Active Learning for Image Segmentation with Binary User Feedback

Debanjan Goswami Shayok Chakraborty
Department of Computer Science, Florida State University

Abstract

Deep learning algorithms have depicted commendable performance in a variety of computer vision applications. However, training a robust deep neural network necessitates a large amount of labeled training data, which is time-consuming and labor-intensive to acquire. This problem is even more serious for an application like image segmentation, as the human oracle has to hand-annotate each and every pixel in a given training image, which is extremely laborious. Active learning algorithms automatically identify the salient and exemplar samples from large amounts of unlabeled data, and tremendously reduce human annotation effort in inducing a machine learning model. In this paper, we propose a novel active learning algorithm for image segmentation, with the goal of further reducing the labeling burden on the human oracles. Our framework identifies a batch of informative images, together with a list of semantic classes for each, and the human annotator merely needs to answer whether a given semantic class is present or absent in a given image. To the best of our knowledge, this is the first research effort to develop an active learning framework for image segmentation, which poses only binary (yes/no) queries to the users. We pose the image and class selection as a constrained optimization problem and derive a linear programming relaxation to select a batch of (image-class) pairs, which are maximally informative to the underlying deep neural network. Our extensive empirical studies on three challenging datasets corroborate the potential of our method in substantially reducing human annotation effort for real-world image segmentation applications.

1. Introduction

Semantic segmentation (labeling every pixel in an image with its category) is one of the core tasks of visual recognition and is extensively used in a variety of applications, including autonomous driving, medical imaging and video surveillance among others [14]. With the advent and popularity of deep learning, several deep architectures have been studied for image segmentation, which have depicted state-of-the-art results [47] [45] [27]. However, for these models to work reliably, a large amount of training data (in the form of pixel-level annotated images) are required, which involves significant time and human labor. Thus, an algorithm to reduce human annotation effort is critically important to train deep learning models for image segmentation applications.

Active Learning (AL) algorithms identify the most informative samples from vast amounts of unlabeled data [37]. This tremendously reduces the human annotation effort in training a machine learning model, as only the samples that are selected by the algorithm need to be labeled manually. Further, since the model gets trained on the exemplar samples from the data, it typically depicts better generalization performance than a passive learner, where the training data is sampled at random. AL has been successfully used in a variety of applications, including computer vision [44], text analysis [40], bioinformatics [30] and medical diagnosis [16] among others. The growing popularity of deep learning has motivated research in the field of *deep active learning*, to efficiently train the data-hungry deep learning models [34].

The paucity of human labor and the need to use it more efficiently is even more pronounced for an application like image segmentation, due to the enormous time and effort associated with labeling every pixel in an image. This necessitates specialized query and annotation mechanisms for the AL algorithms to be feasible in a real-world setting. In this paper, we propose a novel AL algorithm to address this challenging problem, in an effort to alleviate the labeling burden on human oracles [1] while inducing a deep learning model for image segmentation. Our algorithm queries a batch of (image-class) pairs and for each pair, poses the question: "*Does the image i contain the semantic class j?*" [2] The human annotator merely has to provide a binary "*yes / no*" feedback for each query. This is depicted in Figure 1.

Providing such feedback is extremely easy and less prone to annotation errors; it is also significantly less time-consuming and burdensome than providing pixel-level annotations. Our contributions in this paper can be summa-

[1] the terms *user*, *annotator*, *oracle* and *labeler* are used interchangeably in this paper

[2] the term *class* is used to mean *semantic class* in this paper

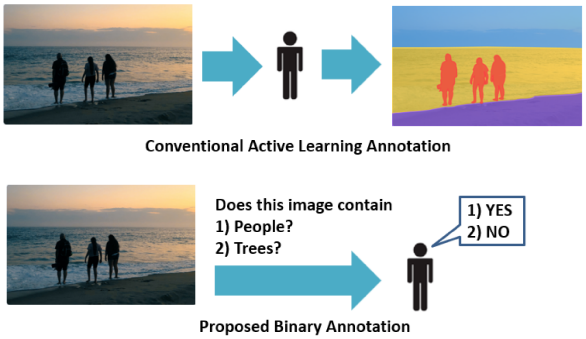


Figure 1. Figure showing the conventional active learning query (top) and the proposed binary query mechanism (down). Best viewed in color.

rized as follows:

- We present a novel AL framework for image segmentation, which poses only binary ("*yes / no*") queries regarding the presence / absence of a semantic class in a given image. To our knowledge, this is the first active learning framework for semantic image segmentation which poses only binary queries to the human annotators.
- We pose the image and class selection as a constrained optimization problem, and derive a linear programming relaxation to select a batch of (image-class) pairs, which are maximally informative to the underlying deep neural network.
- We conduct user studies to estimate the time and human effort required to annotate an image at the pixel-level, region-level and binary-level (our method). This can provide valuable insights and enable us to study the trade-off between the human annotation effort and the generalization capability of the trained deep neural network, for different types of annotation strategies.
- We conduct extensive empirical studies on three benchmark datasets to study the performance of our framework against competing baselines.

2. Related Work

Active Learning: AL is a well-researched problem in the machine learning community [37]. Uncertainty sampling is the most common strategy for active learning, where unlabeled samples with the highest prediction uncertainties are queried for annotation [25] [21] [11] [18]. The growing success and popularity of deep learning have motivated researchers to explore the problem of deep active learning (DAL), where the goal is to select the informative unlabeled samples to train a deep neural network [34]. Common DAL techniques include learning a loss prediction module to predict the loss value of an unlabeled sample and querying samples accordingly [44], selecting informative unlabeled samples for AL and simultaneously, searching for the best neural architectures on-the-fly [13], a sampling technique based on diverse gradient embeddings [2], a technique which captures the information balance between the uncertainty of underlying softmax probability and the label variable and queries samples accordingly [41] and a technique to select a *Coreset* of samples, such that the model learned over the selected subset is competitive for the remaining data points [36]. Techniques based on adversarial learning have depicted particularly impressive performance in this context [39] [29] [46].

Beyond the conventional label query, a body of research in AL has focused on the development of novel query and annotation mechanisms to further reduce the labeling burden on human users. Binary feedback mechanism has been studied, where the active learner queries a pair of images, and the human annotator has to specify whether or not the two images belong to the same category [21] [12]. In another variant, the learner queries an unlabeled image together with a class label, and the human annotator has to specify whether the selected image belongs to that class [19] [4]. Along similar lines, AL has been exploited in clustering, where a pair of samples is queried and the oracles need to specify whether or not the samples in a pair correspond to the same cluster [5]. Although the query mechanism is binary, these methods query the label of an image as a whole, and not the presence of a semantic class within an image, and hence are not directly applicable to the problem of image segmentation.

Active Learning for Image Segmentation: Providing pixel-level annotations to train an image segmentation model is a time-consuming and expensive process. To address this challenge, weakly supervised semantic segmentation techniques have been developed, such as providing the presence or absence of classes in an image during training [33, 42], pointing to an object of interest [3], bounding box annotations [31], free-form squiggles [26] and noisy web tags [1]. However, these methods utilized the weak supervision only during model training (as a term in the training loss function) and did not use AL to identify the informative images or the semantic classes within an image.

As in conventional AL, uncertainty and representativeness based metrics have been exploited for AL in the context of semantic segmentation [43]. Metrics like view-point entropy have been studied for multi-view datasets [38]. Desai and Ghose [10] proposed a semi-supervised AL technique which selects a representative set of unlabeled images for dense pixel-level annotation. The recent trend in this line of research is to identify the informative *regions* within an image and getting them annotated by the human label-

ers, rather than the entire image. Hwang *et al.* [20] recently proposed an AL framework which selects informative local image regions (based on an uncertainty and class balance criterion) and requests an oracle for a multi-hot vector annotation to indicate all the classes existing in the selected region. Kim *et al.* [23] recently proposed an adaptive superpixelization strategy and an uncertainty based acquisition function to identify the informative superpixels for manual annotation. Other strategies to select informative image regions for annotation include superpixel entropy [22], informativeness, combined with annotation cost and the spatial coherency of an image [28] and self-consistency under equivariant transformations [15]. Although annotating image regions is less strenuous than providing pixel-level annotations, it still requires the human oracles to meticulously label all the pixels in the queried regions, which can be quite time-consuming, particularly if the queried region involves multiple semantic classes. In contrast, our framework requests only binary feedback regarding the presence / absence of specific classes in an image, which requires much lesser annotation effort and facilitates an easier mode of interaction between the user and the system. We now describe our framework.

3. Proposed Framework

3.1. Problem Formulation

Consider an active image segmentation problem where we are given a labeled training set L and an unlabeled set U. Let N denote the number of unlabeled images, $N = |U|$. Images in L are provided with pixel-level annotations. Let w be the deep neural network trained on L, and C be the number of semantic classes in the dataset. We are given a query budget B and a parameter C_{max} which denotes the maximum number of classes that can be queried per image (to ensure that the queries are distributed across a large number of images). Our objective is to select a batch of images, together with a list of classes for each image for binary user query, such that the total number of queries does not exceed the budget B, and the user response about the presence/absence of the semantic classes augments maximal information to the deep learning model.

In order to identify the optimal set of images and semantic classes to be queried, we need a metric to quantify the utility score of a batch of (image-class) pairs. We used a criterion based on class presence uncertainty and image redundancy for this purpose. The first criterion ensures that we query those (image-class) pairs where there is maximal uncertainty regarding the presence of the given class in the given image; the redundancy criterion ensures that we query a diverse set of images in our batch and avoid duplicate image queries. Since our objective was to develop an AL framework for image segmentation with binary queries, and not to design a new AL sampling criterion, we used a criterion based on uncertianty and diversity due to its prior success in DAL research [34].

Computing Class Presence Uncertainty: Let p_{ij} denote the probability that image i contains the semantic class j (computed using the current deep neural network w, as the average probability of pixels belonging to the semantic class j within image i). We used Shannon's entropy to compute the prediction uncertainty of the presence of semantic class j in image i:

$$H_{ij} = -p_{ij} \log p_{ij} - (1 - p_{ij}) \log(1 - p_{ij}) \qquad (1)$$

Using this, we computed a confidence matrix $G \in \Re^{C \times N}$, where $G(j, i)$ denotes the confidence of the deep model in predicting the presence of class j in image i (high entropy corresponds to low confidence and vice versa):

$$G(j, i) = \frac{\alpha}{H_{ij}} \quad i = 1, \ldots N, \;\; j = 1, \ldots C \qquad (2)$$

where α is a constant.

Computing Image Redundancy: We computed a redundancy matrix $R \in \Re^{N \times N}$, where $R(i, j)$ denotes the redundancy between images x_i and x_j in the unlabeled set. The cosine similarity was used to quantify the redundancy between a pair of samples (due to its prior success in diversity computation in AL research [8]):

$$R(i, j) = cos(\mathcal{F}(x_i), \mathcal{F}(x_j)) \qquad (3)$$

where $cos(\mathcal{F}(x_i), \mathcal{F}(x_j)) = \frac{\mathcal{F}(x_i)^\top \mathcal{F}(x_j)}{||\mathcal{F}(x_i)||.||\mathcal{F}(x_j)||}$ and $\mathcal{F}(x)$ denotes the deep feature representation of image x. A low value of $R(i, j)$ implies that images x_i and x_j have low redundancy between them. All the values in G and R were mapped in the $(0, 1)$ range. Depending on the application, other metrics can also be used to compute the uncertainty and redundancy terms.

3.2. Active Sampling Framework

Given G and R, our objective is to query a batch of (image-class) pairs such that in each pair, the deep model has low confidence in predicting the presence of the given class in the given image, and the queried images have minimal redundancy among them. We define a binary matrix $M \in \{0, 1\}^{N \times C}$, where each row corresponds to an unlabeled image and each column corresponds to a semantic class. A value of 1 in a row denotes that the image should be selected for annotation, and the position(s) of 1 in a particular row of M denote the semantic class(es) that should be used to pose the binary queries for this image. We also define a binary vector $v \in \{0, 1\}^{N \times 1}$ where $v_i = 1$ denotes that image x_i is selected for annotation, and $v_i = 0$ denotes that it is not selected. The active selection of (image-class)

pairs can thus be posed as the following optimization problem:

$$\begin{aligned} \min_{M,v} \quad & \mathrm{Tr}(MG) + \lambda v^\top Rv \\ s.t. \quad & \langle M, E \rangle = B \\ & (M.e)_i \leq C_{max}, \forall i \\ & v_i = \min(1, (M.e)_i), \forall i \\ & v_i, M_{ij} \in \{0, 1\}, \forall i, j \end{aligned} \tag{4}$$

where $\lambda > 0$ is a weight parameter governing the relative importance of the two terms, E is a matrix of size $N \times C$ (same size as M) with all entries 1, e is a vector of size $C \times 1$ with all entries 1, B is the labeling budget, $\langle ., . \rangle$ denotes the inner product operator and Tr denotes the trace of a matrix. The first term in the objective function denotes that the deep model has low confidence in predicting the presence of the selected semantic classes in the corresponding selected images; the second term ensures that the selected images have minimal redundancy among them. The first constraint denotes the total number of queries posed by M is equal to the specified budget; the second constraint ensures that the number of 1s in each row of M is less than or equal to C_{max}, that is, the number of queries posed for each image is less than or equal to the pre-specified limit C_{max}; the third constraint denotes that v_i is equal to 1 if there is at least one entry with value 1 in row i of M (image x_i is selected for annotation), and v_i is equal to 0 if all the entries in row i of M have value 0 (image x_i is not selected); the fourth constraint denotes that v is a binary vector and M is a binary matrix. We now present a theorem to solve this optimization problem.

Theorem 1. *The optimization problem defined in Equation (4) can be expressed as an equivalent linear programming (LP) problem.*

Proof. We simplify the definition of v in the third constraint and rewrite the optimization problem as:

$$\begin{aligned} \min_{M,v} \quad & \mathrm{Tr}(MG) + \lambda v^\top Rv \\ s.t. \quad & \langle M, E \rangle = B \\ & (M.e)_i \leq C_{max}, \forall i \\ & M_{ij} \leq v_i, \forall i, j \\ & v_i, M_{ij} \in \{0, 1\}, \forall i, j \end{aligned} \tag{5}$$

The constraint $M_{ij} \leq v_i, \forall i, j$ denotes that if row i in M has at least one entry as 1, then v_i has to be 1. If row i in M has all entries as 0, then v_i is free to be 0 or 1. However, we are solving a minimization problem with $v^\top Rv$ in the objective, and R has only non-negative entries; this criterion will force v_i to be equal to 0, as that will result in a better (lower) value of the objective. This shows that the constraint $v_i = \min(1, (M.e)_i), \forall i$ in Equation (4) is equivalent to the linear constraint $M_{ij} \leq v_i, \forall i, j$ in Equation (5).

The first term in the objective function can be expressed as a linear term: $\mathrm{Tr}(MG) = \sum_{i,j} G_{ij}.M_{ji}$. Also, let $z_{ij} = v_i.v_j$. Clearly, Z is a binary matrix of size $N \times N$ with all entries 0 or 1. The second term in the objective can then be written as:

$$v^\top Rv = \sum_{i,j} z_{ij}.r_{ij} \tag{6}$$

The optimization problem can thus be expressed as:

$$\begin{aligned} \min_{M,v,Z} \quad & \sum_{i,j} G_{ij}.M_{ji} + \lambda \sum_{i,j} z_{ij}.r_{ij} \\ s.t. \quad & \sum_{i,j} M_{ij} = B \\ & z_{ij} = v_i.v_j, \forall i, j \\ & (M.e)_i \leq C_{max}, \forall i \\ & M_{ij} \leq v_i, \forall i, j \\ & v_i, M_{ij}, Z_{ij} \in \{0, 1\}, \forall i, j \end{aligned} \tag{7}$$

Now, we attempt to express the quadratic equality $z_{ij} = v_i.v_j, \forall i, j$ as a linear term. The quadratic equality implies that z_{ij} equals 1 only when both v_i and v_j are 1 and equals 0 otherwise. This can be expressed as the linear inequality $v_i + v_j \leq 1 + 2z_{ij}, \forall i, j$. From the inequality, we note that when both v_i and v_j are 1, z_{ij} is forced to be 1. When v_i and v_j are both 0, or one of them is 0 and the other one is 1, z_{ij} is free to be 0 or 1. Using the same argument as before, we note that we are solving a minimization problem with $\sum_{i,j} z_{ij}.r_{ij}$ in the objective and R has only non-negative entries; thus, the nature of the problem will force z_{ij} to be 0 as it will produce a lower value of the objective. Replacing the quadratic equality with the linear inequality, we express the optimization problem as follows:

$$\begin{aligned} \min_{M,v,Z} \quad & \sum_{i,j} G_{ij}.M_{ji} + \lambda \sum_{i,j} z_{ij}.r_{ij} \\ s.t. \quad & \sum_{i,j} M_{ij} = B \\ & v_i + v_j \leq 1 + 2z_{ij}, \forall i, j \\ & (M.e)_i \leq C_{max}, \forall i \\ & M_{ij} \leq v_i, \forall i, j \\ & v_i, M_{ij}, Z_{ij} \in \{0, 1\}, \forall i, j \end{aligned} \tag{8}$$

In this optimization problem, both the objective function and the constraints are linear in the variables M, v and Z. It is thus a linear programming (LP) problem.

□

We vectorize the variables M, v and Z, append them one below the other and express the objective function and the constraints in terms of this new variable. We relax the integer constraints into continuous constraints and solve the problem using an off-the-shelf LP solver. After obtaining the continuous solution, we recover the integer solution using a rounding approach where the B highest entries in M are reconstructed as 1 and the other entries as 0, observing the constraints. The pseudo-code of our algorithm is depicted in Algorithm 1 (for one active learning iteration).

Algorithm 1 The Proposed Active Learning Algorithm with Binary User Feedback

Require: Labeled training set L, unlabeled set U, query budget B, parameters α, C_{max} and λ, a deep neural network architecture for image segmentation

1: Train the deep model on the training set L
2: Compute the confidence matrix G using the probabilities of the trained deep model (Equation (2))
3: Compute the redundancy matrix R (Equation (3))
4: Solve the LP problem derived from Equation (4) after relaxing the integer constraints (details provided in the Appendix)
5: Round the solution to derive the matrix M
6: Select the unlabeled images and the corresponding semantic classes to pose the binary queries based on the entries in M
7: Update the deep model with the user response to the binary queries (detailed in Section 4.4)

4. Experiments and Results

4.1. Datasets

We used three challenging datasets to study the performance of our framework: (i) **Flickr-Landscapes** [32]; (ii) **Cityscapes** [9]; and (iii) **PASCAL VOC12** [17]. All these are benchmark datasets commonly used to validate the performance of image segmentation algorithms.

4.2. Comparison Baselines

We used a total of five methods as comparison baselines in our work. *These baselines were selected to cover a range of annotation techniques (pixel-level, region-level and binary-level), since the fundamental contribution of this research is a novel annotation technique for AL in image segmentation.*

Pixel-level annotation: In this category, a batch of unlabeled images were queried and all the pixels of all the queried images were annotated. We used *SSAL* [10], a recently proposed semi-supervised AL algorithm which selects a representative set of unlabeled images for annotation, as the comparison baseline in this category.

Region-level annotation: Here, a batch of regions were queried from the unlabeled images and all the pixels in the queried regions were annotated. We used two very recently proposed techniques as comparison baselines in this category: *PixBal* [20], which queries image regions based on an uncertainty and class balance criterion, and *AMSP+S* [23], which uses an adaptive merging strategy and an uncertainty based acquisition function to identify the informative superpixels for annotation.

Binary-level annotation: In this category, binary queries were posed regarding the presence / absence of specific semantic classes in the unlabeled images (similar to our method). The proposed algorithm is the first AL framework with binary-level annotation for image segmentation; we hence used the following methods as comparison baselines: *Random-Random (RR)*, which randomly selects a subset of images and randomly queries B semantic classes from the selected images; and *Entropy-Entropy (EE)*, where a batch of images were selected based on the entropy of the underlying model and the semantic classes producing the highest prediction entropy values were queried from each.

4.3. Experimental Setup

Each dataset was divided into three parts: (i) an initial training set L; (ii) an unlabeled set U; and (iii) a test set. The number of images in the initial training, unlabeled and test sets were 500, $1,200$ and $1,000$ respectively for all three datasets. All the images in L were provided with pixel-level annotations. A query budget B (taken as 200 for Cityscapes and PASCAL and 400 for Flickr) was imposed in each AL iteration, and the experiments were conducted for 25 AL iterations. The query budget denotes the number of binary queries that can be posed (for the binary-level annotation methods, *RR*, *EE* and our method) or the number of image regions that can be queried (for the region-level annotation methods, *PixBal* and *AMSP+S*). However, since we had $1,200$ images in our unlabeled set, using a query budget of 200 for the pixel-level annotation baselines would have exhausted the unlabeled pool after 6 AL iterations. We hence set the query budget to $48 (= 1200/25)$ in each AL iteration for the pixel-level baseline *SSAL*, so that the unlabeled pool is completely exhausted after 25 AL iterations.

After each AL iteration, the selected samples were annotated and appended to the training set; the deep neural network was retrained and tested on the test set. The objective was to study the improvement in performance on the test set with increasing number of label queries. The value of α in Equation (2) was set as 1, the parameter C_{max} in Equation (4) was taken as 5, and the weight parameter λ in Equation (4) was taken as 1 for all the datasets. All the results were averaged over 3 runs (with different training, unlabeled and

Annotation Task	Flickr		Cityscapes		PASCAL	
	Time	**Ease**	**Time**	**Ease**	**Time**	**Ease**
Pixel-level	7.8 ± 2.9mins	5.5 ± 1.2	37.5 ± 6.3mins	3.6 ± 1.6	18.2 ± 4.3mins	5.2 ± 1.7
Region-level	1.6 ± 1.2mins	7.3 ± 2.7	3.6 ± 0.7mins	5.5 ± 1.8	2.7 ± 1.1mins	6.7 ± 2.3
Binary-level	2 ± 0.3secs	10 ± 0.0	4 ± 0.8secs	10 ± 0.0	3 ± 1.4secs	10 ± 0.0

Table 1. User study results. The table reports the average time and ease of annotation to annotate **one** complete image at the pixel-level, **one** region within an image at the pixel-level, and to answer **one** binary query posed for a given image, for the three datasets. The results were averaged across all images for a given dataset and all annotators.

test sets) to rule out the effects of randomness.

Base Model: We used the *DeepLabV3+* model with the ResNet101 backbone (pre-trained on ImageNet) as our base model due to its promising performance in image segmentation applications [6]. The same architecture was used for all the baseline methods, for fair comparison.

Evaluation Metrics: The **mean intersection-over-union (mIoU)** was used as the evaluation metric, as commonly done in image segmentation research [6]. Since our comparison baselines span different categories of annotation, we also used the **human annotation time / effort** as an evaluation metric.

4.4. Implementation Details

Model Updating with Binary User Annotations: Training a deep neural network for image segmentation with only image-level annotations (whether a given semantic class is present / absent in an image) has been explored in previous research [33] [42]. We used the idea proposed by Pinheiro and Collobert [33] to update the DeepLabV3+ model with the binary user annotations obtained through AL. *Note that, these methods proposed an algorithm for only image segmentation, and not an active learning algorithm for image segmentation with binary user feedback, which is unexplored, and is our fundamental contribution in this work.*

4.5. User Study to Estimate Human Annotation Time / Effort

To accurately estimate the human annotation time (and hence, effort) required to annotate an image at the pixel-level, region-level and binary-level, we conducted a user study. 10 images were selected at random from each of the three datasets. For each image, the following tasks were posed:

(i) Annotators were asked to segment each image at the pixel level with the different categories of objects and mark each category with a different color (pixel-level annotation)

(ii) Annotators were asked to annotate all the pixels within a given region (super-pixel) of an image with the different categories of objects and mark each category with a different color (region-level annotation)

(iii) Annotators were asked a question regarding the presence of an object in each image and had to provide a binary response: "YES / NO" (binary-level annotation)

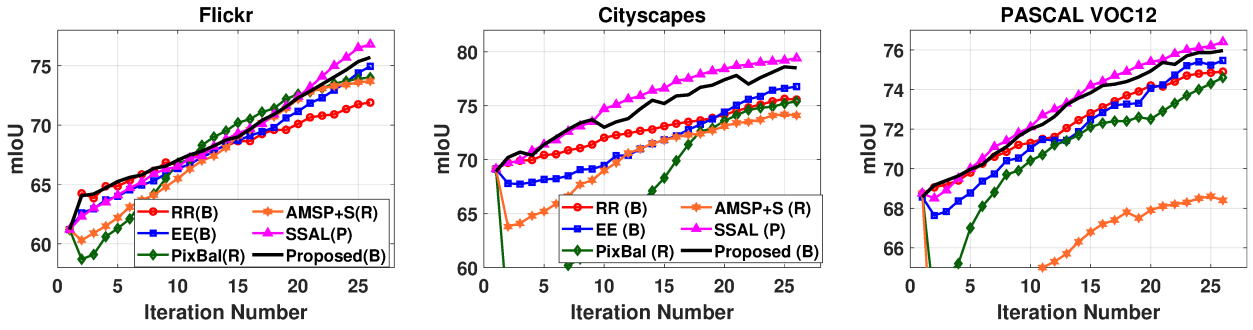
Annotators were provided with the *LabelMe* annotation tool [35] to segment the images. The time taken for each annotation task was noted. The annotators were also asked to provide a rating, denoting the ease of annotation for each task, on a scale of 1 to 10, 1 being VERY DIFFICULT and 10 being VERY EASY. Each image was annotated (at the pixel, region, and binary levels) by 3 human annotators independently.

The user-study results are reported in Table 1, which depicts the average time and ease of annotation across all images and annotators, for the three datasets. The resolution of each image was 513×513 for Flickr, 768×768 for Cityscapes and 513×513 for PASCAL VOC. As evident from the table, pixel-level annotation entailed the maximum amount of time (and hence, human labor). Annotating a given region within an image took considerably less amount of time. As expected, binary-level annotations were the most efficient in terms of time and took only a few seconds for each image. We also note that the pixel-level annotations were the most difficult to provide, followed by region-level annotations. All the annotators reported that binary annotations were the easiest and the most convenient to provide and it consistently received the highest rating of 10. This user study demonstrates the tremendous savings in human annotation effort that can be achieved using the proposed binary-level annotation technique for image segmentation applications.

4.6. Active Learning Performance

The active learning performance results are shown in Figure 2. In each graph, the x-axis denotes the iteration number and the y-axis denotes the mean IoU on the test set.

mIoU Analysis: The proposed method comprehensively outperforms the two other AL techniques that utilize binary user feedback: *RR* and *EE*. In almost all the iterations across all three datasets, our framework depicts a better mIoU value compared to these two baselines. The final mIoU achieved by our method after 25 AL iterations is also higher than *RR* and *EE*, for all three datasets. This shows that our algorithm can successfully identify the exemplar (image-class) pairs which augment maximal information to the deep learning model, thereby enabling it to attain much better generalization capabilities. The *PixBal* and *AMSP+S*



(a) Flickr (b) Cityscapes (c) PASCAL

Figure 2. Active Learning performance comparison. The x-axis denotes the iteration number and the y-axis denotes the mean IoU on the test set. Query budget = 200 for Cityscapes and PASCAL and 400 for Flickr in each AL iteration. Here, **B** denotes binary-level annotation, **R** denotes region-level annotation and **P** denotes pixel-level annotation. Best viewed in color.

Dataset	RR(B)	EE(B)	PixBal(R)	AMSP+S(R)	SSAL(P)	Proposed(B)
Flickr	71.9 ± 0.44	74.95 ± 0.21	74.0 ± 0.41	73.7 ± 0.58	76.8 ± 0.57	75.7 ± 0.13
Cityscapes	75.56 ± 0.37	76.76 ± 0.25	75.4 ± 1.48	74.1 ± 0.07	79.4 ± 0.52	78.5 ± 0.47
PASCAL	74.9 ± 0.17	75.46 ± 0.35	74.6 ± 0.23	68.4 ± 0.17	76.4 ± 0.29	75.96 ± 0.06

Table 2. Final mIoU achieved by all the methods after 25 AL iterations. Here, **B** denotes binary-level annotation, **R** denotes region-level annotation and **P** denotes pixel-level annotation.

methods require human annotators to annotate image regions; however, the proposed method also consistently outperforms both these methods and depicts a faster growth in the mIoU values with increasing AL iterations. Both these methods use an algorithm to compute the superpixels of a given image; the informative superpixels are then queried for manual annotation. However, as mentioned in [23], the generated superpixels can be inaccurate and noisy, which can potentially affect the final segmentation result. Our method, on the other hand, does not involve any superpixels and poses a binary query regarding the presence / absence of a semantic class in an image, and avoids these issues. The *SSAL* method (which requires human users to annotate all the pixels in a given image) marginally outperforms the proposed algorithm.

Table 2 shows the final mIoU attained by all the methods after 25 AL iterations. We note that the proposed method outperforms the binary-level annotation methods (*RR* and *EE*) in terms of the final mIoU for all the datasets; it also outperforms the region-level annotation techniques (*PixBal* and *AMSP+S*) and achieves a higher final mIoU than these methods. The pixel-level annotation method *SSAL* attains a marginally higher final mIoU than our method; the maximum difference between the final mIoU achieved by our method and *SSAL*, across all the three datasets, is about 1.1%.

Human Annotation Time Analysis: Table 3 depicts an estimate of the total human annotation time (in hours) that has to be expended over the 25 AL iterations, for all the methods studied. These figures were obtained by multiplying the values in Table 1 by the number of annotations performed in each AL iteration and the total number of AL iterations. For instance, for the Cityscapes dataset, the time for pixel-level annotation was computed as: 37.5mins (time taken to annotate one image at the pixel-level) $\times$ 48 (no. of images annotated in each AL iteration) $\times$ 25 (no. of AL iterations); similarly, the time for region-level annotation was computed as: 3.6mins (time taken to annotate the pixels in one region in an image) $\times$ 200 (number of regions annotated in each AL iteration) $\times$ 25 (no. of AL iterations); and the time for the proposed binary annotation was computed as: 4secs (time taken to answer one binary query) $\times$ 200 (number of binary queries answered in each AL iteration) $\times$ 25 (no. of AL iterations).

Dataset	Binary-Level	Region-Level	Pixel-Level
Flickr	5.56	266.67	156
Cityscapes	5.56	300	750
PASCAL	4.16	225	364

Table 3. Approximate total time (**in hours**) to be expended for annotation (for the binary-level, region-level and pixel-level methods) over 25 AL iterations for all the three datasets. Query budget = 200 for Cityscapes and PASCAL and 400 for Flickr in each AL iteration. Query budget denotes the number of binary queries answered for binary-level annotation methods (*RR, EE, Proposed*), and number of image regions annotated for the region-level annotation methods (*PixBal* and *AMSP+S*). Pixel-level annotation method (*SSAL*) annotates all the $1,200$ unlabeled images at the pixel-level (48 images in each AL iteration for all the datasets).

From Figure 2 and Table 3, it is evident that our method

Backbone	RR(B)	EE(B)	PixBal(R)	AMSP+S(R)	SSAL(P)	Proposed(B)
Xception	72.6 ± 0.18	71.25 ± 0.63	72.3 ± 0.31	72.8 ± 0.38	73.5 ± 0.43	72.8 ± 0.11

Table 4. Final mIoU achieved by all the methods after 25 AL iterations for the Xception backbone. Here, **B** denotes binary-level annotation, **R** denotes region-level annotation and **P** denotes pixel-level annotation.

requires substantially less human annotation time and effort, while producing mIoU values that are better than the region-level annotation methods (*PixBal* and *AMSP+S*) and comparable to the pixel-level annotation method *SSAL*. For the PASCAL VOC dataset for instance, the final mIoU achieved by our binary query framework is 75.96, and the difference is less than 0.5% compared to the value achieved by *SSAL* (Table 2). However, the total annotation time required by *SSAL* is 87.5 times greater than our method (Table 3). Our method also requires 54.08 times lesser annotation time than *PixBal* and *AMSP+S* and outperforms them in terms of the mIoU achieved. These results corroborate the promise and potential of our binary query and annotation technique to substantially reduce human annotation effort, with only a marginal loss in performance, in an application like image segmentation, where annotating a single data instance is extremely time-consuming and laborious.

4.7. Study of Backbone Network Architecture

In this experiment, we studied the effect of the backbone network architecture used in the DeepLabV3+ model (we used ResNet-101 as the default backbone architecture). The results on the Cityscapes dataset (with query budget 200) using the XceptionNet [7] backbone are shown in Figure 3.

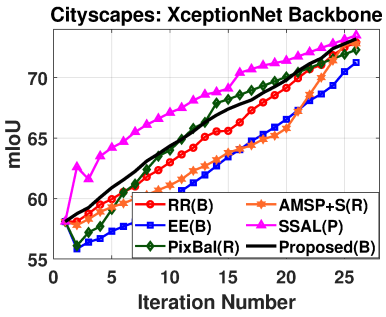


Figure 3. Study of backbone network architecture on the Cityscapes dataset. Query budget = 200. Best viewed in color.

mIoU Analysis: Our framework once again comprehensively outperforms the binary-level baselines (*RR* and *EE*) and the region-level baseline *AMSP+S*; it depicts comparable performance to the other region-level baseline, *PixBal*. The proposed method is marginally outperformed by the pixel-level baseline *SSAL*. The final mIoU achieved by all the method are reported in Table 4. *SSAL* achieved the highest value of 73.5, while our method achieved the second highest value of 72.8, together with *AMSP+S* (only 0.7% lower). This shows the robustness of our framework to the backbone network architecture.

Human Annotation Time Analysis: Since we have only changed the backbone network architecture in this experiment (and not the query budget), the total annotation time computed in Table 3 for the Cityscapes dataset is also applicable for this experiment. As evident from Table 3, the total annotation time required by the region-level (*PixBal* and *AMSP+S*) and pixel-level annotation (*SSAL*) methods are 53.95 times and 134.89 times greater than our method respectively. Our framework thus depicts impressive performance compared to the region-level and pixel-level annotation baselines, and is significantly more efficient in terms of the total human annotation time and effort compared to these baselines.

5. Conclusion and Future Work

In this paper, we proposed a novel active learning framework for semantic image segmentation, which poses only binary queries regarding the presence / absence of a semantic class in a given image. To the best of our knowledge, this is the first research effort to develop such an active query mechanism in the context of image segmentation. We posed the image and class selection as a constrained optimization problem and derived an LP relaxation to identify a batch of (image-class) pairs for active query. Our empirical results demonstrated the promise and potential of our framework to drastically reduce human annotation effort in training a deep neural network for semantic segmentation applications. We hope this research will motivate the development of novel AL algorithms, particularly for applications where labeling a single data instance involves significant manual work. As part of future research, we plan to explore GPU-based parallel algorithms (such as the one proposed in [24]) to improve the computational overhead of solving the LP problem.

6. Acknowledgment

This research was supported in part by the National Science Foundation under Grant Number: IIS-2143424 (NSF CAREER Award).

References

[1] E. Ahmed, S. Cohen, and B. Price. Semantic object selection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2

[2] J. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. In *International Conference on Learning Representations (ICLR)*, 2020. 2

[3] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. What's the point: Semantic segmentation with point supervision. In *European Conference on Computer Vision (ECCV)*, 2016. 2

[4] A. Bhattacharya and S. Chakraborty. Active learning with n-ary queries for image recognition. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 2

[5] A. Biswas and D. Jacobs. Active image clustering: Seeking constraints from humans to complement algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 2

[6] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision (ECCV)*, 2018. 6

[7] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 8

[8] C. Coleman, E. Chou, J. Katz-Samuels, S. Culatana, P. Bailis, A. Berg, R. Nowak, R. Sumbaly, M. Zaharia, and I. Yalniz. Similarity search for efficient active learning and search of rare concepts. In *AAAI Conference on Artificial Intelligence*, 2022. 3

[9] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5

[10] S. Desai and D. Ghose. Active learning for improved semi-supervised semantic segmentation in satellite images. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2022. 2, 5

[11] Yoav Freund, Sebastian Seung, Eli Shamir, and Naftali Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28(2-3):133–168, 1997. 2

[12] Y. Fu, B. Li, X. Zhu, and C. Zhang. Active learning without knowing individual instance labels: A pairwise label homogeneity query approach. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 26(4), 2014. 2

[13] Y. Geifman and R. El-Yaniv. Deep active learning with a neural architecture search. In *Neural Information Processing Systems (NeurIPS)*, 2019. 2

[14] S. Ghosh, N. Das, I. Das, and U. Maulik. Understanding deep learning techniques for image segmentation. *ACM Computing Surveys*, 52(4), 2020. 1

[15] A. Golestaneh and K. Kitani. Importance of self-consistency in active learning for semantic segmentation. In *British Machine Vision Conference (BMVC)*, 2020. 3

[16] M. Gorriz, A. Carlier, E. Faure, and X. Giro i Nieto. Cost-effective active learning for melanoma segmentation. In *Neural Information processing Systems (NeurIPS) Workshop*, 2017. 1

[17] B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik. Semantic contours from inverse detectors. In *IEEE International Conference on Computer Vision (ICCV)*, 2011. 5

[18] S. Hoi, R. Jin, J. Zhu, and M. Lyu. Batch mode active learning and its application to medical image classification. In *International Conference on Machine Learning (ICML)*, 2006. 2

[19] P. Hu, Z. Lipton, A. Anandkumar, and D. Ramanan. Active learning with partial feedback. In *International Conference on Learning Representations (ICLR)*, 2019. 2

[20] S. Hwang, S. Lee, H. Kim, M. Oh, J. Ok, and S. Kwak. Active learning for semantic segmentation with multi-class label query. In *Neural Information Processing Systems (NeurIPS)*, 2023. 3, 5

[21] A. Joshi, F. Porikli, and N. Papanikolopoulos. Breaking the interactive bottleneck in multi-class classification with active selection and binary feedback. In *CVPR*, 2010. 2

[22] T. Kasarla, G. Nagendar, G. Hegde, V. Balasubramanian, and C. Jawahar. Region-based active learning for efficient labeling in semantic segmentation. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 3

[23] H. Kim, M. Oh, S. Hwang, S. Kwak, and J. Ok. Adaptive superpixel for active learning in semantic segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 3, 5, 7

[24] J. Li, R. Lv, X. Hu, and Z. Jiang. A GPU-based parallel algorithm for large scale linear programming problem. In *Intelligent Decision Technologies*, 2011. 8

[25] X. Li and Y. Guo. Adaptive active learning for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[26] D. Lin, J. Dai, J. Jia, K. He, and J. Sun. Scribblesup: Scribble-supervised convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[27] Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong, F. Wei, and B. Guo. Swin transformer v2: Scaling up capacity and resolution. In *arXiv 2111.09883v1*, 2021. 1

[28] R. Mackowiak, P. Lenz, O. Ghori, F. Diego, O. Lange, and C. Rother. Cereals-cost-effective region-based active learning for semantic segmentation. In *British Machine Vision Conference (BMVC)*, 2018. 3

[29] C. Mayer and R. Timofte. Adversarial sampling for active learning. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020. 2

[30] H. Osmanbeyoglu, J. Wehner, J. Carbonell, and M. Ganapathiraju. Active machine learning for transmembrane helix prediction. *BMC Bioinformatics*, 11(1), 2010. 1

[31] G. Papandreou, L. Chen, K. Murphy, and A. Yuille. Weakly- and semi-supervised learning of a dcnn for semantic image segmentation. In *IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[32] T. Park, M. Liu, T. Wang, and J. Zhu. Semantic image synthesis with spatially-adaptive normalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 5

[33] P. Pinheiro and R. Collobert. From image-level to pixel-level labeling with convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 6

[34] P. Ren, Y. Xiao, X. Chang, P. Huang, Z. Li, B. Gupta, X. Chen, and X. Wang. A survey of deep active learning. *ACM Computing Surveys*, 54(9), 2021. 1, 2, 3

[35] B. Russell, A. Torralba, K. Murphy, and W. Freeman. Labelme: a database and web-based tool for image annotation. *International Journal of Computer Vision (IJCV)*, 2007. 6

[36] O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations (ICLR)*, 2018. 2

[37] B. Settles. Active learning literature survey. In *Technical Report 1648, University of Wisconsin-Madison*, 2010. 1, 2

[38] Y. Siddiqui, J. Valentin, and M. Nießner. Viewal: Active learning with viewpoint entropy for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[39] S. Sinha, S. Ebrahimi, and T. Darrell. Variational adversarial active learning. In *IEEE International Conference on Computer Vision (ICCV)*, 2019. 2

[40] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research (JMLR)*, 2:45–66, 2001. 1

[41] J. Woo. Active learning in bayesian neural networks with balanced entropy learning principle. In *International Conference on Learning Representations (ICLR)*, 2023. 2

[42] J. Xu, A. Schwing, and R. Urtasun. Tell me what you see and i will show you where it is. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2, 6

[43] L. Yang, Y. Zhang, J. Chen, S. Zhang, and D. Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2017. 2

[44] D. Yoo and I. Kweon. Learning loss for active learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2

[45] Y. Yuan, X. Chen, X. Chen, and J. Wang. Segmentation transformer: Object-contextual representations for semantic segmentation. In *European Conference on Computer Vision (ECCV)*, 2020. 1

[46] B. Zhang, L. Li, S. Yang, S. Wang, Z. Zha, and Q. Huang. State-relabeling adversarial active learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[47] Y. Zhu, K. Sapra, F. Reda, K. Shih, S. Newsam, A. Tao, and B. Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 1