., users may make edits on any areas without knowing which edit is most useful to the model for improving the results. A fundamental challenge of interactive segmentation is how to provide users informative guidance to improve the segmentation accuracy with minimum annotation effort.

To address the above challenges, we formulate interactive segmentation as a continual learning problem. The segmentation tasks are organized as a sequence. In each task, the model is trained with only one or a few images and along with limited user annotations. This is different from conventional continual learning, where one task usually contains a large batch of data instances, because interactive segmentation aims to leverage limited user annotations to improve performance. We also design a training-testing protocol to evaluate how a model can effectively learn from user annotations to improve the segmentation performance on 1) the current image and 2) the unseen images in future tasks, while mitigating the deteriorated performance on

*Corresponding author.

3) previously-seen images. We refer to the three objectives mentioned above as *basic performance*, *forward knowledge transfer*, and *backward knowledge transfer*.

We propose a **C**ontinual **L**earning framework for **I**nteractive **S**egmentation (referred to as CLIS) to achieve the three objectives simultaneously. It employs a task-specific probabilistic mask, generated from a Bayesian non-parametric Indian Buffet Process (IBP), to control the network's convolutional kernel activation. A different set of kernels are activated for each task, aiming to extract the most suitable features for segmenting the corresponding images. Besides, the IBP prior encourages kernels frequently activated in previous tasks to be more likely activated in the current task, effectively leveraging knowledge learned from the previous tasks. This prior knowledge of the convolutional kernels and activation masks is utilized by the model to generate initial predictions augmented with multi-source uncertainty estimations. If the segmentation deviates from a user's opinion, s/he may choose to annotate on selective areas. The uncertainty estimation guides the user to annotate the most important areas and minimize the overall manual annotation effort. The model will be updated based on the annotations and generate refined segmentation results. At the same time, new knowledge from the user is encoded as the posterior probabilistic distributions for the convolutional kernels and activation masks.

Figure 1 illustrates a typical usage of the proposed framework in one task and demonstrates how user annotations are leveraged to refine the segmentation on a dermatological image. The model consists of two modules: a segmentation module (with uncertainty estimation nested), and an interactive learning module. The segmentation module performs pixel-level classification and infers a binary mask that activates a subset of the convolutional kernels to extract the most suitable features for segmentation in each task. It also performs uncertainty estimation by generating a map that visualizes the model's uncertainty on its prediction. A user may choose to provide additional annotations to refine the segmentation. As shown in Figure 1 (C), the model makes uncertain predictions on the marked areas. Users are guided to edit those areas, and then the interactive learning module propagates from user annotations to other pixels and updates the network. This process may iterate multiple rounds until the user is satisfied with the refined results, and then moves on to the next task.

The major contribution of this paper is three-fold: (i) formulating interactive segmentation as a continual learning problem and designing an evaluation protocol regarding basic performance and forward/backward knowledge transfer, (ii) a task-aware segmentation framework with a binary mask that activates a subset of the convolutional kernels to extract the most suitable features in each task, and (iii) introducing uncertainty maps to provide users informative guidance and effectively improve the segmentation accuracy with minimum annotation effort.

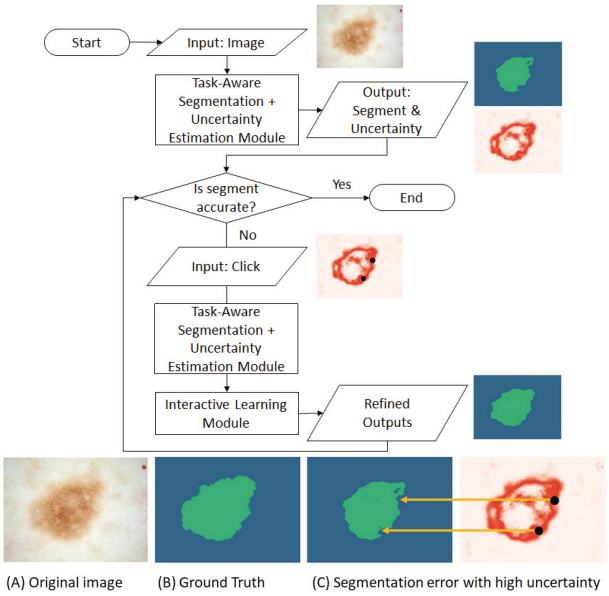

Figure 1: Schematic view of the proposed framework (Top) and an illustrative example of errors in initial segmentation of a dermatological image (Bottom). Different colors denote different semantic classes: healthy skin (blue), lesion (green). Areas with high uncertainty are colored red in the uncertainty map.

Related Works

Continual Learning Continual learning aims to train models to learn over time by encoding new knowledge while retaining previously learned knowledge. Catastrophic forgetting is a long-standing issue for continual learning (Thrun and Mitchell 1995). It usually occurs when new data differs significantly from previous data and causes learned knowledge as the network parameters to be overwritten (French 1999). Existing works address this issue through regularization, isolation, or rehearsal.

Isolation-based methods allocate different neural resources to encode the information for different tasks. For example, progressive networks (Rusu et al. 2017) freeze networks modules trained on previous tasks and expand the network architecture with additional modules for new tasks. PackNet (Mallya and Lazebnik 2018) uses iterative pruning to exploit the redundancies in large deep networks and free up parameters for learning new tasks. However, isolation-based approaches usually lack a principled way to determine the optimal sets of parameter when task information is unknown for new data, or requires storing all the isolated parameters for each task. Regularization-based models impose constraints on updating the neural weights and penalizing the changes. Learning without forgetting (LwF) (Li and Hoiem 2017) uses knowledge distillation to enforce the network predictions with old parameters to be similar to those with updated parameters. Elastic weight consolidation (EWC) (Kirkpatrick et al. 2017) and Synaptic Intelligence (SI) (Zenke, Poole, and Ganguli 2017) impose a weighted quadratic penalty on the update of the parameters, where the weight encodes the relative importance of parameters to the model's performance on previous tasks. However, regularization approaches may suffer from the trade-off between

the model's performance on the old and new tasks. The proposed framework is closely related to the works of variational continual learning (Nguyen et al. 2018; Kessler et al. 2019), which leverages Bayesian neural networks to retain a distribution over model parameters, and IBP priors to automatically determine the complexity of network. The cost of storage does not significantly grow as tasks accumulate. It also achieves good performance on both old and new tasks by balancing the reuse of learned kernels and the generation of new kernels.

Interactive Segmentation Interactive segmentation allow user to make edits by clicks (Xu et al. 2016), scribbles (Grady et al. 2005), bounding boxes (Rajchl et al. 2016; Castrejon et al. 2017), or extreme points (Maninis et al. 2018; Khan et al. 2019). Interactive segmentation models need to be pre-trained using a hold-out labeled dataset to make initial predictions before user interactions. After user annotations are collected, a majority of the models refine segmentation through spatial regularization using post-processing techniques such as conditional random fields and graph cuts (Wang et al. 2018; Dhara et al. 2018; Zhou, Chen, and Wang 2019). Since the network parameters are not updated, they do not extract knowledge from user annotations for segmenting other images. A few models propagate user annotations to unannotated pixels and retrain the model (Lin et al. 2016), but they do not incorporate specific mechanisms for knowledge transfer and thus may suffer from catastrophic forgetting.

Uncertainty quantifies the degree to which a machine learning model is unconfident about its predictions and implies whether users can trust the results (Gal 2016). The Bayesian convolutional network is a popular technique of uncertainty estimation for semantic segmentation (Kendall, Badrinarayanan, and Cipolla 2015; Jena and Awate 2019). The model may report high uncertainty on visually-difficult areas, possibly due to low contrast or brightness, and out-of-distribution areas due to the different distributions between training and testing images. Interactive segmentation poses a new challenge as uncertainty estimation shall be further used to inform users for selective annotation. However, integrating uncertainty with deep learning-based interactive segmentation is under-explored. One notable work leverages uncertainty to actively query labels on an entire image (Yang et al. 2017), but strictly speaking, it is not an interactive segmentation model as it does not refine segmentation results.

The Framework of Continual Learning for Interactive Segmentation

In this section, we start by presenting the overall architecture of the proposed CLIS framework. We then describe task formulation in continual learning and a unique training-testing protocol to best support interactive image segmentation. Finally, we present the details for continual knowledge learning through Bayesian nonparametric modeling.

Overview of the Architecture. Segmentation models such as fully convolutional network (Long, Shelhamer, and Darrell 2015) and U-Net (Ronneberger, Fischer, and Brox 2015) usually apply an encoder-decoder architecture. The proposed framework follows the design of encoder-decoder, but introduces several components uniquely designed for interactive segmentation and continual learning, which are presented in Figure 2 and discussed below.

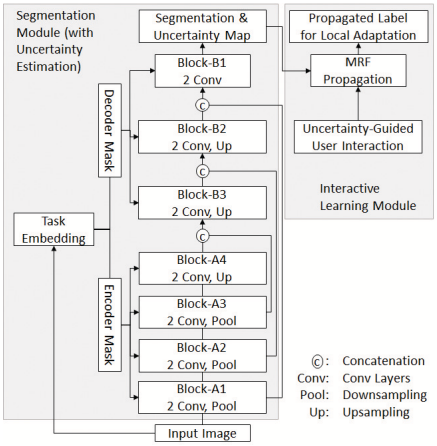

Figure 2: Architecture of the proposed framework

The segmentation module consists of an encoder-decoder architecture (Block A1-A4 and Block B1-B3). We propose to use Bayesian nonparametric modeling to incorporate knowledge learned from previous tasks as probabilistic prior to regularize parameter updates in the current task. We introduce binary masks to control convolutional kernel activation of the encoders and decoders, aiming to activate the optimal set of kernels for each task. The task embedding extracts information from images to determine the optimal kernel activation masks with or without user annotations. The uncertainty estimation is naturally obtained from the Bayesian convolutional layers and visualized as uncertainty maps to guide user annotations. The interactive learning module is inspired by Markov Random Fields (MRF) to propagate user annotations to unannotated pixels by minimizing a customized energy function.

Task Formulation for Interactive Segmentation

Interactive segmentation models usually require pre-training to make initial predictions given new images. We first use a hold-out dataset that consists of images and pixel-wise labels to pre-train the network. We then formulate interactive image segmentation as a continual learning problem with a sequence of T tasks as $\{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_T\}$. In task $\mathcal{T}_t$, the model is provided with S new images $X_t = \{x_t^1, x_t^2, ..., x_t^S\}$ and makes initial predictions, and the user provides selective annotations $A_t = \{a_t^1, a_t^2, ..., a_t^S\}$, which are considered ground truth and used to update the model's parameters. This setting can be categorized as continual learning with new instances (Lomonaco and Maltoni 2017) where all classes are known in pre-training while each subsequent task contains instances with a different subset of semantic classes.

We evaluate an interactive segmentation framework's performance from three aspects: *basic performance*, *forward knowledge transfer*, and *backward knowledge transfer*. The

basic performance measures how the model learns from user annotations to improve the current image's segmentation. The model is trained at task $\mathcal{T}_t$ using the images and the corresponding annotations and evaluated using the ground truth of unannotated pixels $U_t = \{u_t^1, u_t^2, ..., u_t^S\}$. The forward knowledge transfer measures how the model leverages user annotations to improve the segmentation of other unseen images. After training at task $\mathcal{T}_t$, the model is evaluated using unseen images from future tasks $\{\mathcal{T}_{t+1}, ..., \mathcal{T}_T\}$ in terms of initial segmentation. The backward knowledge transfer measures how the model prevents deteriorated performance on previously-seen images. The model is trained at task $\mathcal{T}_t$ and evaluated using images from previous task $\{\mathcal{T}_1, ..., \mathcal{T}_{t-1}\}$ in terms of initial prediction. In summary, at task $\mathcal{T}_t$, the training set $\mathcal{S}_t$ and evaluation set $\mathcal{Q}_t$ are

$$\begin{aligned}\mathcal{S}^t = \{X_t, A_t\}, \text{Basic}: &\mathcal{Q}_t = \{X_t, U_t\}\\ \text{Forward}: &\mathcal{Q}_t = \{X_{t+1,...,T}, U_{t+1,...,T}\}\\ \text{Backward}: &\mathcal{Q}_t = \{X_{1,...,t-1}, U_{1,...,t-1}\}\end{aligned}$$

Continual Knowledge Learning through Bayesian Nonparametric Modeling

The proposed framework organizes tasks in a sequential order to facilitate interactive segmentation for knowledge acquisition from users. Thus, it is critical to properly maintain important (latent) features and knowledge learned through prior tasks for future unseen tasks. Furthermore, as the tasks may be highly heterogeneous, not all the learned features are relevant to a given task. Instead, a specific subset of features should be selected to optimize the segmentation performance.

To address these challenges, we propose to allocate the neural network resources dynamically through Bayesian nonparametric modeling. The proposed model allows the latent feature space to continuously grow with new tasks while being capable of extracting a subset of latent features most relevant to a given task through posterior inference. In particular, for each convolutional layer l, we introduce a layer-wise binary mask to determine whether a kernel is activated or not given a specific image. Given the l-th layer's input as h^{l-1}, the k-th convolutional kernel W_k^l and the binary mask z_k^l, the output feature map h^l is

$$h^l = \{h_k^l\}_k, \quad h_k^l = z_k^l(W_k^l \circledast h^{l-1}) \tag{1}$$

where $\circledast$ denotes the convolution operator. If $z_k^l = 0$, the corresponding channel h_k^l is 0, indicating that the kernel does not contribute to this task.

To achieve dynamic architecture evolution and task-specific feature extraction, we place an Indian Buffet process (IBP) prior (Ghahramani and Griffiths 2006) on the binary mask for each layer:

$$v_k^l \sim \text{Beta}(a_0, b_0),\ \pi_k^l \sim \prod_{j=1}^{k} v_j^l,\ z_k^l \sim \text{Bern}(\pi_k^l) \tag{2}$$

where z_k^l is a Bernoulli variable indicating whether a kernel is activated or not. The feature space of kernels is potentially infinite in theory. However, to make the model computationally feasible, we take the finite approximation of IBP and place a truncation threshold K^l for the maximum number of features. For each convolutional kernel W_k^l, we place an element-wise Gaussian prior

$$W_k^l = \{w_{ki}^l\}_i, \quad w_{ki}^l \sim N(\mu_0, (\sigma_0)^2) \tag{3}$$

where i is the index of elements within the kernel.

The IBP prior allows potentially infinite latent features to jointly explain the ever-increasing data instances (images/tasks in our case). It is analogous to customers arriving at an Indian Buffet restaurant with unlimited dishes. In our interactive segmentation case, each layer is placed an IBP prior, where the tasks correspond to customers, and the convolutional kernels correspond to the dishes. An important characteristic of the IBP is that a new customer is more likely to take dishes that have been taken by a lot of previous customers, while it is still possible to take some new dishes. This characteristic is useful in our framework to automatically determine the optimal number of kernels and encourage reusing existing kernels in a new task. If the new task is similar to some previous tasks, the masks should be similar to the corresponding previous masks, indicating that the kernel activations are mostly the same. If the task is different from any previous tasks, the masks shall have some new non-zero entries, meaning that some new kernels are activated for feature extraction and interpretation.

Posterior Inference Since exact inference of the posterior of z_k^l and W_k^l is intractable, we introduce a variational distribution q to approximate those posteriors. Let $\boldsymbol{\phi} = \{\mathbf{v}, \mathbf{z}, \mathbf{W}\}$ denote all the corresponding parameters, and $\mathbf{D}_t$ denote the data that the network is fitting at task t. Assume $q(\boldsymbol{\phi})$ is factorized as

$$q(\boldsymbol{\phi}) = \prod_{k,l} q(v_k^l) q(z_k^l | v_k^l) \prod_i q(w_{ki}^l) \tag{4}$$

The key idea of variational continual learning is using the variational posterior distribution of task $\mathcal{T}_{t-1}$ as a prior distribution for task $\mathcal{T}_t$: $q_{t-1}(\boldsymbol{\phi}) = p_t(\boldsymbol{\phi})$. For the first task$\mathcal{T}_1$, the prior is set to the parameters learned from pre-training. The total loss function for task t is given as the negative evidence lower bound:

$$L_t = KL[q_t(\boldsymbol{\phi}) || q_{t-1}(\boldsymbol{\phi})] - E_{q_t(\boldsymbol{\phi})}[\ln p(\mathbf{D}_t | \boldsymbol{\phi})] \tag{5}$$

where the first term is the KL divergence that regularizes the change of parameters between task $\mathcal{T}_t$ and task $\mathcal{T}_{t-1}$, and the second term is the expectation of the log likelihood. Plugging in (4), the loss is expanded as

$$\begin{aligned}L_t = &KL[q_t(\mathbf{v}) || q_{t-1}(\mathbf{v})] + KL[q_t(\mathbf{z}|\mathbf{v}) || q_{t-1}(\mathbf{z}|\mathbf{v})]\\ &+ KL[q_t(\mathbf{W}) || q_{t-1}(\mathbf{W})] - E_{q_t(\boldsymbol{\phi})}[\ln p(\mathbf{D}_t | \mathbf{z}, \mathbf{W})]\end{aligned} \tag{6}$$

Eq (6) is the general expression of the loss function. However, each term may take different forms on different stages of interactive segmentation, which are detailed in the following sections.

We use reparameterization tricks to make the variational parameters learnable by the network through gradient descent. At task $\mathcal{T}_t$, we plug in the parameters $a_{k,t-1}^l, b_{k,t-1}^l$

learned from the previous task as a prior and notice that $q_t(v_k^l) \sim \text{Beta}(a_{k,t}^l, b_{k,t}^l)$. The first KL divergence term of (6) is approximated as

$$\begin{aligned} KL[q_t(\mathbf{v})||q_{t-1}(\mathbf{v})] &= \sum_{k,l} \ln \frac{B(a_{k,t-1}^l, b_{k,t-1}^l)}{B(a_{k,t}^l, b_{k,t}^l)} \\ &+ (a_{k,t}^l - a_{k,t-1}^l)\psi(a_{k,t}^l) + (b_{k,t}^l - b_{k,t-1}^l)\psi(b_{k,t}^l) \\ &+ (a_{k,t-1}^l + b_{k,t-1}^l - a_{k,t}^l - b_{k,t}^l)\psi(a_{k,t}^l + b_{k,t}^l) \end{aligned} \tag{7}$$

where B and ψ denote beta and digamma functions, respectively. The discrete nature of Bernoulli variables makes backpropagation infeasible. Following (Maddison, Mnih, and Teh 2016), we relax the hard constraint of Bernoulli variables with continuous ones as a concrete Bernoulli distribution, which ranges between $[0, 1]$ and is peaked at 0 and 1. Given π_k^l in the forward pass, z_k^l is now sampled from

$$z_k^l = \left\{1 + \exp[-\tau^{-1}(\ln \pi_k^l - \ln(1 - \pi_k^l) + \epsilon)]\right\}^{-1} \tag{8}$$

where τ is the temperature controlling the smoothness of concerte distribution, and $\epsilon \sim \text{Logistic}(0, 1)$ is randomly sampled from a logistic distribution (Balakrishnan 1991). With the relaxation, the second term of (6) is approximated as

$$\begin{aligned} KL[q_t(\mathbf{z}|\mathbf{v})||q_{t-1}(\mathbf{z}|\mathbf{v})] &= \sum_{k,l} \pi_{k,t}^l (\ln \pi_{k,t}^l - \ln \pi_{k,t-1}^l) \\ &+ (1 - \ln \pi_{k,t}^l)(\ln(1 - \ln \pi_{k,t}^l) - \ln(1 - \ln \pi_{k,t-1}^l)) \end{aligned} \tag{9}$$

At task $\mathcal{T}_t$, we plug in the parameters $\mu_{ki,t-1}^l, \sigma_{ki,t-1}^l$ learned from the previous task as Gaussian priors, and the third KL divergence term is estimated similarly.

Before user interaction, the model is pre-trained using a small set of data to learn the initial weights of the network, and the total loss is defined in (6), where the log-likelihood term is the expectation of cross-entropy loss of pixel-wise labels. During pre-training, we use Monte-Carlo sampling to draw samples for each layers' mask and kernels using (2) and (3), and estimate the output using (1). The total loss can be back-propagated to optimize $\mathbf{z}$ and $\mathbf{W}$.

Predicting Initial Segmentation

Given new images from task $\mathcal{T}_t$, the proposed framework first generates initial predictions for the user's reference. At this time, user annotations are not provided yet, and thus the ground-truth labels are not available. It poses a question of how the initial kernel activation for this task can be inferred. To address this problem, we propose to leverage the image data and task embedding. Each image is embedded via a variational autoencoder, which stacks three convolutional blocks and two fully-connected layers, to generate a compact representation. The image embeddings and the corresponding masks from previous tasks are stored as pairs. For a new task, its image embedding is matched to the nearest neighbor and used to retrieve the corresponding mask for initial kernel activation. Then the forward pass goes through the encoders and decoders to generate an initial segmentation and uncertainty map.

Interactive Segmentation

Once the initial segmentation is generated, users may start to adjust the initial result through annotations. Our framework accompanies the prediction with important uncertainty information to direct users to the most informative parts of the segmentation map and lessen the overall annotation effort. After the annotations are provided, they will be used to refine the model to generate an improved segmentation more consistent with users' expectations.

Uncertainty Estimation The proposed framework generates pixel-wise uncertainty estimation and visualizes as uncertainty maps. We follow the work of (Kendall and Gal 2017), which considers two sources of uncertainty: epistemic and aleatoric. Epistemic uncertainty quantifies the uncertainty in the model parameters and stems from limited training data, while the aleatoric uncertainty measures the noise inherent in each data instance. The uncertainty estimation can be naturally implemented via Monte-Carlo sampling with the Bayesian architecture. The total uncertainty is evaluated using the predictive entropy:

$$U_m = -(\bar{\boldsymbol{\theta}}_m)^T \ln \bar{\boldsymbol{\theta}}_m \tag{10}$$

where $\bar{\boldsymbol{\theta}}_m$ is predicted probability vector of pixel m averaged across MC samples.

Propagating User Annotations Given the uncertainty map, users are guided to focus on segmentation areas with high uncertainty, which are more likely to be inaccurate. It reduces users' effort to check the entire image to locate incorrect segmentation areas. Then the user annotations are propagated from annotated areas to unannotated regions to generate an updated segmentation map $\boldsymbol{\theta}^*$. Intuitively, the updated segmentation map should not deviate too much from the original result while being consistent with user annotations. Following the work with Markov random field for propagating user annotations (Lin et al. 2016), we introduce an energy function G, which consists of a unary term G_{una}, an annotation consistency term G_{ann}, and a pairwise similarity term G_{par}.

$$G = \sum_m G_{una}(m) + \sum_m G_{par}(m) + \sum_{m,n} G_{ann}(m, n) \tag{11}$$

where m and n are indices of pixels. The unary term encourages $\boldsymbol{\theta}^*$ to be similar to the original prediction $\bar{\boldsymbol{\theta}}$ for each pixel m. It is defined as the KL divergence between two categorical distributions parameterized by $\boldsymbol{\theta}_m^*$ and $\bar{\boldsymbol{\theta}}_m$:

$$G_{una}(m) = (\boldsymbol{\theta}_m^*)^T (\ln \boldsymbol{\theta}_m^* - \ln \bar{\boldsymbol{\theta}}_m) \tag{12}$$

The pairwise term encourages visually similar pixels to have similar propagated probability vectors. It takes the form of weighted symmetric KL divergence:

$$\begin{aligned} G_{par}(m, n) &= \exp(-\|\mathbf{r}_m - \mathbf{r}_n\|_2^2) \times [(\boldsymbol{\theta}_m^*)^T (\ln \boldsymbol{\theta}_m^* \\ &- \ln \boldsymbol{\theta}_n^*) + (\boldsymbol{\theta}_n^*)^T (\ln \boldsymbol{\theta}_n^* - \ln \boldsymbol{\theta}_m^*)]/2 \end{aligned} \tag{13}$$

where (m, n) denotes a pair of pixels and $\mathbf{r}_m$ denotes the LAB color vector of pixel m. The annotation consistency

term enforces a labeled pixel has a probability vector almost the same as the one-hot user annotation.

$$G_{ann}(m) = -\lambda_m(\mathbf{a}_m)^T \ln \boldsymbol{\theta}^*_m \tag{14}$$

where $\mathbf{a}_m$ is user annotation at pixel m, the weight λ_m is set to a sufficiently large value if pixel m is annotated and $\lambda_m = 0$ if otherwise.

Learning from User Annotations The propagated map is used to evaluate the loss and update the network parameters to adapt to user annotations. The total loss is defined in (6). Notice that $\mathbf{z}$ and $\mathbf{W}$ now denote the binary masks and kernels from all the encoders and decoders, and $\mathbf{D}_t = \{X_t, Y_t^*\}$ where Y_t^* denotes the propagated maps, and the term $E_{q_t(\boldsymbol{\phi})}$ equals the cross-entropy L_{ce}:

$$E_{q_t(\boldsymbol{\phi})}[\ln p(Y_t^*|\boldsymbol{\phi})] = -E_{q_t(\boldsymbol{\phi})}[\frac{1}{M}\sum_m \lambda_m \mathbf{y}^*_m \ln(\hat{\mathbf{y}}_m)] \tag{15}$$

where $\mathbf{y}^*_m$ is the one-hot vector of pixel m's propagated label, and $\hat{\mathbf{y}}_m$ is the predicted probability vector. λ_m is the corresponding weights for pixel m. For annotated pixels, λ_m is set to a sufficiently large value to enforce they are correctly predicted after network updates. For the other pixels, it is set to the default value 1.

The total loss is backpropagated to refine the network parameters. With the updated parameters of $\mathbf{z}$ and $\mathbf{W}$, the model performs another feed-forward to generate a refined segmentation map and the corresponding uncertainty estimation for user reference. This process may iterate multiple rounds until the user is satisfied.

Experiments

In this section, we report our experimental results on three real-world image datasets to demonstrate the proposed framework's performance.

Datasets. The performance of the proposed framework is evaluated using three image datasets. The first dataset is Cityscapes (Cordts et al. 2016) containing street scenes from 50 different cities, with pixel-level annotations of 5000 frames and 8 major semantic categories. The second dataset is MaSTr1325 (Bovcon et al. 2019) for marine semantic segmentation and obstacle detection. The third dataset is from the ISIC challenge for skin lesion analysis towards melanoma detection (Codella et al. 2018) with dermoscopic lesion images. Each image corresponds to a binary segmentation map that partitions primary lesion areas and healthy skins. To reduce potential overfitting, data augmentation is performed in a similar way as (Ronneberger, Fischer, and Brox 2015) using horizontal flipping, random shifting, and distortion of HSV channels.

Experimental setup. For all three datasets, we formulate twenty task sequences, each of which contains eight consecutive interactive segmentation tasks. In each task, we randomly sample one new image and simulate user interactions with the proposed framework. Given the segmentation result, sixteen erroneous areas with the high uncertainty are annotated, and the model is updated to generate a refined segmentation. We set the maximum iteration of user interactiaons to two, and record the basic performance. The performance on forward and backward knowledge transfer is evaluated after tasks 4 and 8, respectively. Then the results are averaged among task sequences. The hyper-parameters are set to $\alpha_0 = 1$, $\beta_0 = 1$, $\mu_0 = 0$, $\sigma_0 = 1$, $\tau = 2$. We use the Slic algorithm (Achanta et al. 2012) to group visually-similar pixels into superpixels and reduce the computational cost. The Adam optimizer is used for gradient-based model updates once user annotations are collected. It should be noted that simulated user interactions are suitable for quantitative evaluation because they provide a controlled environment and eliminate the factors that could affect human actions (e.g., humans usually pay more attention to the center of images and areas with high-contrast). In the appendix, we include examples of actual user interactions to illustrate the framework's usage in practice[1].

Comparison baselines and evaluation metrics. We compare with LwF (Li and Hoiem 2017), EWC (Kirkpatrick et al. 2017) and PackNet (Mallya and Lazebnik 2018). LwF and EWC are representative regularization-based approaches that introduce additional loss terms for protecting consolidated knowledge. In contrast, the proposed framework applies variational inference for network updates and uses KL divergence loss terms to regularize kernels and masks. PackNet is a representative isolation-based approach that implements network pruning to allocate different neurons for different tasks, while the proposed framework relies on a task-specific mask to control kernel activation. For performance evaluation, we use mean intersect-over-union (IoU) as metric.

Performance comparison. We first present qualitative comparisons with the baselines in Figure 3. After training the model at Task 4, the refined segmentation map is visualized for basic performance. We also visualize the initial predictions of the image from Task 6 for forward knowledge transfer and the image from Task 2 for backward knowledge transfer. The proposed framework usually performs better in identifying boundaries (e.g., the boundaries of buildings in Dataset 1 Task 4, vehicles in Dataset 2 Task 2, and lesions in Dataset 3 Tasks 2). The performance may be attributed to (1) uncertainty-guided user interactions that provide informative annotations and (2) task-specific mask that activates suitable kernels for feature extraction and interpretation.

Quantitative comparisons are provided in Table 1. In most cases, the proposed CLIS framework outperforms the competing baselines. LwF leverages knowledge distillation to regularize the network update. However, for tasks with small sizes, it may not be as effective as EWC, which directly regularizes the update of the parameters. PackNet incorporates a flexible architecture and performs well on preserving knowledge, but lacks a principled way of fine-tuning the existing kernels. Thus some kernels may overfit to the noises.

Discussions. Using a layer-wise binary mask to control kernel activation also improves the framework's interpretability because we can track the activated kernels at each

[1]The source code and the appendix are available at https://github.com/ritmininglab/CLIS

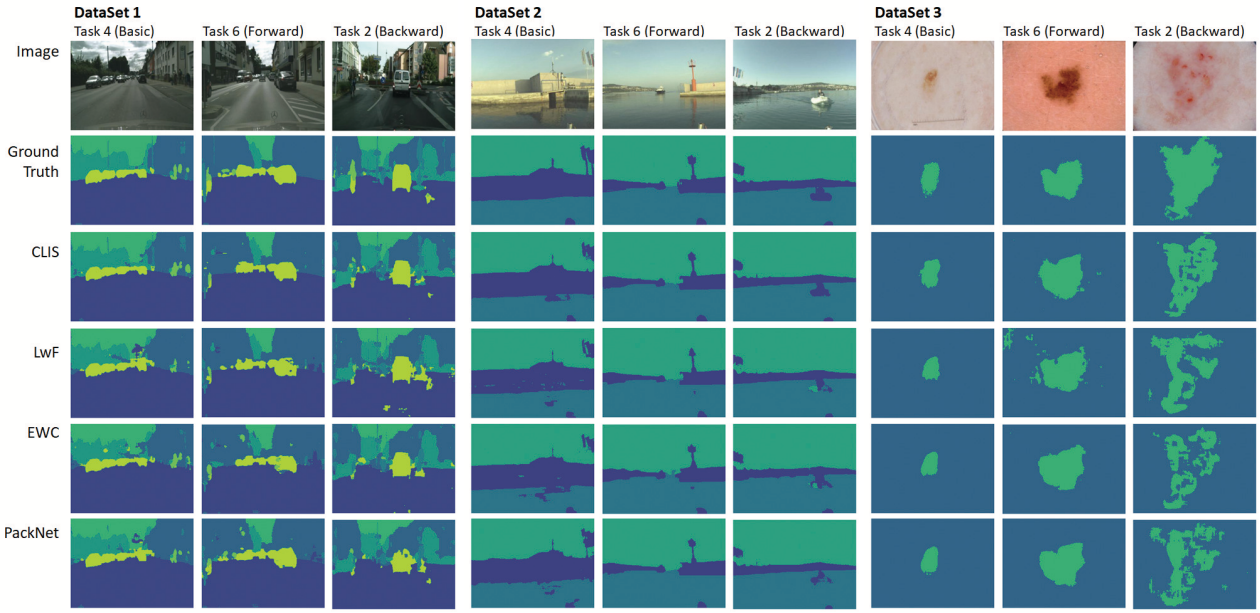

Figure 3: Illustrative examples of segmentation results from Dataset 1 (Left) and 2 (Middle) and 3 (Right)

	Evaluation	CLIS	LwF	EWC	PackNet
Dataset1	Basic	0.893	0.859	0.874	0.869
	Forward	0.836	0.818	0.825	0.827
	Backward	0.857	0.831	0.846	0.843
Dataset2	Basic	0.957	0.937	0.945	0.942
	Forward	0.928	0.914	0.923	0.919
	Backward	0.941	0.920	0.933	0.927
Dataset3	Basic	0.782	0.749	0.765	0.757
	Forward	0.697	0.674	0.689	0.684
	Backward	0.731	0.703	0.722	0.715

Table 1: Quantitative comparison of basic performance, forward and backward knowledge transfer (IoU)

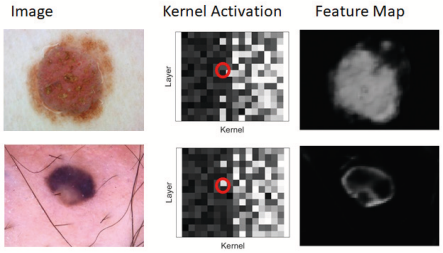

Figure 4: Illustrative examples of kernel activation for two tasks: Due to space limit, not all kernels are visualized in the kernel activation map. Dark grid indicates the corresponding kernel is activated.

task. In Figure 4, the images from the two tasks reveal distinct disease morphology. We visualize the kernel activations as gray-scale matrices for those tasks. Each row of the matrix corresponds to a layer, and each column corresponds to a kernel. The highlighted seventh kernel from the sixth layer is activated in the first task (i.e., the corresponding parameter π is close to 1) but not in the second task (i.e., π is close to 0). We also visualize the related output feature maps of this kernel in both tasks. It shows that the kernel captures important features of the disease morphology in the first task but almost no features in the second task. We also conduct an ablation study to evaluate the contribution of uncertainty-guided interaction to model performance. Please refer to the appendix for details.

Overconfidence in predictions may happen where a misclassified region corresponds with low uncertainty. It usually results from the out-of-distribution (OOD) issue. The uncertainty estimation method applied to the proposed framework is able to detect OOD. However, it should be noted that an uncertainty estimation method may not accurately detect all OOD cases. Practically, one solution is to use ensemble methods (i.e., using multiple uncertainty estimation methods jointly, because the chance of missing an OOD case by all methods is lower than relying on a single method). Another solution is to allow end-users to make final judgments.

Conclusion

In this work, we formulate interactive segmentation as a continual learning problem and propose a framework to effectively learn from user annotations to improve the segmentation performance on current and future tasks and prevent catastrophic forgetting on previous tasks. Uncertainty information is leveraged to provide users with informative guidance. An interesting future direction is speeding up the network updates to reduce the waiting time during user interaction. Another direction is extending the allowed interaction from clicks to other types such as scribbles and boxes to make the interaction process more efficient.

Acknowledgements

This research was partially supported by NSF IIS award IIS-1814450 and ONR award N00014-18-1-2875. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the official views of any funding agency. We would like to thank the anonymous reviewers for reviewing the manuscript.

Ethical Impact

This work will provide both theoretical underpinning and empirical evaluation on the design of human-in-the-loop learning in interactive segmentation, aiming to infuse human expertise into the training-evaluation-feedback loop and make the model more customized to human needs. To the best of our understanding, we do not identify the negative societal implications of the proposed framework.

References

Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; and Süsstrunk, S. 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence* 34(11): 2274–2282.

Balakrishnan, N. 1991. *Handbook of the logistic distribution*. CRC Press.

Bovcon, B.; Muhovič, J.; Perš, J.; and Kristan, M. 2019. The MaSTr1325 dataset for training deep USV obstacle detection models. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.

Castrejon, L.; Kundu, K.; Urtasun, R.; and Fidler, S. 2017. Annotating object instances with a polygon-rnn. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5230–5238.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. 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.

Codella, N. C.; Gutman, D.; Celebi, M. E.; Helba, B.; Marchetti, M. A.; Dusza, S. W.; Kalloo, A.; Liopyris, K.; Mishra, N.; Kittler, H.; et al. 2018. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, 168–172. IEEE.

Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Dhara, A. K.; Ayyalasomayajula, K. R.; Arvids, E.; Fahlström, M.; Wikström, J.; Larsson, E.-M.; and Strand, R. 2018. Segmentation of post-operative glioblastoma in mri by u-net with patient-specific interactive refinement. In *International MICCAI Brainlesion Workshop*, 115–122. Springer.

French, R. M. 1999. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences* 3(4): 128–135.

Gal, Y. 2016. Uncertainty in deep learning. *University of Cambridge* 1: 3.

Ghahramani, Z.; and Griffiths, T. L. 2006. Infinite latent feature models and the Indian buffet process. In *Advances in neural information processing systems*, 475–482.

Grady, L.; Schiwietz, T.; Aharon, S.; and Westermann, R. 2005. Random walks for interactive organ segmentation in two and three dimensions: Implementation and validation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 773–780. Springer.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

Jena, R.; and Awate, S. P. 2019. A bayesian neural net to segment images with uncertainty estimates and good calibration. In *International Conference on Information Processing in Medical Imaging*, 3–15. Springer.

Kendall, A.; Badrinarayanan, V.; and Cipolla, R. 2015. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. *arXiv preprint arXiv:1511.02680* .

Kendall, A.; and Gal, Y. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, 5574–5584.

Kessler, S.; Nguyen, V.; Zohren, S.; and Roberts, S. 2019. Indian Buffet Neural Networks for Continual Learning. *arXiv preprint arXiv:1912.02290* .

Khan, S.; Shahin, A. H.; Villafruela, J.; Shen, J.; and Shao, L. 2019. Extreme Points Derived Confidence Map as a Cue for Class-Agnostic Interactive Segmentation Using Deep Neural Network. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 66–73. Springer.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114(13): 3521–3526.

Li, Z.; and Hoiem, D. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence* 40(12): 2935–2947.

Lin, D.; Dai, J.; Jia, J.; He, K.; and Sun, J. 2016. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3159–3167.

Lomonaco, V.; and Maltoni, D. 2017. Core50: a new dataset and benchmark for continuous object recognition. *arXiv preprint arXiv:1705.03550* .

Long, J.; Shelhamer, E.; and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3431–3440.

Maddison, C. J.; Mnih, A.; and Teh, Y. W. 2016. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712* .

Mallya, A.; and Lazebnik, S. 2018. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7765–7773.

Maninis, K.-K.; Caelles, S.; Pont-Tuset, J.; and Van Gool, L. 2018. Deep extreme cut: From extreme points to object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 616–625.

Nguyen, C. V.; Li, Y.; Bui, T. D.; and Turner, R. E. 2018. Variational Continual Learning. In *International Conference on Learning Representations*.

Rajchl, M.; Lee, M. C.; Oktay, O.; Kamnitsas, K.; Passerat-Palmbach, J.; Bai, W.; Damodaram, M.; Rutherford, M. A.; Hajnal, J. V.; Kainz, B.; et al. 2016. Deepcut: Object segmentation from bounding box annotations using convolutional neural networks. *IEEE transactions on medical imaging* 36(2): 674–683.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 234–241. Springer.

Rusu, A. A.; Večerík, M.; Rothörl, T.; Heess, N.; Pascanu, R.; and Hadsell, R. 2017. Sim-to-real robot learning from pixels with progressive nets. In *Conference on Robot Learning*, 262–270.

Thrun, S.; and Mitchell, T. M. 1995. Lifelong robot learning. *Robotics and autonomous systems* 15(1-2): 25–46.

Wang, G.; Zuluaga, M. A.; Li, W.; Pratt, R.; Patel, P. A.; Aertsen, M.; Doel, T.; David, A. L.; Deprest, J.; Ourselin, S.; et al. 2018. DeepIGeoS: a deep interactive geodesic framework for medical image segmentation. *IEEE transactions on pattern analysis and machine intelligence* 41(7): 1559–1572.

Xu, N.; Price, B.; Cohen, S.; Yang, J.; and Huang, T. S. 2016. Deep interactive object selection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 373–381.

Yang, L.; Zhang, Y.; Chen, J.; Zhang, S.; and Chen, D. Z. 2017. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention*, 399–407. Springer.

Zenke, F.; Poole, B.; and Ganguli, S. 2017. Continual learning through synaptic intelligence. *Proceedings of machine learning research* 70: 3987.

Zhou, B.; Chen, L.; and Wang, Z. 2019. Interactive Deep Editing Framework for Medical Image Segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 329–337. Springer.