Dual-Level Adaptive Information Filtering for Interactive Image Segmentation

Ervine Zheng **Qi Yu*** **Rui Li** **Pengcheng Shi** **Anne Haake**
Rochester Institute of Technology

Abstract

Image segmentation can be performed interactively by accepting user annotations to refine the segmentation. It seeks frequent feedback from humans, and the model is updated with a smaller batch of data in each iteration of the feedback loop. Such a training paradigm requires effective information filtering to guide the model so that it can encode vital information and avoid overfitting due to limited data and inherent heterogeneity and noises thereof. We propose an adaptive interactive segmentation framework to support user interaction while introducing dual-level information filtering to train a robust model. The framework integrates an encoder-decoder architecture with a style-aware augmentation module that applies augmentation to feature maps and customizes the segmentation prediction for different latent styles. It also applies a systematic label softening strategy to generate uncertainty-aware soft labels for model updates. Experiments on both medical and natural image segmentation tasks demonstrate the effectiveness of the proposed framework.

1 INTRODUCTION

Human-in-the-loop machine learning leverages both human knowledge and machine intelligence to train accurate and reliable models (Wu et al., 2021). Unlike the conventional model development process, where the model is trained with a large amount of data before testing, human-in-the-loop machine learning seeks more frequent feedback from human users. As a result, the model is updated with a smaller batch of data in each iteration of the feedback loop. Such a training paradigm requires information filtering mechanisms to guide the model to extract vital information to be encoded by the model while neglecting noises.

In this paper, we integrate human-in-the-loop machine learning with semantic image segmentation. The interactive task is formulated as a procedure where the model makes initial segmentation prediction for given images, and users interact with the system by annotating a few pixels as supervision, which is used by the model to refine the segmentation results. It offers a viable solution to tackle complex segmentation tasks, especially those from specialized domains (*e.g.*, medicine and security surveillance), where fully automatic models cannot guarantee perfect segmentation results that are satisfactory to end-users.

This setting introduces unique challenges of information filtering. First, the heterogeneous image styles may incur the problem of distributional shift between the training images and the images provided by the user. Consider medical image segmentation. The new images provided by the user for segmentation may vary due to different imaging devices, morphologic characteristics, anatomic structures, patient-specific issues, and other factors. In those cases, the latent styles may incur highly uncertain model predictions and hurt the interaction process. The second challenge is how to leverage limited user annotations efficiently. Given a new image, users typically annotate a few areas rather than labeling all pixels. Therefore post-processing techniques (Dhara et al., 2018; Zhou et al., 2019) are used to assign class labels to unannotated regions. However, those assigned labels are not ground truth, and therefore simply treating them as target labels for model update may incur errors.

To address the above challenges, we propose an interactive framework with dual-level information filtering mechanisms. The first-level information filtering aims to disentangle the latent image styles from contents. Prior works in this direction address the issue of heterogeneous styles by transforming the styles of testing

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151. Copyright 2022 by the author(s). * Corresponding author.

images to that of training images and performing segmentation on transformed images using a Gram matrix for style alignment (Ma et al., 2019), adaptive instance normalization (Liu et al., 2020), or adversarial training (Hou et al., 2019). However, the style transformation essentially involves image generation, and thus generation errors can be introduced and hurt the downstream segmentation performance. In addition, the above prior works are proposed for automated segmentation tasks, whereas the style disentanglement for interactive segmentation is under-explored.

In contrast, the proposed method directly analyzes the latent styles of images and applies style-specific augmentation to feature maps. Specifically, we leverage an encoder-decoder architecture to encode the global knowledge that is necessary to perform segmentation. The styles of images are grouped into latent patterns. Since the exact number of patterns is unknown, we introduce a Dirichlet process prior to automatically discover the optimal number of patterns. Given the style pattern assignment, a light-weighted convolutional block is applied through residual connections to the decoder layers to adjust those layers' output feature maps. In this way, the style-related local information is augmented to the global information and contributes to the improved segmentation performance.

The second-level information filtering aims to spatially down weight the noisy areas of the image. Soft label classification provides a promising direction. Different from one-hot hard labels, soft labels are probability vectors ranging from (0,1) that implies a pixel can be a member of multiple classes with corresponding probabilities (Galstyan and Cohen, 2007). Prior works leverage soft labels to train robust segmentation models by calculating soft masks during data pre-processing (Gros et al., 2021), applying label softening guided by superpixels (Li et al., 2020), or using knowledge distillation to down weight corrupted labels (Zhang et al., 2020). In summary, the label softening techniques are based on low-level features of the image or the annealed probability vector. Besides, the prior works are proposed for automated segmentation tasks, whereas label softening for interactive segmentation is under-explored.

In contrast, the proposed model leverage uncertainty estimation to generate soft labels based on the initial segmentation prediction, which is combined with user-provided labels on annotated areas to update network parameters. Specifically, uncertainty quantifies the degree to which a machine learning model is unconfident about its predictions. Using a Bayesian network with Monte-Carlo sampling, the uncertainty estimation from initial segmentation is systematically integrated into the soft labels for pixels with unconfident predictions. Noisy areas are naturally down-weighted during network updates and lead to a less significant impact on the network parameters.

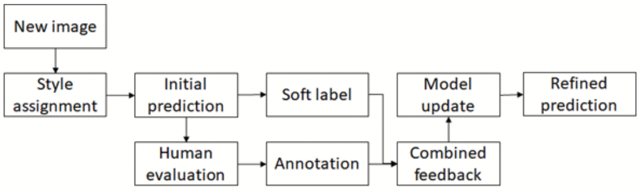


Figure 1: The high-level workflow of the proposed interactive segmentation framework

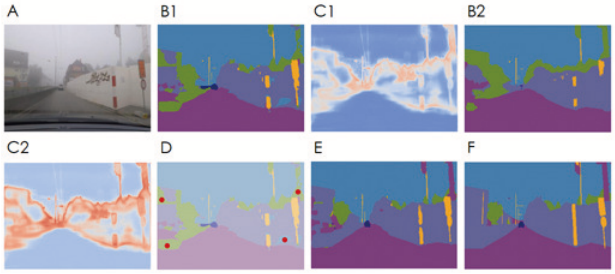


Figure 2: An illustrative example. Given an image (A), the model applies the first-level information filtering and makes an initial prediction (B1) with predictive entropy visualized in (C1). Without information filtering, the segmentation (B2) is worse, and the corresponding entropy (C2) is higher. After that, the user annotates a few pixels (D) highlighted in red, and the model applies the second-level information filtering and refines segmentation (E). Ground-truth is given in (F).

Figures 1 and 2 provide a high-level summary of the proposed framework and an illustrative example. For the first-level information filtering, the model selectively triggers the residual block to adjust the output of decoder layers for improved segmentation performance. In addition, the model can discover new style patterns during the interaction process and allocate additional neural resources to make the model adapt to the image. The second-level information filtering is applied to transform the initial segmentation prediction to uncertainty-aware soft labels, which are then combined with user annotations for network updates.

The major contribution of this paper is threefold:

- an adaptive interactive segmentation framework to support user interaction while introducing dual-level information filtering to train a robust model,
- a style-aware augmentation module that extracts style information from images and customizes the segmentation prediction for latent styles,
- a systematic label softening strategy that leverages uncertainty information to generate soft labels and integrate them with user annotations for effective

model updates.

2 RELATED WORKS

Recently, deep learning-based algorithms have achieved great success in interactive segmentation. User interactions are collected in different forms, such as clicks (Xu et al., 2016), scribbles (Lin et al., 2016), and bounding boxes (Castrejon et al., 2017). After user annotations are collected, a majority of algorithms refine segmentation through spatial regularization using post-processing techniques such as conditional random fields and graph cuts (Dhara et al., 2018; Zhou et al., 2019) to combine the initial segmentation prediction with annotations. To constrain annotated areas to have correct labels, a few models treat user annotations as partial labels to retrain the model or perform a back-propagating refinement scheme (Kontogianni et al., 2019; Jang and Kim, 2019). Most prior works are formulated in the setting of class-agnostic foreground segmentation or binary segmentation (Wang et al., 2018a,b), while a few works considers the semantic segmentation setting (Lin et al., 2016; Bearman et al., 2016). Our setting falls into the latter category. It should be noted that the feedback loop in the interactive setting requires frequent model updates with a relatively small batch of data, and therefore information filtering is critical to the model for encoding vital information while neglecting noises. However, integrating information filtering with segmentation is relatively under-explored, and our work aims to fill the gap.

The proposed dual-level information filtering strategy is closely related to style disentanglement and domain adaptation. Existing works leverage style transformation for segmentation tasks where the style of testing images deviates from training images. (Ma et al., 2019) proposes to apply style transformation of testing images by aligning the Gram matrix of transformed images with the training images and perform segmentation on transformed images. (Liu et al., 2020) proposes to leverage adaptive instance normalization for style transformation, while (Hou et al., 2019) proposes to leverage adversarial training. However, the style transformation essentially involves image generation, and generation errors can be introduced and hurt the downstream segmentation performance. In contrast, the proposed method directly applies style-specific augmentation to customize segmentation results.

The proposed dual-level information filtering strategy is also closely related to soft label classification. Soft labels are widely used in specialized domains (e.g., medicine) where the determination of ground truth labels is usually difficult. A few existing works consider incorporating soft labels to segmentation tasks for training robust and generalizable models. (Gros et al., 2021) proposes to apply soft masks as a result of data pre-processing and leverage normalized ReLU activation with adaptive wing loss for training segmentation models. (Li et al., 2020) propose to soften pixel labels depending on how superpixels interact with the ground-truth segmentation boundaries, while Zhang et al. (2020) proposes to leverage knowledge distillation to down weight the corrupted labels. In summary, the label softening techniques are based on low-level features of the image or the annealed probability vectors. In contrast, the proposed model leverage uncertainty estimation to generate soft labels.

We provide additional discussions about other related concepts in Appendix C.

3 THE PROPOSED FRAMEWORK

In this section, we provide details of the proposed framework, including the backbone Bayesian neural network, the information filtering strategy, and the posterior inference for updating the model parameters. Major notations are summarized in Appendix A.

3.1 Backbone Network

The proposed framework uses an encoder-decoder architecture for segmentation. The mainstream encoder and decoder have stacked convolutional blocks. Each block has two convolutional layers with ReLU activation, with pooling layers for the encoder blocks and upsampling layers for the decoder blocks. Unlike conventional encoder-decoder networks, we use a Bayesian network as the backbone for uncertainty estimation and regularization. For each convolutional layer l, we place a Gaussian prior on its kernels as

$$\mathbf{W}^l = \left\{w_i^l\right\}_i, \quad w_i^l \sim N(\mu_0, \sigma_0^2) \tag{1}$$

where i is the index of elements within the kernel. Such settings are also applied to the bias terms. For the rest of the paper, we use $g(\cdot)$ to denote the Bayesian convolutional layer.

$$\mathbf{h}^l = g(\mathbf{h}^{l-1}) = \text{relu}\left(\mathbf{W}^l \circledast \mathbf{h}^{l-1}\right) \tag{2}$$

where $\circledast$ denotes the convolutional operation, $\mathbf{h}^{l-1}$ and $\mathbf{h}^l$ denote layer input and output. Bias terms are omitted in the equations for simplicity, and the default activation is ReLU if not specified. Unlike conventional networks, the training of BNN aims to optimize the posterior distribution of the network's weights, denoted as $q(w_i^l) = N(\mu_i^l, (\sigma_i^l)^2)$. A common practice is to use the reparameterization tricks during feed-forwards to sample w_i^l from the posterior distribution and update the posterior parameters during backpropagation.

3.2 First-Level Information Filtering

The first-level information filtering leverages the style information of each image and applies style-aware augmentation to customize the segmentation prediction for distinct images.

It should be noted that the style patterns are not directly discovered from image data. Instead, each pattern corresponds to a set of residual blocks that adjust decoder layers' output feature maps to generate accurate and confident predictions. Since the number of latent style patterns is unknown, we use a Dirichlet process prior to automatically determine the optimal number of patterns based on the nature of the data. The DP prior allows potentially infinite patterns to explain target pixel-wise labels given input images. It is analogous to customers entering a Chinese restaurant with unlimited tables (Blei and Jordan, 2006). A new customer sits down at a table with a probability proportional to the number of customers already sitting there. Additionally, a customer opens a new table with a probability proportional to the scaling parameter. This probability distribution over the tables follows a Dirichlet process. An important characteristic of the DP is that a new customer is more likely to sit at a table that has been taken by a lot of previous customers, while it is still possible to open a new table. This characteristic is useful in our framework to determine the optimal number of patterns and encourage reusing existing patterns.

To make the corresponding parameters learnable through stochastic gradient descent, we leverage the stick-breaking representation of the Dirichlet process with truncation (Blei and Jordan, 2006).

$$\{v_m\}_{m=1}^{M} \sim \mathrm{Beta}(1, b_0), \quad c_m = v_m \prod_{m'=1}^{m-1} (1 - v_{m'})$$
$$\mathbf{z}_n = [z_{n,1}, ..., z_{n,M}] \sim \mathrm{Cat}([c_1, ..., c_M]) \tag{3}$$

where Beta denotes Beta distribution and Cat denotes Categorical distribution. $\{c_m\}_m$ can be interpreted as stick portions and $\mathbf{z}_n$ can be interpreted as a pattern assignment where m is the index of latent pattern and n is the index of data instance. M is the truncation level. Although the number of patterns is potentially infinite, a common practice is to set a truncation level, which is a sufficiently large positive integer.

Again, we parameterize the posterior distribution for auxiliary variable v_m as $\mathrm{Beta}(a_m, b_m)$. To avoid the arg max operation during sampling, we relax Beta distribution to a concrete distribution, and one option is the Kumaraswamy distribution (Kumaraswamy, 1980) with the density function

$$\mathrm{Kuma}(v_m; a_m, b_m) = a_m b_m v_m^{a_m - 1} (1 - v_m^{a_m})^{b_m - 1} \tag{4}$$

and in our case, the samples can be drawn from

$$v_m \sim (1 - u^{1/b_m})^{1/a_m}, \quad u \sim \mathrm{Uniform}(0, 1) \tag{5}$$

The posterior distribution for pattern assignment $\mathbf{z}_n$ is reparameterized as categorical distribution $\mathrm{Cat}(\zeta_n)$, which is relaxed to a concrete distribution. One option of relaxation is the Gumbel-softmax distribution (Jang et al., 2016) with the density function

$$\mathrm{GS}(z_{n,1:M}; \zeta_n, \tau)$$
$$= \Gamma(M) \tau^{M-1} \left(\sum_m \zeta_{n,m} / z_{n,m}^{\tau} \right)^{-M} \prod_m (\zeta_{n,m} / z_{n,m}^{\tau+1}) \tag{6}$$

where τ is the temperature parameter. And in our case, the samples can be drawn from

$$\mathbf{z}_n \sim \mathrm{softmax}(\ln(\zeta_n) + \epsilon)/\tau), \quad \epsilon \sim \mathrm{Gumbel}(0, 1) \tag{7}$$

Given the style pattern assignment sampled from the concrete distribution, the augmentation module applies style-specific augmentation to customize the segmentation prediction for distinct images. For each style, a light-weighted residual convolutional block is applied to adjust the output feature maps of the decoder layers. We design the residual block to include a number of depth-wise and point-wise convolutional kernels. There are d^{l-1} depth-wise kernels in the shape of $3 \times 3 \times 1$ followed by d^l point-wise convolutional kernels in the shape of $1 \times 1 \times d^{l-1}$, where d^{l-1} is the number of channels in input feature map and d^l is the number of channels in output feature map. Compared with conventional convolutional kernels, the combination of depth-wise and point-wise kernels has fewer trainable parameters and at the same time achieves comparable performance (Howard et al., 2017).

Denote the main-stream decoder layer as $g(\cdot)$, the depth-wise convolutional filter for pattern m as $g_{m,1}(\cdot)$ and the point-wise convolutional filter as $g_{m,2}(\cdot)$, the augmented output is

$$\mathbf{h}^l = g(\mathbf{h}^{l-1}) + \sum_m z_{n,m} g_{m,2}(g_{m,1}(g(\mathbf{h}^{l-1}))) \tag{8}$$

where $\mathbf{h}^{l-1}$ is the input feature map and $\mathbf{h}^l$ is the output feature map. Again, we apply the setting of Bayesian convolutional layers to $g_{m,1}(\cdot)$ and $g_{m,2}(\cdot)$. This design is suitable for the proposed residual block. Recall that the main-stream decoder is responsible for making segmentation predictions, and therefore the global information necessary to perform segmentation is already captured by the main-stream decoder. In contrast, the residual block is responsible for augmentation that customizes the segmentation prediction for different style patterns, and a light-weighted architecture is efficient

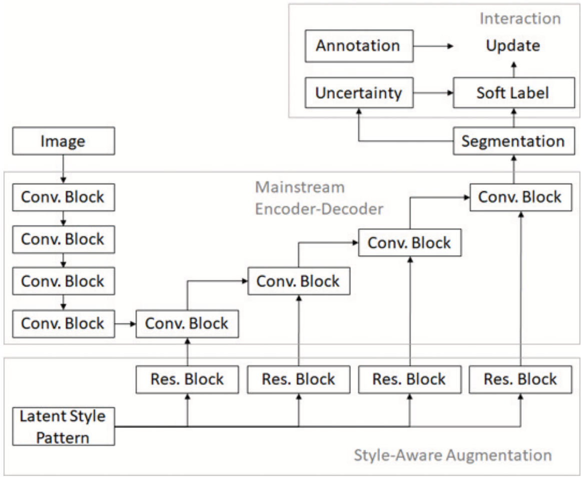


Figure 3: Schematic view of the proposed framework

to achieve the goal. In addition, the light-weighted architecture has fewer trainable parameters and thus speeds up the training process. A schematic view of the proposed framework is provided in Figure 3.

For the mainstream encoder-decoder, each convolutional block contains two convolutional layers, followed by down-sampling for the encoder or up-sampling for the decoder. The style-aware augmentation model consists of a number of light-weight residual blocks to adjust the output feature maps of decoder layers. Each latent style has one residual block connected to each decoder convolutional block. Each image corresponds to a latent style pattern assignment vector. Given the pattern assignment, the corresponding residual block is used. It should be noted that the proposed framework has an interaction module, but it does not involve any new layers. Instead, the interaction module generates a soft label map based on initial segmentation prediction. Then user-provided labels on annotated areas are combined with the soft labels on unannotated areas to generate the target labels, which are used to retrain the network and update parameters of the style-aware augmentation module, so the network can predict the refined segmentation.

3.3 Model Training And Initial Prediction

The model needs to be pretrained to make initial predictions. During pre-training, all modules are trainable, including the mainstream encoder and decoder, as well as the style discovery module and the style-aware augmentation modules. Here we denote the parameters of encoder and decoder as $\mathbf{W}$, the parameters of the style-aware augmentation module as $\mathbf{W^a}$. Let $\boldsymbol{\phi} = \{\mathbf{v}, \mathbf{z}, \mathbf{W}, \mathbf{W^a}\}$ denote all trainable parameters, and $(\mathbf{X}, \mathbf{Y})$ denote the training data. We introduce a variational distribution q to approximate the posterior distribution of $\boldsymbol{\phi}$. A general form of the objective function is given as the negative evidence lower bound:

$$L = KL[q(\boldsymbol{\phi})||p(\boldsymbol{\phi})] - E_q[\ln p(\mathbf{Y}|\boldsymbol{\phi}, \mathbf{X})] \tag{9}$$

where the first term is the KL divergence that regularizes the parameters between the prior distribution and the posterior distribution, and the second term is the expectation of cross-entropy.

To update model parameters, we assume $q(\boldsymbol{\phi})$ is factorized as follows to expand the loss in (9).

$$q(\boldsymbol{\phi}) = \prod_m q(v_m) \prod_n q(z_{m,n}|v_m) \prod_{l,i} q(w_i^l) \prod_{m,l,i} q((w_m^a)_i^l) \tag{10}$$

The optimization problem is defined by minimizing the evidence lower bound as

$$\begin{aligned} \arg \min_{q(\mathbf{z},\mathbf{v},\mathbf{W},\mathbf{W}^a)} & L = KL[q(\mathbf{W})||p(\mathbf{W})] \\ & + KL[q(\mathbf{v})||p(\mathbf{v})] + KL[q(\mathbf{z})||p(\mathbf{z}|\mathbf{v})] \\ & + KL[q(\mathbf{W}^a)||p(\mathbf{W}^a)] - E_q[\ln p(\mathbf{Y}|\mathbf{z}, \mathbf{W}, \mathbf{W}^a, \mathbf{X})] \end{aligned} \tag{11}$$

The first KL-divergence term is expanded as

$$\begin{aligned} & KL[q(\mathbf{W})||p(\mathbf{W})] \\ = & \sum_{l,i} \left[\ln \sigma_0 - \ln \sigma_i^l + \frac{(\sigma_i^l)^2 + (\mu_i^l - \mu_0)^2}{2(\sigma_0)^2} - \frac{1}{2} \right] \end{aligned} \tag{12}$$

$KL[q(\mathbf{W}^a)||p(\mathbf{W}^a)]$ is calculated in the same way. The second KL-divergence term is expanded as

$$\begin{aligned} KL[q(\mathbf{v})||p(\mathbf{v})] = & \sum_m \ln \frac{B(1, b_0)}{B(a_m, b_m)} + (a_m - 1)\psi(a_m) \\ & + (b_m - b_0)\psi(b_m) + (1 + b_0 - a_m - b_m)\psi(a_m + b_m) \end{aligned} \tag{13}$$

where $B(\cdot)$ denotes Beta function and $\psi(\cdot)$ denotes digamma function. The third KL-divergence term is expanded as

$$KL[q(\mathbf{z})||p(\mathbf{z}|\mathbf{v})] = \sum_{m,n} \zeta_m (\ln \zeta_{n,m} - \ln c_m) \tag{14}$$

To make initial segmentation for a new image x_n, it is critical to determine the appropriate style pattern and use the corresponding augmentation to enhance the quality of initial segmentation. Since user annotation is not collected yet, a good augmentation should result in a confident prediction. Therefore, we optimize the following objective function with respect to $q(z_n)$ while fixing all other parameters

$$\arg \min_{q(\mathbf{z})} L = KL[q(\mathbf{z})||p(\mathbf{z}|\mathbf{v})] - E_q[p(\hat{\mathbf{Y}}) \ln p(\hat{\mathbf{Y}})] \tag{15}$$

where $\hat{\mathbf{Y}}$ denotes prediction. The first term serves as regularization and the second term is the entropy. The optimal $q(z)$ corresponds to the optimal augmentation, and the initial prediction is the corresponding $\hat{\mathbf{Y}}$.

3.4 Second-Level Information Filtering

Given the initial segmentation, the user may selectively annotate a few pixels, and the network leverages both the initial segmentation and user annotation to update its parameters. The second-level information involves uncertainty estimation to generate soft labels based on the initial segmentation prediction, which is then used for network updates.

For uncertainty estimation, We follow (Kendall and Gal, 2017) to model epistemic and aleatoric uncertainties. Epistemic uncertainty is estimated using the Bayesian neural network with Monte-Carlo sampling, while the aleatoric uncertainty is evaluated by adding a head to the segmentation decoder's last layer before softmax to estimate the variance. Instead of predicting the logit for pixel m, the network outputs the mean o_m and variance ω_i^m. The logit $\hat{\zeta}_m$ is sampled from a Gaussian distribution and squashed through softmax to generate the predicted probability $\hat{\theta}_m$

$$\hat{\zeta}_m \sim N(o_m, \omega_m^2), \quad \hat{\theta}_m = \text{softmax}(\hat{\zeta}_m) \tag{16}$$

The soft label is the expectation of the predicted probability, which estimated by the sample mean of $\hat{\theta}_m$.

$$\tilde{y}_m = E_{\zeta_m \sim N(o_m, \omega_m)}[\hat{\theta}_m] \approx \frac{1}{S} \sum_{s=1}^{S} \hat{\theta}_m^{(s)} \tag{17}$$

Once user annotation is collected, we combine user-provided labels on annotated areas with the soft labels on unannotated areas to update network parameters. It should be noted that the neighboring pixels to user annotations should be assigned the same label as the annotated pixels if they reveal similar low-level features. Therefore, we calculate superpixels using the SLIC algorithm (Achanta et al., 2012) and assign the corresponding superpixel with the user-provided label.

Soft labels are a natural choice for updating the network to adapt to user annotation, because we do not know the ground truth labels for most pixels except for those annotated by users. In addition, noisy areas of an image are usually blurred, confusing, or visually difficult to recognize. Forcing the model to fit those areas increases the risk of overfitting. Such noise can be quantified by the aleatoric uncertainty, parameterized by the standard deviation term ω_m for pixel m. With a large ω_m, the soft labels calculated by (17) are far away from the one-hot vector. Intuitively, the uncertainty information is systematically integrated into the soft label for pixels with unconfident predictions. When the model is updated to fit the soft labels, areas with high aleatoric uncertainty are naturally down-weighted.

To show such characteristics, we formally establish the relationship between aleatoric uncertainty and the 'softness' of labels: *a larger aleatoric uncertainty leads to softer labels, indicating a less certain prediction.* Recall that the mean of logits is a vector $o_m = (o_{m,1}, o_{m,2} \dots, o_{m,K})^\top$, where K is the number of classes. For simplicity, we assume that the variance ω_m^2 is shared for all $k \in [K]$. Next, we first show that a larger ω_m makes the approximate expectation of the softmax values 'softer'.

Lemma 1. *As ω_m increases, (i) the approximate upper bound of $E[\theta_m]$ decreases and (ii) the approximate lower bound of $E[\theta_m]$ increases.*

Proof sketch. The proof of Lemma 1 requires sorting entries in o_m in an descending order so that $o_{m,1} \geq o_{m,2} \dots \geq o_{m,K}$. Then we estimate the expectation of softmax probability as

$$\tilde{E}[\theta_{m,k}] = \left[1 + \sum_{k' \neq k} \exp(c_{m,kk'})\right]^{-1} \tag{18}$$

where $c_{m,kk'} \propto o_{m,k} - o_{m,k'}$ It is straightforward to show that

$$\forall k_1 > k_2 : \quad \tilde{E}[\theta_{m,k_1}] \geq E[\theta_{m,k_2}] \tag{19}$$

And using counterevidence, we can show that

$$\tilde{E}[\theta_{m,1}] \geq 1/K \geq \tilde{E}[\theta_{m,K}] \tag{20}$$

By differentiating $\tilde{E}[\theta_{m,k}]$ with respect to ω_m^2

$$\frac{\partial}{\partial(\omega_m^2)} \tilde{E}[\theta_{m,k}] = -\frac{1}{\left[1 + \sum_{k' \neq k} \exp(c_{m,kk'})\right]^2} \times \left[\sum_{k' \neq k} \exp(c_{m,kk'})(o_{m,k} - o_{m,k'}) a (1 + 2a\omega_m^2)^{-\frac{3}{2}}\right] \tag{21}$$

where $a = 0.368$ according to (Daunizeau, 2017). When $k = 1$, Eq (21) is negative, indicating that as the aleatoric uncertainty increases, the upper bound in $E[\theta_{m,k}]$ is decreasing and goes towards $1/K$. When $k = K$. Eq (21) is positive, indicating that as the aleatoric uncertainty increases, the lower bound of $E[\theta_{m,k}]$ is increasing and goes towards $1/K$. □

The detailed proof is provided in Appendix B.

3.5 Model Updates with User Annotations

During network updates during feedback iterations, we propose to fix the mainstream encoder and decoder. This is because the mainstream encoder-decoder architecture is responsible for performing coarse segmentation in a global manner. The corresponding knowledge is already learned during pretraining. In contrast, the style-aware augmentation module is responsible for making augmentation so that the segmentation prediction is customized for different style patterns. After pretraining, new images with heterogeneous styles are provided to the model, and some images may be of brand new styles. In this case, it is proposed to update the module accordingly so that the model can adapt to new patterns, and the objective function is modified as

$$\arg \min_{q(\mathbf{z},\mathbf{v},\mathbf{W}^a)} L = KL[q(\mathbf{z})||p(\mathbf{z}|\mathbf{v})] + KL[q(\mathbf{v})||p(\mathbf{v})] + KL[q(\mathbf{W}^a)||p(\mathbf{W}^a)] - E_q[\ln p(\tilde{\mathbf{Y}}|\mathbf{z}, \mathbf{W}, \mathbf{W}^a, \mathbf{X})] \tag{22}$$

where $\tilde{\mathbf{Y}}$ denotes the combination of soft and user annotated labels. It should be noted that the $p(\mathbf{v})$, $p(\mathbf{z})$ and $p(\mathbf{W}^a)$ in (22) are different from those used for (11). In (22), we plug in the posterior distribution of $(\mathbf{v}, \mathbf{z}, \mathbf{W}^a)$ learned from pretraining as the prior distribution, so that the knowledge acquired from the pretraining process is used to regularize network updates.

The last term in (22) for soft labels can be interpreted from the perspective of KL divergence. Note that the soft label $\tilde{y}$ is essentially a distribution that describes the fuzzy label (*i.e.,* a pixel should belong to class k with probability $\tilde{y}_k$). Then the KL divergence between the fuzzy label and the predicted probability by the refined network is

$$KL(p(\tilde{y})\|p(\hat{y})) = \sum_k p(\tilde{y}_k)(\ln p(\tilde{y}_k) - \ln p(\hat{y}_k)) \tag{23}$$

If $\tilde{y}$ is one-hot, the KL divergence degenerates to the categorical cross-entropy.

In summary, there are three stages: i) The model needs to be pretrained to perform initial segmentation. Then, in one iteration of interactive segmentation, ii) the model makes an initial prediction, and iii) collects annotations from users to update parameters and refine the segmentation. At the first stage, all modules are trainable; at the second stage, only the pattern assignment parameter is trainable; at the third stage, the pattern assignment and residual blocks are trainable while other modules are frozen. The algorithm of the whole process is summarized in Appendix A.

4 EXPERIMENTS

In this section, we report our experimental results in two testing cases with real-world image datasets. The first case focuses on medical images. Algorithms are evaluated using the HAM dataset, which contains dermatoscopic images with seborrheic keratosis, melanoma, and benign tumors from different populations (Tschandl et al., 2018). The models are trained on the PH2 dermatoscopic images dataset (Mendonça et al., 2013) in the realm of pigmented lesions so that the model experiences diversified styles during the training and testing phases. The second case focuses on natural images. Algorithms are evaluated using ACDC (Sakaridis et al., 2021) containing street scenes in different weather conditions, including normal, light fog, and heavy fog, with pixel-level annotations of eight major semantic categories. The models are trained on the Cityscape dataset (Cordts et al., 2016) with urban scenes with normal weather conditions. To reduce potential overfitting, we perform data augmentation for data used for pretraining using cropping, flipping, image translation and modification of HSV channels. Adam optimizer is used for gradient-based optimization. For model updates, we set the number of iterations to 200. For model pre-training, the hyper-parameters are set to $\beta_0 = 1$, $\mu_0 = 0$, $\sigma_0 = 1$, $M = 10$.

Experimental setup. A group of college students participated as volunteers for evaluating the interactive system. As discussed in the introduction, users (i.e., participants) performed the annotation task in the following steps. 1) Given a testing image, the model makes initial segmentation prediction; 2) The user provides annotation by clicking on a few pixels with wrong segmentation and typing the correct class labels into a pop-up textbox; 3) After receiving user feedback, a refined segmentation is provided by the model and compared with the ground-truth of all pixels for evaluation. For both cases, 80 images are randomly selected from the testing data for evaluation. It should be noted that user study is time-consuming and difficult to perform on a massive scale. We provide instructions to the participants as follows: 1) If applicable, the user should avoid annotating on pixels that are overly close to the boundaries of segments; 2) If applicable, the user should annotate pixels over a large area of the image rather than focusing on a small region.

For evaluating the effectiveness of an interactive system, one important aspect is the quality of refined results given a limited annotation budget. For interactive segmentation, one widely-applied option of annotation budget is the number of clicks provided by users (Mahadevan et al., 2018). Users provide annotations through an interface that includes a pop-up window for visualizing initial segmentation and collecting clicks, and a textbox for typing in the corresponding labels. The user interface is shown in Appendix D. Users provide click-based annotations in a progressive way, and the number of clicks ranges from 1 to 16 for all the baselines. We report the quality of refined segmentation in terms of mean intersect over union (IoU) averaged across testing images and classes given different numbers of clicks.

Baselines. We include baseline methods MLG (Majumder and Yao, 2019), SU (Lin et al., 2016), SPS (Bearman et al., 2016) and BRS (Sofiiuk et al., 2020).

These baselines are representative methods that allow user interactions through clicks, which is the same as our experiment settings. MLG generates binary foreground-background interaction maps and distance-based maps based on user annotations and feeds them to the network for refining segmentation. For the ACDC dataset with multiple classes, the foreground-background maps are extended to the multiclass setting where user annotations are converted into multiclass guidance maps rather than foreground-background maps. BRS considers backpropagation refinement to adjust misclassified pixels that are inconsistent with user annotations. SPS leverages point supervision with a weighting factor that quantifies the relative importance of each supervised point. SU leverages a graphical model that propagates information from annotations to unmarked pixels to update network parameters.

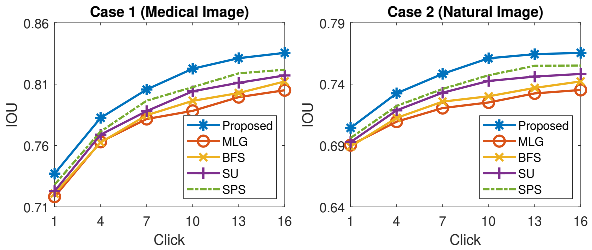


Figure 4: Quantitative comparison of refined segmentation (IoU) with respect to the number of interactions

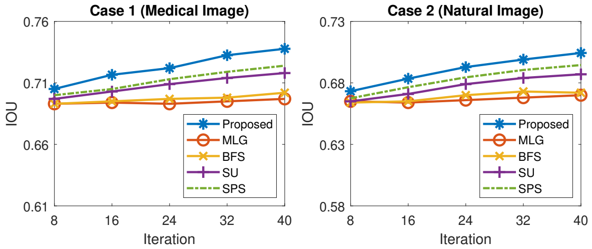


Figure 5: Performance comparison of initial segmentation (IoU) with respect to the number of existing iterations. An upward trend indicates that the model learns from previous iterations of interaction to improve its performance on new images.

Performance comparison. We first present the results of performance improvement with different numbers of clicks in Figure 4. In most cases, the proposed framework outperforms the baselines given the same number of clicks. MLG does not update network parameters after pre-training, and thus its capability to adapt to user annotations is limited. SU and SPS involve the update of network parameters and auxiliary feature maps and perform better. However, they are not specifically developed with information filtering, and thus its performance may deteriorate if the overfitting issue emerges.

Qualitative comparisons with illustrative examples of refined segmentation after 16 clicks are provided in Figure 6. The proposed framework usually achieves better performance in identifying boundaries of segments and visually difficult instances.

Another important perspective is to evaluate how the model leverages user annotation to improve the performance on future segmentation tasks. The key motivation is that the model should ideally learn from users so that it can perform better in the future, even without user annotation. Recall that the interaction process takes new images one by one. Therefore, we report the performance of initial segmentation on new images in terms of IoU after the model interacts with users for a number of iterations (*i.e.,* each iteration corresponds to a new image, and the model leverages user annotation on the image to update its parameters). We compare with baselines and report the results in Figure 5. The proposed framework exhibits an upward trend and outperforms baselines. It indicates that the performance is improved by learning from user annotation, and the information filtering strategy effectively extracts useful knowledge from user annotation while down-weighting noisy information.

A side product of the proposed framework is latent style pattern discovery in an unsupervised setting. We consider the street scene images as illustrative examples, where the weather condition is regarded as an important factor of style patterns. These images correspond to diverse weather conditions, including normal, light fog, and heavy fog. During the inference phase, by optimizing (15), the proposed framework automatically assigns latent style patterns to each image. By grouping images with respect to the pattern assignment, we observe some interpretable patterns with illustrative examples shown in the top two rows of Figure 7. The first pattern can be interpreted as heavy fog, and the second can be interpreted as light fog, while the rest two patterns correspond to normal weather with minor variants. Similarly, for dermatoscopic images, we visualize some illustrative examples as shown in the bottom two rows of Figure 7. The second pattern can be interpreted as excessive hairs, the third pattern can be interpreted as low contrast between lesion and healthy skin, while the first and forth pattern corresponds to moderate contrast.

Ablation Study. We also conduct an ablation study to compare with alternative designs of the proposed framework. The proposed method uses style-aware augmentation to assign latent style patterns (*i.e.,* the first-level information filtering strategy) to new images to improve the segmentation performance. In addition, it uses uncertainty-aware label softening (*i.e.,* the second-level information filtering strategy) to generate

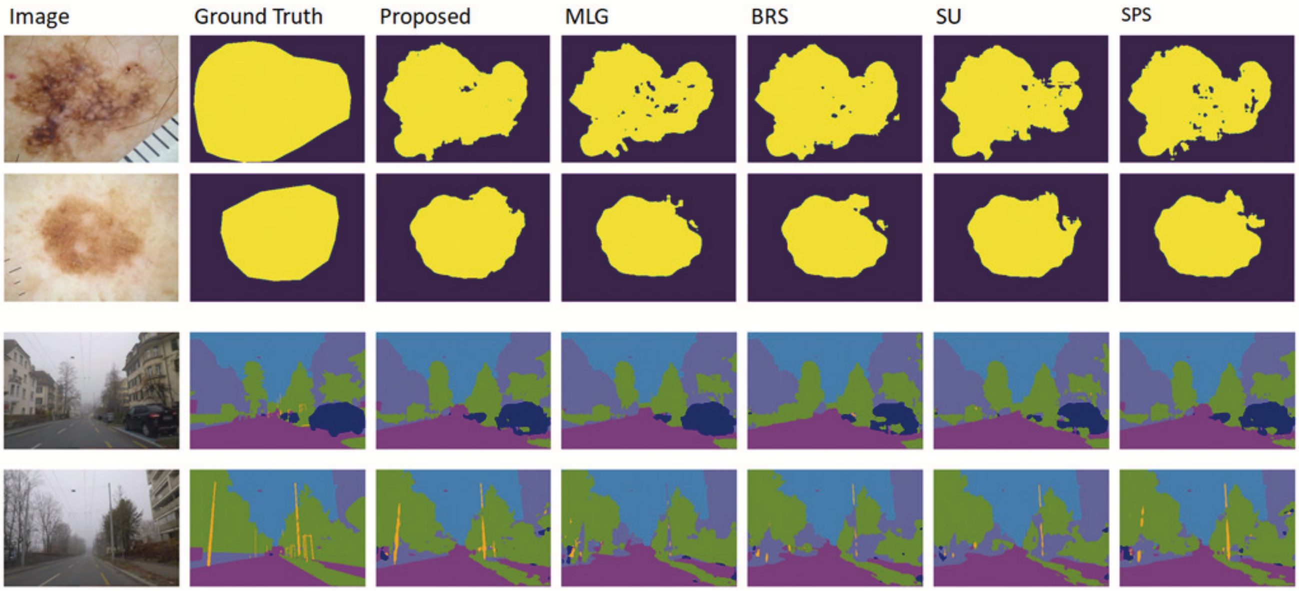


Figure 6: Illustrative examples of refined segmentation results after user interaction with 16 clicks. Different color denote different semantic classes.

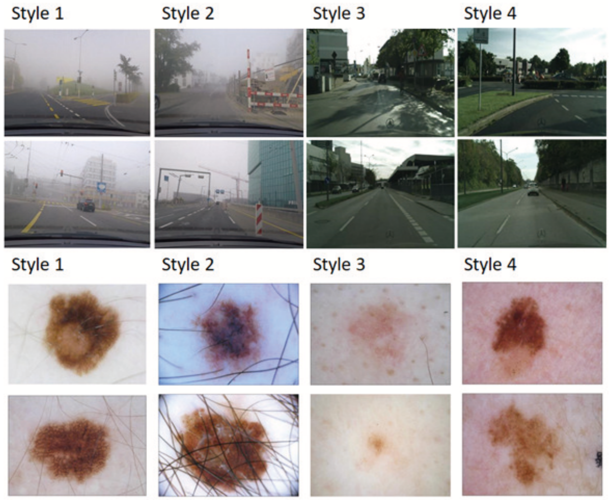


Figure 7: Illustrative examples of images automatically assigned to different latent style patterns

soft labels based on initial segmentation prediction, which is then combined with user annotation to retrain the model and generate refined results. Alternative design choices are to remove either or remove both (*i.e.,* plain architecture). We compare the proposed design with those three alternative approaches, and report results in Figure 8. The results are summarized in terms of refined segmentation (IoU) with different numbers of clicks. And the results are averaged across testing images. It can be seen that both alternative approaches underperform the proposed design.

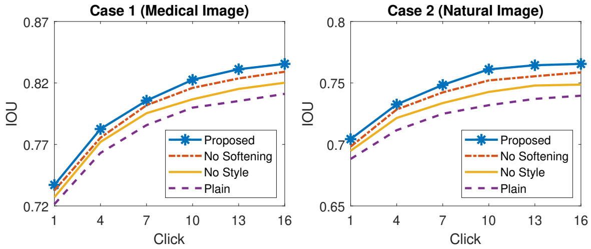


Figure 8: Ablation study. Results are reported in terms of refined segmentation (IoU) on two datasets with respect to the number of interactions.

Due to the page limit, we provide additional experiment results on winter scene image segmentation in Appendix E.

5 CONCLUSION

We propose an adaptive segmentation framework to support user interaction while introducing dual-level information filtering to train a robust model. The first-level information filtering disentangles the style and content of the image and customizes the segmentation prediction for latent image styles. The second-level information filtering applies label softening by leveraging uncertainty information to generate soft labels and integrate them with user annotations for effective model updates. The proposed framework may find its potential application for interactive segmentation tasks in specialized domains, such as medicine, security intelligence, and autonomous driving.

Acknowledgements

This research was partially supported by NSF IIS award IIS-1814450 and ONR award N00014-18-1-2875. Any opinions, findings, and 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.

References

Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

Amy Bearman, Olga Russakovsky, Vittorio Ferrari, and Li Fei-Fei. What's the point: Semantic segmentation with point supervision. In *European conference on computer vision*, pages 549–565. Springer, 2016.

David M Blei and Michael I Jordan. Variational inference for dirichlet process mixtures. *Bayesian analysis*, 1(1):121–143, 2006.

Lluis Castrejon, Kaustav Kundu, Raquel Urtasun, and Sanja Fidler. Annotating object instances with a polygon-rnn. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5230–5238, 2017.

Chengliang Chai and Guoliang Li. Human-in-the-loop techniques in machine learning. *Data Engineering*, 37:16, 2020.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Jean Daunizeau. Semi-analytical approximations to statistical moments of sigmoid and softmax mappings of normal variables. *arXiv preprint arXiv:1703.00091*, 2017.

Ashis Kumar Dhara, Kalyan Ram Ayyalasomayajula, Erik Arvids, Markus Fahlström, Johan Wikström, Elna-Marie Larsson, and Robin Strand. Segmentation of post-operative glioblastoma in mri by u-net with patient-specific interactive refinement. In *International MICCAI Brainlesion Workshop*, pages 115–122. Springer, 2018.

Aram Galstyan and Paul R Cohen. Empirical comparison of "hard" and "soft" label propagation for relational classification. In *International Conference on Inductive Logic Programming*, pages 98–111. Springer, 2007.

Charley Gros, Andreanne Lemay, and Julien Cohen-Adad. Softseg: Advantages of soft versus binary training for image segmentation. *Medical Image Analysis*, 71:102038, 2021.

Xianxu Hou, Jingxin Liu, Bolei Xu, Bozhi Liu, Xin Chen, Mohammad Ilyas, Ian Ellis, Jon Garibaldi, and Guoping Qiu. Dual adaptive pyramid network for cross-stain histopathology image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 101–109. Springer, 2019.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Won-Dong Jang and Chang-Su Kim. Interactive image segmentation via backpropagating refinement scheme. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5297–5306, 2019.

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

Theodora Kontogianni, Michael Gygli, Jasper Uijlings, and Vittorio Ferrari. Continuous adaptation for interactive object segmentation by learning from corrections. *arXiv preprint arXiv:1911.12709*, 2019.

Ponnambalam Kumaraswamy. A generalized probability density function for double-bounded random processes. *Journal of hydrology*, 46(1-2):79–88, 1980.

Hang Li, Dong Wei, Shilei Cao, Kai Ma, Liansheng Wang, and Yefeng Zheng. Superpixel-guided label softening for medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 227–237. Springer, 2020.

Di Lin, Jifeng Dai, Jiaya Jia, Kaiming He, and Jian Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3159–3167, 2016.

Zhendong Liu, Xin Yang, Rui Gao, Shengfeng Liu, Haoran Dou, Shuangchi He, Yuhao Huang, Yankai Huang, Huanjia Luo, Yuanji Zhang, et al. Remove appearance shift for ultrasound image segmentation via fast and universal style transfer. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pages 1824–1828. IEEE, 2020.

Chunwei Ma, Zhanghexuan Ji, and Mingchen Gao. Neural style transfer improves 3d cardiovascular mr image segmentation on inconsistent data. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 128–136. Springer, 2019.

Sabarinath Mahadevan, Paul Voigtlaender, and Bastian Leibe. Iteratively trained interactive segmentation. *arXiv preprint arXiv:1805.04398*, 2018.

Soumajit Majumder and Angela Yao. Content-aware multi-level guidance for interactive instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11602–11611, 2019.

Teresa Mendonça, Pedro M Ferreira, Jorge S Marques, André RS Marcal, and Jorge Rozeira. Ph 2-a dermoscopic image database for research and benchmarking. In *2013 35th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pages 5437–5440. IEEE, 2013.

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10765–10775, 2021.

Konstantin Sofiiuk, Ilia Petrov, Olga Barinova, and Anton Konushin. f-brs: Rethinking backpropagating refinement for interactive segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8623–8632, 2020.

Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5(1):1–9, 2018.

Guotai Wang, Wenqi Li, Maria A Zuluaga, Rosalind Pratt, Premal A Patel, Michael Aertsen, Tom Doel, Anna L David, Jan Deprest, Sébastien Ourselin, et al. Interactive medical image segmentation using deep learning with image-specific fine tuning. *IEEE transactions on medical imaging*, 37(7):1562–1573, 2018a.

Guotai Wang, Maria A Zuluaga, Wenqi Li, Rosalind Pratt, Premal A Patel, Michael Aertsen, Tom Doel, Anna L David, Jan Deprest, Sébastien Ourselin, et al. Deepigeos: a deep interactive geodesic framework for medical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 41(7): 1559–1572, 2018b.

Xingjiao Wu, Luwei Xiao, Yixuan Sun, Junhang Zhang, Tianlong Ma, and Liang He. A survey of human-in-the-loop for machine learning. *arXiv preprint arXiv:2108.00941*, 2021.

Ning Xu, Brian Price, Scott Cohen, Jimei Yang, and Thomas S Huang. Deep interactive object selection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 373–381, 2016.

Minqing Zhang, Jiantao Gao, Zhen Lyu, Weibing Zhao, Qin Wang, Weizhen Ding, Sheng Wang, Zhen Li, and Shuguang Cui. Characterizing label errors: Confident learning for noisy-labeled image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 721–730. Springer, 2020.

Bowei Zhou, Li Chen, and Zhao Wang. Interactive deep editing framework for medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 329–337. Springer, 2019.

Supplementary Material: Dual-Level Adaptive Information Filtering for Interactive Image Segmentation

Organization. In this Appendix, we first summarize the workflow and algorithm of the proposed framework, along with major mathematical notations in Appendix A. We formally prove how a high aleatoric uncertainty can make the predicted labels 'softer' under the proposed framework in Appendix B. We provide additional discussion of related concepts in Appendix C. We then provide additional details of experiments in Appendix D.

A Summary of Workflow, Algorithm and Notations

Table 1: Summary of Important Notations

h^l	output feature map of layer l
$\mathbf{W}^l$	layer l's weights
w_i^l	i-th entry of layer l's weights
(μ_i^l, σ_i^l)	parameter of posterior distribution of w_i^l
v_m, c_m	auxiliary variables of the stick-breaking representation of Dirichlet process for style pattern m
$\mathbf{z}_n$	style pattern assignment vector for image n
$z_{n,m}$	m-th entry of vector $\mathbf{z}_n$
(a_m, b_m)	parameter of posterior distribution of v_m
ζ_n	parameter of posterior distribution of $\mathbf{z}_n$
$\boldsymbol{\phi}$	all weights of the network
$\mathbf{W}$	all weights of convolutional blocks in the mainstream encoder-decoder
$\mathbf{W}^{\mathbf{a}}$	all weights of residual blocks in the style-aware augmentation module
$q(\cdot)$	variational distribution to approximate the posterior of variables
$\mathbf{X}$	images
$\mathbf{Y}$	ground-truth segmentation labels for all pixels
$\hat{\mathbf{Y}}$	initial prediction of segmentation labels for all pixels
$\hat{\theta}_m, \hat{\zeta}_m$	predicted softmax probability vector and logit vector for pixel m's label
(o_m, ω_m^2)	predicted mean and variance of pixel m's logit
$\tilde{y}_m$	soft label for pixel m
$\tilde{\mathbf{Y}}$	the combination of soft labels on unannotated areas and user-provided labels on annotated areas to update the network

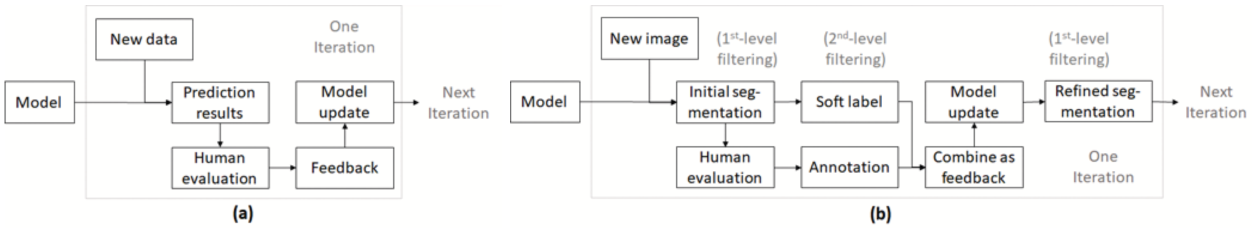
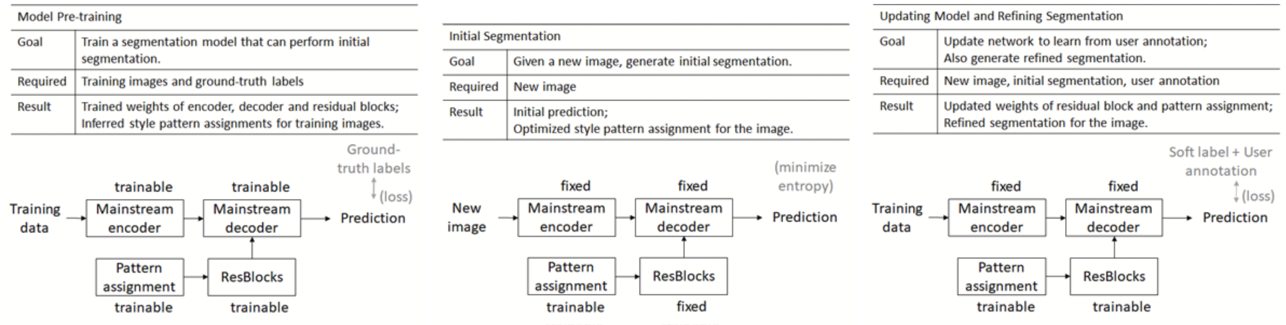


Figure 9: The workflow of general human-in-the-loop model training (a) and the workflow of the proposed interactive segmentation framework (b).

We first summarize the workflow of our framework and make a connection to the general human-in-the-loop model training. With humans involved in model training, a general workflow (Chai and Li, 2020) is described in Figure 1 (a). When applied to segmentation tasks, we customize the workflow as shown in Figure 9 (b). There are three adjustments from the general human-in-the-loop model training: 1) human annotation for segmentation is on a few pixels rather than all pixels on an image; 2) After retraining the model for updating parameters, the model makes a prediction on the same image as a refined segmentation; 3) The proposed information filtering strategies are involved: the first-level filtering aims to leverage style patterns to generate a better segmentation, while the second-level filtering aims to generate soft labels based on the initial prediction, while is combined with the human annotation to update the model.



Model Pre-training	
Goal	Train a segmentation model that can perform initial segmentation.
Required	Training images and ground-truth labels
Result	Trained weights of encoder, decoder and residual blocks; Inferred style pattern assignments for training images.

Initial Segmentation	
Goal	Given a new image, generate initial segmentation.
Required	New image
Result	Initial prediction; Optimized style pattern assignment for the image.

Updating Model and Refining Segmentation	
Goal	Update network to learn from user annotation; Also generate refined segmentation.
Required	New image, initial segmentation, user annotation
Result	Updated weights of residual block and pattern assignment; Refined segmentation for the image.

Figure 10: Illustration of the three stages including model pre-training, initial segmentation and model update.

We visualize the three stages in Figure 10 and summarize the key steps in Algorithms 1, 2 and 3.

Algorithm 1 Model Pretraining

```
Require: Training images $\mathbf{X}$ and ground-truth labels $\mathbf{Y}$;
Given hyperparameters $b_0$, $\mu_0$, $\sigma_0$;
for epoch $e = 1 : maxEpoch$ do
    Sample weights of mainstream encoder-decoder $\mathbf{W}$ and weights of residual blocks $\mathbf{W}^a$ using Eq.(2)
    Sample auxiliary variable $\mathbf{v}$ using Eq.(4)
    for images $n = 1 : N$ do
        Sample latent pattern assignment $z_n$ using Eq.(6)
        Feed forward using Eqs.(2), (8) to generate predicted probability for all pixels
        Evaluate loss using Eqs.(11)-(14)
        Backpropagate via stochastic gradient descent to update posterior parameters of $q(\mathbf{W})$, $q(\mathbf{W}^a)$, $q(\mathbf{v})$ and latent pattern assignment $q(z_n)$
    end for
end for
```

Algorithm 2 Initial Segmentation

```
Require: A new image $X_n$ and pretrained model;
Import parameters learned from pretraining $q(\mathbf{W})$, $q(\mathbf{W}^a)$, $q(\mathbf{v})$ and fix them;
for epoch $e = 1 : maxEpoch$ do
    Sample weights of mainstream encoder-decoder $\mathbf{W}$ and weights of residual blocks $\mathbf{W}^a$ using Eq.(2)
    Sample auxiliary variable $\mathbf{v}$ using Eq.(4)
    for images $n = 1 : N$ do
        Sample latent pattern assignment $z_n$ using Eq.(6)
        Feed forward using Eqs.(2), (8) to generate predicted probability for all pixels
        Evaluate predictive entropy using Eq.(15)
        Backpropagate to optimize latent pattern assignment $q(z_n)$
    end for
end for
Feed forward using Eqs.(2), (8) to generate predicted probability for all pixels as initial segmentation
```

B Proof of Lemma 1

In this section, we formally show the relationship between aleatoric uncertainty and the 'softness' of labels: *a large aleatoric uncertainty leads to softer labels, indicating a less uncertain prediction.* An important tool we leverage is the approximation of the expected softmax values.

Proof. First, we sort entries in o_m in an descending order so that $o_{m,1} \geq o_{m,2} ... \geq o_{m,K}$. Notice that when

Algorithm 3 Updating Model and Refining Segmentation

```
Require: Initial prediction of segmentation
Predict mean and variance of pixel-wise logits using Eq.(16)
Generate pixel-wise soft label using Eq.(17)
Collect user annotations and replace the soft label on annotated pixels with user-provided labels;
for $epoche = 1 : maxEpoch$ do
    Sample weights of mainstream encoder-decoder $\mathbf{W}$ and weights of residual blocks $\mathbf{W}^a$ using Eq.(2)
    Sample auxiliary variable $\mathbf{v}$ using Eq.(4)
    for images $n = 1 : N$ do
        Sample latent pattern assignment $z_n$ using Eq.(6)
        Feed forward using Eqs.(2), (8) to generate predicted probability for all pixels
        Evaluate loss using Eqs.(11)-(14)
        Backpropagate to update posterior parameters of $q(\mathbf{W}^a)$, $q(\mathbf{v})$ and latent pattern assignment $q(z_n)$
    end for
end for
Feed forward using Eqs.(2), (8) to generate predicted probability for all pixels as refined segmentation
Proceed to another new image
```

variance is zero, ζ is degenerated to point estimate. The corresponding softmax value is

$$\tilde{y}_m = \left[\frac{\exp\left(o_{m,1}\right)}{\sum_k \exp\left(o_{m,k}\right)}, \frac{\exp\left(o_{m,2}\right)}{\sum_k \exp\left(o_{m,k}\right)} \cdots \frac{\exp\left(o_{m,K}\right)}{\sum_k \exp\left(o_{m,k}\right)} \right] \tag{24}$$

When the variance is a finite positive number, the expectation $E[\theta_m]$ does not have an exact analytical form. We use a semi-analytical approximation [3], which is derived by matching the statistical moments. Specifically, the k-th element of $E[\theta_m]$ can be approximated by

$$E[\theta_{m,k}] \approx \left[2 - K + \sum_{k' \neq k} \frac{1}{E[\text{sigm}(\zeta_{m,k} - \zeta_{m,k'})]} \right]^{-1} \tag{25}$$

where $E[\text{sigm}(\zeta_{m,k} - \zeta_{m,k'})]$ is the expectation of sigmoid of $\zeta_{m,k} - \zeta_{m,k'}$. Since the sum of two Gaussian variables is still Gaussian, we have

$$(\zeta_{m,k} - \zeta_{m,k'}) \sim N(o_{m,k} - o_{m,k'}, 2(\omega_m)^2) \tag{26}$$

According to (Daunizeau, 2017), the expectation of sigmoid term can be further approximated as

$$E[\text{sigm}(\zeta_{m,k} - \zeta_{m,k'})] \approx \text{sigm}\left(\frac{o_{m,k} - o_{m,k'}}{\sqrt{1 + 2a\omega^2}} \right) \tag{27}$$

where $a = 0.368$.

Using the substitution $c_{m,kk'} = \frac{o_{m,k} - o_{m,k'}}{\sqrt{1+2a\omega^2}}$, we have

$$E[\theta_{m,k}] \approx \tilde{E}[\theta_{m,k}] = \frac{1}{1 + \sum_{k' \neq k} \exp(c_{m,kk'})} \tag{28}$$

It is straightforward to show that

$$\forall k_1 > k_2 : \quad \tilde{E}[\theta_{m,k_1}] \geq E[\theta_{m,k_2}] \tag{29}$$

And using counterevidence, we can show that

$$\tilde{E}[\theta_{m,1}] \geq \frac{1}{K} \geq \tilde{E}[\theta_{m,K}] \tag{30}$$

To investigate how this approximation changes with ω, we differentiate $\tilde{E}[\theta_{m,k}]$ with respect to (ω_m^2)

$$\begin{aligned} \frac{\partial}{\partial(\omega_m^2)} \tilde{E}[\theta_{m,k}] \approx & -\frac{1}{[1 + \sum_{k' \neq k} \exp(c_{m,kk'})]^2} \\ & \times \left[\sum_{k' \neq k} \exp(c_{m,kk'})(o_{m,k} - o_{m,k'}) a (1 + 2a\omega_m^2)^{-\frac{3}{2}} \right] \end{aligned} \tag{31}$$

Now we consider $k = 1$. Since $\forall k' \neq 1, \quad o_{m,1} \geq o_{m,k'}$, (31) is negative, indicating that as the aleatoric uncertainty increases, the upper bound in $E[\theta_{m,k}]$ is decreasing and goes towards $1/K$. Similarly, we consider $k = K$. Since $\forall k' \neq K, \quad o_{m,K} \leq o_{m,k'}$, (31) is positive, indicating that as the aleatoric uncertainty increases, the lower bound of $E[\theta_{m,k}]$ is increasing and goes towards $1/K$. □

C Additional Discussion of Related Concepts

In this section, we discuss some related concepts by explaining those concepts and clarifying the differences from the proposed model. The related concepts include 1) pattern discovery via probabilistic mixture models; 2) weakly-supervised semantic segmentation; 3) style disentanglement via domain adaptation, and the discussion is summarized in Table 2.

D Additional Details of Experiments

The proposed method and baselines are trained with Intel Core i7-3820 CPU and NVIDIA GeForce RTX2070 GPU. As discussed in the main paper, for medical images, the model is pretrained on the PH2 dermatoscopic image dataset and then evaluated on the HAM dataset. For natural images, the model is pretrained on the Cityscape dataset and then evaluated on the FoggyACDC dataset. This setting is common for style-aware segmentation and domain adaptation to ensure that the model is exposed to diversified image styles during the training and testing phases.

Once the model is pretrained, it is evaluated on interaction tasks where the user provides annotation as supervision to refine the segmentation and update the network. Given a new image, the model makes an initial prediction, and the user provides annotations on selective pixels. Based on both the user annotation and the initial segmentation, the network is updated to generate refined segmentation results and learns from user annotation. Illustrative examples of the interaction process are provided in Figure 11.

The interaction process takes new images one by one. Theoretically, different image orders will affect the model performance, because the model is updated based on new images and the corresponding user interactions. However, we do not observe over-sensitivity in practice. In addition, we randomly sample unseen images from testing data, and the reported results are averaged across images. Therefore, the influence of image order is considered negligible.

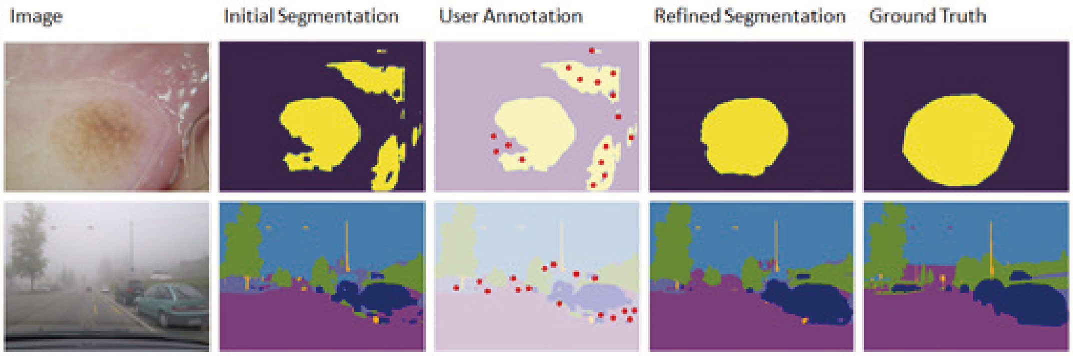


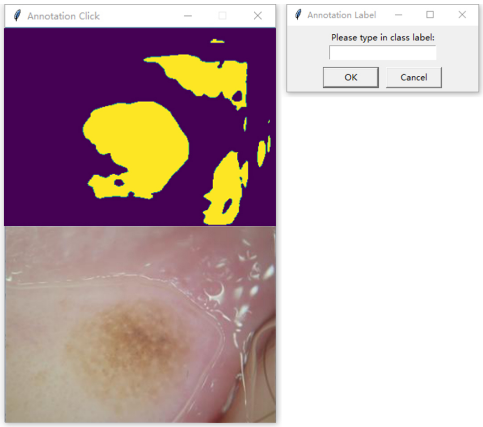
Figure 11: Illustration of the interactive segmentation process. A user provides annotation based on initial segmentation, and annotated pixels are highlighted in red. Once user annotation is collected, the network is updated and a refined segmentation is predicted. It should be noted that the complete ground truth labels are not available to the model; it is used only for evaluating the segmentation performance.

User interaction is implemented through a simple user interface, with an illustrative example shown in Figure 12. The user interface includes a pop-up window visualizing the initial segmentation results and the corresponding image. It allows users to click on the segmentation map to specify the position of annotation. After that, a textbox is generated for the user to type in the corresponding label of the clicked position. Then, the information of annotation is passed back to the model for refined segmentation results, and the network is updated accordingly to encode user knowledge.

Table 2: Discussion of Related Concepts

Concept	**Explanation of concept**	**Difference from the proposed model**
Pattern discovery via probabilistic mixture model	Probabilistic mixture models assume the data is generated from a mixture of patterns. Consider the Gaussian mixture model as an example. The data generation process includes 1) sampling mixture components from prior Gaussian distribution; 2) sampling mixture assignment for a data instance from prior Categorical distribution; 3) given mixture assignment and mixture component, generate the observation of the data instance. In this case, the pattern is the mixture component, characterized by the corresponding Gaussian distribution. Given new data, the pattern assignment is inferred by maximizing the log-likelihood. Gaussian mixture model can be extended to non-parametric models by using a Dirichlet process prior.	Our model assumes latent styles can be grouped into a number of patterns. However, we do not assume the images are generated from a mixture of style patterns. Instead, we assume each style pattern requires unique augmentation to feature maps so that the final segmentation prediction is customized to the style. Given an image, the forward process includes 1) sampling layer weights from augmentation module from prior distribution; 2) sampling latent pattern assignment for an image; 3) given the weights and pattern assignment, generating the customized segmentation. Without user annotation, the pattern assignment of an image is inferred by minimizing predictive entropy. With user annotation, the pattern assignment is inferred by maximizing the likelihood.
Weakly-supervised segmentation	In weakly-supervised segmentation, the model is trained with noisy or limited labels as supervised signals. For example, given some images, the corresponding label used for training could be the label annotation of a few pixels through clicks, scribbles, or bounding boxes. This is different from conventional image segmentation, where the labels of all pixels are available. Weakly-supervised segmentation can be conducted interactively by collecting annotation from users.	During pretraining, our model leverages datasets with ground truth of fully labeled pixels rather than partial labeled ones. During user interactions, the new images provided by the user are from another dataset with distribution shifts and diverse styles. At this phase, only partial labels from annotated areas by the user are used to retrain the model. Our setting is realistic, because one can always find some fully labeled datasets to pretrain the model. But if we customize the model to specific applications, fully labeling the data may be costly, and thus leveraging partial annotation from the user is more viable.
Foreground-background interactive segmentation	In foreground-background segmentation, the user annotates foreground and background classes to determine the target segment. For medical image segmentation, it is similar to semantic segmentation, because the foreground corresponds to disease areas, and the background class corresponds to normal areas.	Our framework focuses on semantic segmentation, and it requires the user to provide labels in terms of semantic class rather than foreground and background. User annotations are treated as supervision to update the model and generate refined segmentation.
Style disentanglement via domain adaptation	Domain adaptation aims to apply the segmentation model trained in data from one source domain to data from another target domain, where distribution shifts exist between different domains. Image style is one of the sources of distribution shift.	Our model considers more fine-grained styles. For instance, multiple latent styles may be discovered from the new images provided by the user. In contrast, domain adaptation essentially considers two 'styles', one from the source domain and one from the target domain.

The experiment results reported in the main paper are based on real-user interactions. These studies received *IRB approval* that strictly protected the rights of the subjects.



Please type in class label:

Figure 12: An illustrative example of the user interface

E Additional Experiment Results

For a more comprehensive evaluation of the proposed model, we provide an additional evaluation of the ACDC dataset for winter scene image segmentation. Similar to before, the models are pre-trained on the Cityscape dataset with urban scenes, and evaluated on the ACDC dataset.

We first compare the performance given different numbers of clicks. For each image, after receiving a number of clicks (with corresponding label annotation), the proposed model and baseline methods refine the segmentation. The results averaged across testing images are reported in Figure 13. In most cases, the proposed framework outperforms the baselines given the same number of clicks.

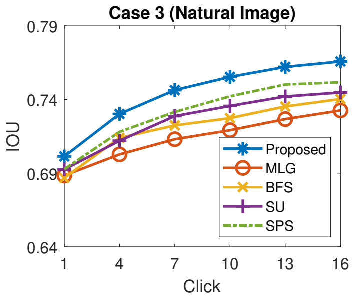


Figure 13: Performance comparison of refined segmentation (IoU) on ACDC dataset with respect to the number of clicks (averaged across testing images)

It should be noted that the baseline methods do not consider diversified image styles. To overcome this limitation, we apply the image style transfer technique proposed by [6] on testing images, and evaluate the baseline methods again as shown in the right plot of Figure 13. The goal of style transfer is to make the testing image's distribution aligned with the training images. In most cases, the proposed framework outperforms the baselines given the same number of clicks.

Qualitative comparisons with illustrative examples of refined segmentation after 16 clicks are provided in Figure 14. The proposed framework achieves better performance in identifying boundaries of segments.

We also evaluate how the model leverages user annotation to improve the performance on future segmentation tasks. We report the performance of initial segmentation in terms of IoU after the model interacts with users for a few iterations. We compare with baselines and report the results on Figure 15. The proposed framework exhibits an upward trend and outperforms baselines.

F Link to the Source Code

The source code is provided at: https://github.com/ritmininglab/DLAIF

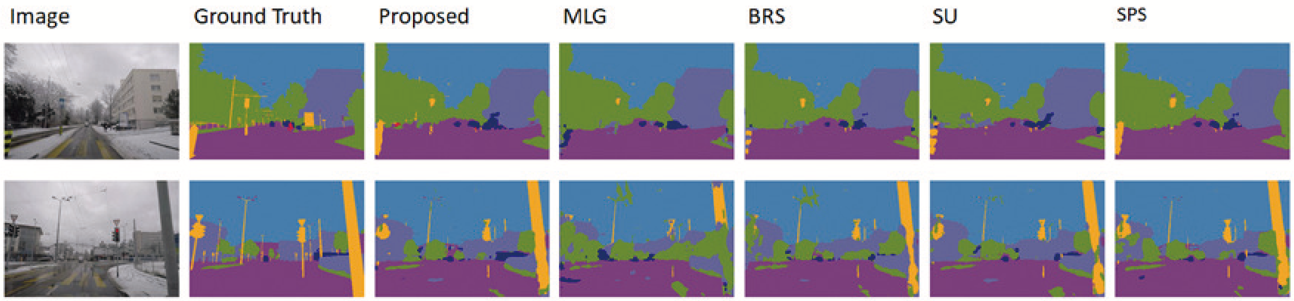


Figure 14: Illustrative examples of refined segmentation results after user interaction with 16 clicks. Different color denote different semantic classes.

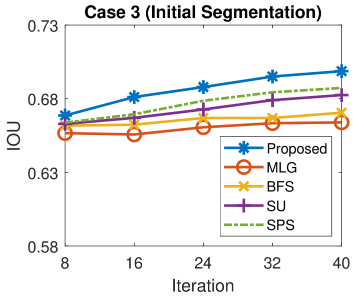


Figure 15: Performance comparison of initial segmentation (IoU) on ACDC dataset with respect to the number of existing iterations. An upward trend indicates that the model learns from previous iterations to improve its performance on new images.

Reference

Chai, C., & Li, G. (2020). Human-in-the-loop Techniques in Machine Learning. Data Engineering, 37.

Daunizeau, J. (2017). Semi-analytical approximations to statistical moments of sigmoid and softmax mappings of normal variables. arXiv preprint arXiv:1703.00091.

Liu, Z., Yang, X., Gao, R., Liu, S., Dou, H., He, S., ... & Ni, D. (2020). Remove appearance shift for ultrasound image segmentation via fast and universal style transfer. In 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI) (pp. 1824-1828). IEEE.

Li, H., Wei, D., Cao, S., Ma, K., Wang, L., & Zheng, Y. (2020). Superpixel-Guided Label Softening for Medical Image Segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 227-237). Springer, Cham.