ViewAL: Active Learning With Viewpoint Entropy for Semantic Segmentation

Yawar Siddiqui[1] Julien Valentin[2] Matthias Nießner[1]

[1]Technical University of Munich [2]Google

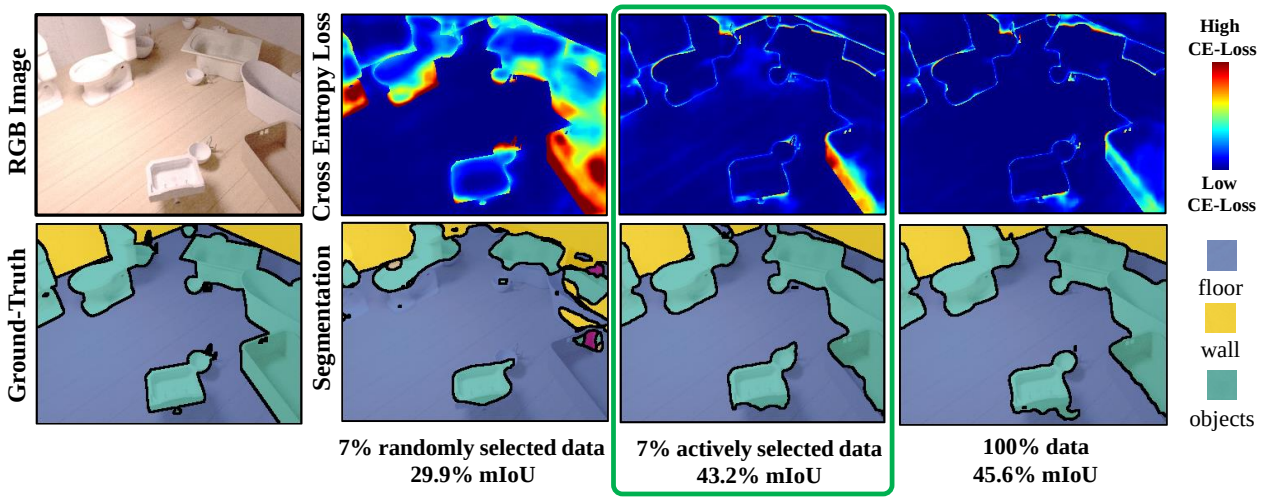


Figure 1: ViewAL is an active learning method that significantly reduces labeling effort: with maximum performance attained by using 100% of the data (last column), ViewAL achieves 95% of this performance with only 7% of data of SceneNet-RGBD [29]. With the same data, the best state-of-the-art method achieves 88% and random sampling (2nd column) yields 66% of maximum attainable performance.

Abstract

We propose ViewAL [1], a novel active learning strategy for semantic segmentation that exploits viewpoint consistency in multi-view datasets. Our core idea is that inconsistencies in model predictions across viewpoints provide a very reliable measure of uncertainty and encourage the model to perform well irrespective of the viewpoint under which objects are observed. To incorporate this uncertainty measure, we introduce a new viewpoint entropy formulation, which is the basis of our active learning strategy. In addition, we propose uncertainty computations on a superpixel level, which exploits inherently localized signal in the segmentation task, directly lowering the annotation costs. This combination of viewpoint entropy and the use of superpixels allows to efficiently select samples that are highly informative for improving the network. We demonstrate that our proposed active learning strategy not only yields the best-performing models for the same amount of required labeled data, but also significantly reduces labeling effort. Our method achieves 95% of maximum achievable network performance using only 7%, 17%, and 24% labeled data on SceneNet-RGBD, ScanNet, and Matterport3D, respectively. On these datasets, the best state-of-the-art method achieves the same performance with 14%, 27% and 33% labeled data. Finally, we demonstrate that labeling using superpixels yields the same quality of ground-truth compared to labeling whole images, but requires 25% less time.

[1]Source code available: https://github.com/nihalsid/ViewAL

1. Introduction

With the major success of deep learning on major computer vision tasks, such as image classification [23, 46, 51, 53], object detection [16, 10, 15, 35], pose estimation [58, 34, 20, 32], or semantic segmentation [26, 36, 1, 4, 63], both network sizes and the amount of data required to train these networks has grown significantly. This has led to a dras-

tic increase in the costs associated with acquiring sufficient amounts of high-quality ground truth data, posing severe constraints on the applicability of deep learning techniques in real-world applications. Active learning is a promising research avenue to reduce the costs associated with labeling. The core idea is that the system being trained actively selects samples according to a policy and queries their labels; this can lead to machine learning models that are trained with only a fraction of the data while yielding similar performance. Uncertainty sampling is one of the most popular strategies in active learning to determine which samples to request labels for [57, 14, 2, 27, 21, 54, 50]. Here, the model prefers samples it is most unsure about, based on an uncertainty measure, in order to maximize model improvement. Existing uncertainty sampling techniques almost exclusively operate on single images, which is surprising since many consumer-facing applications, such as robots, phones, or headsets, use video streams or multi-view data coming from 3D environments. As a consequence, geometric constraints inherently present in the real world are largely ignored, but we believe these are particularly interesting for examining the quality of network predictions; i.e., the same surface point in a scene should receive the same label when observed from different view points.

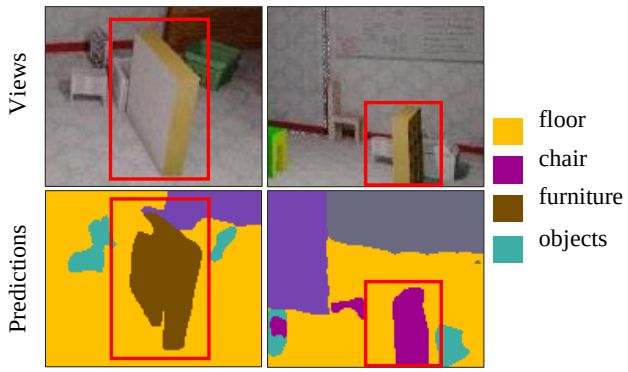


Figure 2: Inconsistencies in segmentations for two views of the same object (*furniture*). While in the first view the object is predicted to be *furniture*, in the second view it is predicted to be a *chair*.

In this work, we propose to exploit these constraints in a novel view-consistency-based uncertainty measure. More specifically, we propose a new viewpoint entropy formulation which is based on the variance of predicted score functions across multiple observations. If predictions for a given (unlabeled) object differ across views, we assume faulty network predictions; we then strive to obtain labels for the most uncertain data samples. In addition, we propose uncertainty computations on a superpixel level, which exploits inherently localized signal in segmentation tasks, directly lowering the annotation costs. This combination of viewpoint entropy and the use of superpixels allows efficient selection of samples that are highly informative for improving the network. Finally, we leverage sampling-based measures such as Monte Carlo (MC) dropout [14], and we show that in conjunction with these measures, we can further improve performance. In summary, our contributions are:

- A novel active learning strategy for semantic segmentation that estimates model uncertainty-based on inconsistency of predictions across views, which we refer to as *viewpoint entropy*.
- A *most informative view* criteria based on KL divergence of prediction probability distributions across view points.
- A superpixel-based scoring and label acquisition method that reduces the labeling effort while preserving annotation quality.

2. Related Work

A thorough review of classical literature on active learning can be found in Settles et al. [47]. As described in [61], given a pool of unlabeled data, there are three major ways to select the next batch to be labeled: uncertainty-based approaches, diversity-based approaches, and expected model change. In uncertainty-based approaches, the learning algorithm queries for samples it is most unsure about. For a probabilistic model in a binary classification problem, this would mean simply choosing the samples whose posterior probability of being positive is nearest to 0.5 [25, 24]. For more than two classes, entropy can be used as an uncertainty measure [48, 19]. A simpler way is to select instances with the least confident posterior probabilities [48]. Another strategy could be to choose samples for which the most probable label and second most probable label have least difference in prediction confidence [21, 37]. Yet another uncertainty-based approach is querying by committee where a committee of multiple models is trained on the labeled data, and unlabeled samples with least consensus are selected for labeling [50, 28].

Uncertainty-based approaches can be prone to querying outliers. In contrast, diversity-based approaches are designed around the idea that informative instances should be representative of the input distribution. Nguyen et. al [33] and Xu et al. [59] use clustering for querying batches. The last method of expected model change [49, 12, 22, 56] queries samples that would change the current model most if their labels were known. It has been successful for small models but has seen little success with deep neural networks because of computational complexity involved.

Quite a lot of the uncertainty-based approaches can be directly used with deep neural networks. Softmax probabilities have been used for obtaining confidence, margin,

and entropy measures for uncertainty [57]. Gal et al. [14] use multiple forward passes with dropout at inference time (Monte Carlo dropout) to obtain better uncertainty estimates. Ensemble methods [2, 6] have also been used to improve upon uncertainty estimates, however, these can be heavy in terms of memory and compute requirements. The loss learning approach introduced in [61] can also be categorized as an uncertainty approach. Sener et al. [44] propose a diversity-based approach, which formulates active learning as core-set selection - choosing a set of points such that a model learned over the selected subset is competitive for the remaining data points. Yang et al. [60] present a hybrid approach, using both uncertainty and diversity signals. They utilize uncertainty and similarity information from a DNN and formulate sample selection as generalized version of the maximum set cover problem to determine the most representative and uncertain areas for annotation.

While most of these methods [57, 2, 44] have been verified on classification tasks, they can be easily adapted to target segmentation. Sun et al. [52] investigate active learning for probabilistic models (e.g Conditional Random Fields) that encode probability distributions over an exponentially-large structured output space (for instance semantic segmentation). Active learning for semantic segmentation with deep neural networks has been specifically investigated in [27, 60, 17]. In [27], the authors use a regional selection approach with cost estimates that combine network uncertainty via MC dropout [13] with an effort estimate regressed from ground-truth annotation click patterns. View consistency for active learning for segmentation has not been investigated to the best of our knowledge. The work of [31] comes close to ours in spirit, in which the authors investigate the effect of using multiple disjoint features (views), each of which describe the target concept. They show the effectiveness of using multiple views in active learning for domains like web page classification, advertisement removal, and discourse tree parsing. The work is extended in [62] for image classification tasks.

3. Method

Our ViewAL method consists of four main steps (see Fig. 3): training the network on the current set of labeled data, estimating the model uncertainty on the unlabeled part of the data, selecting which super pixels to request labels for, and finally obtaining annotations. This series of steps is repeated until the labeling budget is reached or all the data labeled. We now describe these steps in more detail.

3.1. Network Training

We start by training a semantic segmentation network to convergence using currently labeled dataset D_L. Initially, D_L is a small randomly selected subset of the dataset for which ground truth has been obtained.

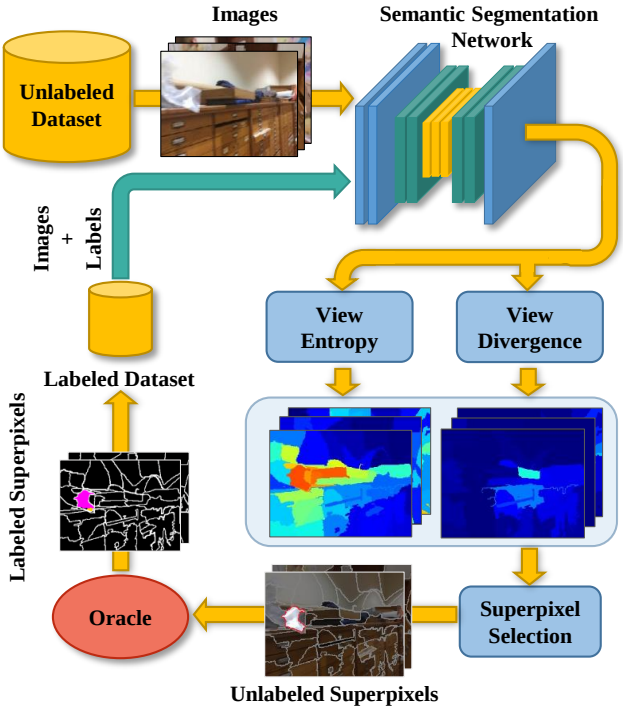


Figure 3: Method overview: in each round of active selection, we first train a semantic segmentation network on the existing labeled data. Second, we use the trained network to compute a view entropy and a view divergence score for each unlabeled superpixel. We then select a batch of superpixels based on these scores, and finally request their respective labels from the oracle. This is repeated until the labeling budget is exhausted or all training data is labeled.

In theory, any semantic segmentation network can be used. We choose DeepLabv3+ [5] with MobileNetv2 [40] as the backbone. We make this choice as DeepLabv3+ is one of the top performing segmentation networks on popular semantic segmentation benchmarks [11, 7], and when combined with the MobileNetv2 backbone, it allows fast training, inference at low memory consumption.

The MobileNetv2 backbone is initialized with weights from a model that was pre-trained on the ILSVRC 1000-class classification task [38]. The rest of the layers use Kaiming initialization [18]. To prevent overfitting, we use blur, random crop, random flip, and Gaussian noise as data augmentations.

3.2. Uncertainty Scoring

Once the network is trained on D_L, our active learning method aims at predicting which samples from the unlabeled part of this dataset, D_U, are the most likely to be the most informative to the current state of the network. To this end, we introduce a new sample selection policy based on view entropy and view divergence scores. Fig. 4 provides an overview of these two new scoring mechanisms.

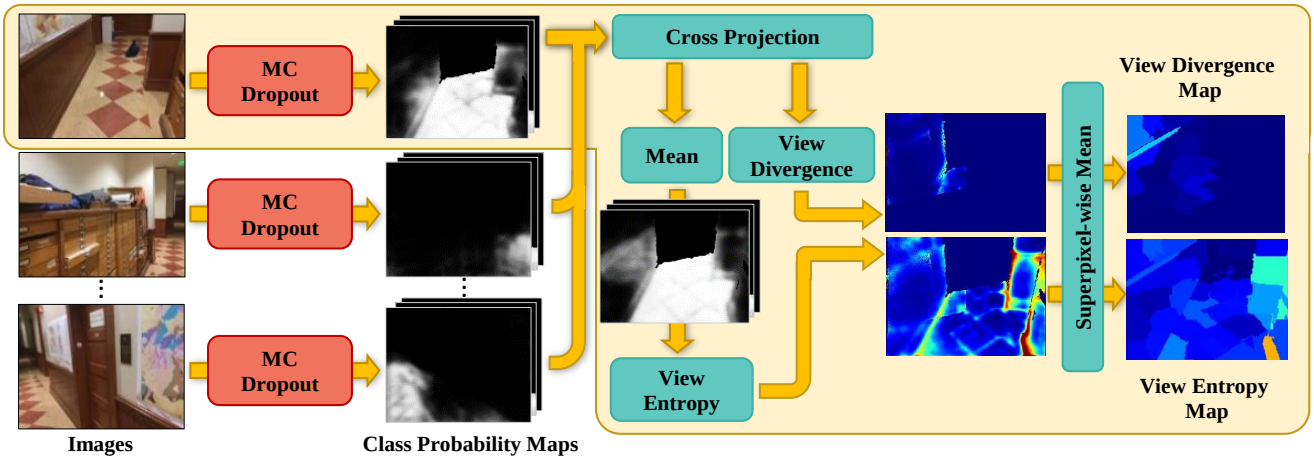


Figure 4: Computation of the view entropy and the view divergence scores. For each unlabeled superpixel in the dataset, we perform MC dropout [13] using 20 runs of dropout and average them to obtain class-probability maps. Next, we back-project each pixel and their associated class-probability distribution to 3D, and re-project all of these to the unlabeled superpixel, effectively providing multiple class-probability predictions per pixel. We then define the view entropy score as the entropy of the average class-probability distribution at each pixel. The view divergence corresponds to the average pairwise KL divergence between the class-distribution at any given pixel and the class-distributions projected at that pixel, effectively capturing the amount of agreement between the prediction in the current view with the prediction coming from other viewpoints. Finally, a view divergence score and a view entropy score is associated with each unlabeled superpixel by averaging the view divergence score and view entropy score of all the pixels they contain.

3.2.1 View Entropy Score

In a nutshell, the proposed view entropy score aims at estimating which objects are consistently predicted the same way, irrespective of the observation viewpoint. For each image, we first calculate its pixel-wise class probability maps using the segmentation network. To make the probability estimates more robust to changes in the input, we use the MC dropout method [13]. The probability for a pixel at position (u, v) in image I_i to belong to class c is given by

$$P_i^{(u,v)}(c) = \frac{1}{D}\sum_{d=1}^{D} P_{i,d}^{(u,v)}(c), \tag{1}$$

where D is the number of test time dropout runs of the segmentation network, and $P_{i,d}^{(u,v)}(c)$ is the softmax probability of pixel (u, v) belonging to class c in the MC dropout run d.

Next, using pose and depth information, all the pixels from the dataset and their associated probability distribution are back-projected to 3D, and projected onto all images. Each pixel (u, v) in image I_i is now associated with a set of probability distributions $\Omega_i^{(u,v)}$, each coming from a different view;

$$\Omega_i^{(u,v)} = \{P_j^{(x,y)}, j \mid I_j(x, y) \text{ cross-projects to } I_i(u, v)\} \tag{2}$$

The mean cross-projected distribution $Q_i^{(u,v)}$ can then be calculated as

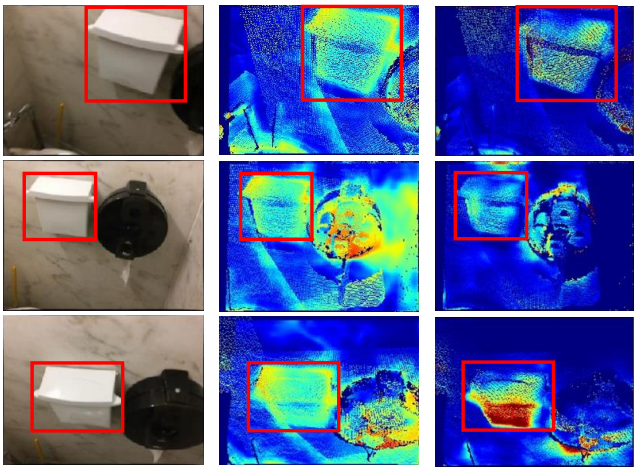
$$Q_i^{(u,v)} = \frac{1}{|\Omega_i^{(u,v)}|}\sum_{P \in \Omega_i^{(u,v)}} P^{(u,v)} \tag{3}$$

which can be seen as marginalizing the prediction probabilities over the observation viewpoints. Finally, the view entropy score $VE_i^{(u,v)}$ for image I_i is given by

$$VE_i^{(u,v)} = -\sum_c Q_i^{(u,v)}(c) \log\left(Q_i^{(u,v)}(c)\right) \tag{4}$$

3.2.2 View Divergence Score

Since the view entropy indicates for each pixel how inconsistent the predictions are across views, this score is the same for all the pixels that are in correspondence (Fig. 5(b)) since it is calculated using probabilities marginalized over different views (Eq. 3). At this stage we are then able to establish the objects for which the network makes view inconsistent predictions, but we still need to determine which view(s) contains the largest amount of beneficial information to improve the network. To this end, we calculate a view divergence score for each pixel, which indicates how predictions about a particular 3D point observed in other views differ from the corresponding prediction in the current image. The view divergence score $VD_i^{(u,v)}$ for a pixel



(a) Image (b) View Entropy (c) View Divergence

Figure 5: View entropy and view divergence scores. For all score-maps, blue indicates low values, and red indicates high values. Pixels that are in correspondence hold the same view entropy score (b), since this score corresponds to a measure computed on the average class-probability of coming from all the pixels in correspondence. We then use the view divergence (c) to define which is the most disagreeing view, and send it to the oracle for annotation.

(u, v) in image I_i is given by

$$VD_i^{(u,v)} = \frac{1}{|\Omega_i^{(u,v)}|} \sum_{P_j \in \Omega_i^{(u,v)}} D_{KL}(P_i^{(u,v)} || P_j^{(u,v)}), \quad (5)$$

where $D_{KL}(P_i^{(u,v)} || P_j^{(u,v)})$ is the KL Divergence between distributions $P_i^{(u,v)}$ and $P_j^{(u,v)}$. A high view divergence score implies that on average, the prediction in the current view is significantly different to the predictions coming from other views (Fig. 5).

3.3. Region Selection

To exploit the structure and locality in semantic segmentation masks, we argue for selecting regions to be labeled instead of whole images. In particular, we opt for using superpixels since they are most of the time associated with a single object-class and therefore lightweight to label for the annotator. Fixed sized rectangular regions would have been another option, but most of the time contain more than one object-class, leading to more effort for the annotator to precisely delineate the boundary between objects. Our implementation uses the SEEDS [55] algorithm for superpixel computation.

For each superpixel r, the two scores VE_i^r and VD_i^r are computed as average of the view entropy and view divergence of all the pixels in r

$$VE_i^r = \frac{1}{|r|} \sum_{(u,v) \in r} VE_i^{(u,v)} \quad (6)$$

$$VD_i^r = \frac{1}{|r|} \sum_{(u,v) \in r} VD_i^{(u,v)}, \quad (7)$$

with $|r|$ is the number of pixels in superpixel r.

Our strategy to select the next superpixel to label consists of two steps. First, we look for the superpixel r from image I_i that has the highest view entropy:

$$(i, r) = \underset{(j,s)}{\text{argmax}}\ VE_j^s \quad (8)$$

Then, we identify the set of superpixels in the dataset whose cross-projection overlap is at least with 50% of (i, r), including self, and denote this set as S. We then look for the superpixel from S that has the highest view divergence as:

$$(k, t) = \underset{(j,s) \in S}{\text{argmax}} \{\, VD_j^s \mid (j, s) \text{ and } (i, r) \textit{ overlap} \,\} \quad (9)$$

All the superpixels in S are then removed from further selection considerations. This selection process is repeated until we select superpixels equivalent to the requested K images.

(a) Image (b) Selected Superpixel (c) Acquired Labels

Figure 6: Label acquisition. We ask the oracle to label only the superpixels selected by our method (marked red in (b)). The remaining superpixels of the ground-truth map, shown in black in (c), are marked with the *ignore* label.

3.4. Label Acquisition

Next, we acquire labels for superpixels selected in Section 3.3. Instead of using a real annotator, we simulate annotation by using the ground truth annotation of the superpixels as the annotation from the oracle. These labeled regions are then added to the labeled dataset and removed from the unlabeled dataset. The labeled dataset therefore is comprised of a lot of images, each with a subset of their superpixels labeled. The unlabeled superpixels of these images are marked with the *ignore* label (Fig. 6).

The active selection iteration concludes with the retraining of the network with the updated dataset D_L from scratch.

4. Results

4.1. Datasets and Experimental Settings

We evaluate our approach on three public datasets, SceneNet-RGBD [30, 29], ScanNet [9], and Matterport3D [3]. All datasets have a large number of RGBD frames across indoor scenes along with their semantic annotations. SceneNet-RGBD is a synthetic dataset containing large-scale photorealistic renderings of indoor scene trajectories. ScanNet contains around 2.5M views in 1513 real indoor scenes. Matterport3D has around 200K RGBD views for 90 real building-scale scenes. We use a subset of images (72290, 23750, and 25761 for SceneNet-RGBD, ScanNet, and Matterport3D, respectively), since active learning iterations on the entire datasets would be too expensive in terms of compute. Further, we resize all images to a resolution of 320×240 pixels. We refer to the supplementary material for statistics of dataset subsets. We use the official train/test scene splits for training and evaluation for ScanNet and Matterport3D. For SceneNet-RGBD, as train set, we use every tenth frame from 5 of the 17 training splits, and for validation, every 10th frame from the validation split. The seed set is initialized with fully annotated 1.5% of training images that are randomly selected from the unlabeled dataset. For ScanNet and Matterport3D, the mIoU is reported on the test set on convergence. For SceneNet, we report the mIoU on the validation set, since a public test set is not available. The model is considered converged when the mIoU of the model does not increase within 20 epochs on the validation set. For ScanNet and Matterport3D the networks are trained with SGD with an initial learning rate of 0.01, while for SceneNet-RGBD we use a learning rate of 0.005. The learning rate is decayed on the 40th epoch to 0.1 times its original value. Further, we set momentum to 0.9, and weight decay penalty to 0.0005.

4.2. Comparisons against Active Learning Methods

We compare our method against 9 other active selection strategies. These include random selection (**RAND**), softmax margin (**MAR**) [21, 37, 57], softmax confidence [48, 57] (**CONF**), softmax entropy [19, 57] (**ENT**), MC dropout entropy [14] (**MCDR**), Core-set selection [44] (**CSET**), maximum representativeness [60] (**MREP**), CEAL entropy [57] (**CEAL**), and regional MC dropout entropy [27] (**RMCDR**). For all methods, we use the same segmentation network (DeepLabv3+ [5]); we also use the same random seed during sample selections for a fair comparison. At the end of each active iteration, the active selection algorithm chooses the next K samples to be labeled ($K = 1500, 1250$, and 1000 for SceneNet-RGBD, ScanNet and Matterport3D, respectively). We use 40 superpixels per image in our method and a window size of 40×40 in the case of **RMCDR**. For further details, refer to the supplementary material.

Fig. 7 shows the mIoU achieved by the segmentation model against percentage of dataset labeled used to train the model. All the active learning methods outperform random sampling (**RAND**). Our method achieves a better performance than all other methods for the same percentage of labeled data. On ScanNet, with just 17% labeled data, we are able to obtain an mIoU of 27.7%. This is around 95% of the model performance achieved when trained over the whole dataset (28.9%) on ScanNet dataset. The method that comes closest to ours (**RMCDR**) achieves the same performance with 27% data. We outperform other methods on SceneNet-RGBD and Matterport3D dataset as well, where we achieve 95% of maximum model performance with just 7% and 24% labeled data respectively, while the runner-up method does so with 14% and 33% data on the respective datasets. The results are presented in tabular form in the supplementary material.

4.3. Evaluation on Labeling Effort

The previous experiment compared active learning performance as a function of proportion labeled data used for training. However, it gives no indication of the effort involved in labeling that data. In order to give a clear indication of the actual effort involved, we compare the time taken to annotate images to the time taken to annotate superpixels.

Since the majority of selections made by our approach have just a single ground truth label associated with them (Fig. 8), it is expected that they will require less labeling time. We verified this in a user study, where we asked users to label 50 images and their equivalent 2000 superpixels (40 superpixels per image) selected from the ScanNet dataset. The images were labeled using a LabelMe [39]-like interface, LabelBox[2], while superpixels were labeled using a superpixel-based interface, which allowed boundary corrections. We made sure that the quality of labels for both images and superpixels was comparable (0.953 mIoU) and acceptable (0.926 and 0.916 mIoU against ground-truth, respectively). Further, to reduce the variance in time due to user proficiency, we made sure that an image and its respective superpixels were labeled by the same user.

It took our annotators 271 minutes to annotate 50 images, while it took them just 202 minutes to annotate their superpixels, therefore bringing down the time by 25% and demonstrating that using superpixels indeed reduces labeling effort. Fig. 9 compares performance of our active learning as a function of labeling effort involved, taking into account the reduction in effort due to superpixel-wise labeling. Effort here is measured in terms of time, with 1% effort being equivalent to the time taken to annotation 1% of data using a LabelMe-like interface. Our method achieves 95%

[2] https://labelbox.com/

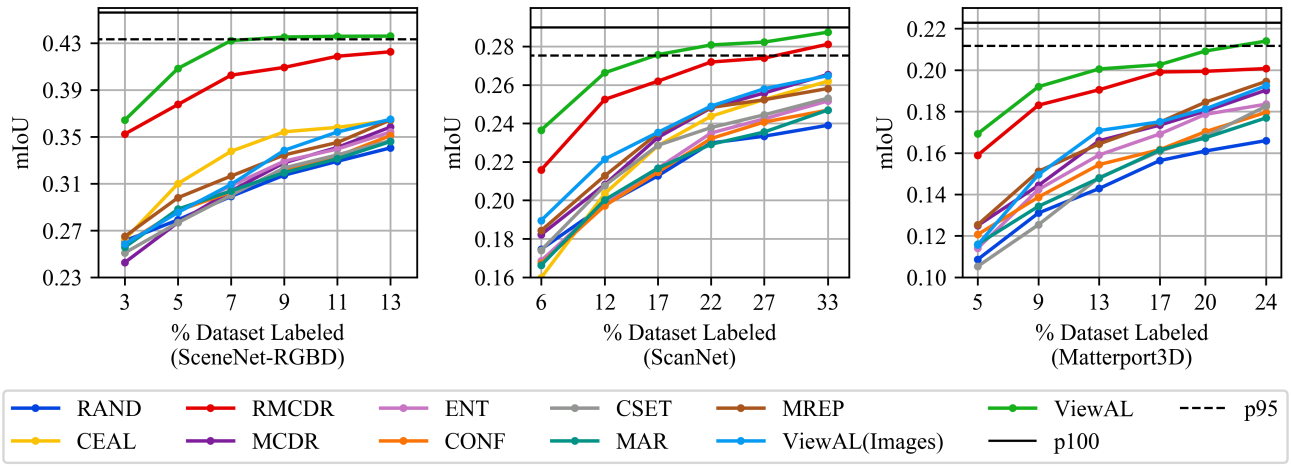


Figure 7: Active learning performance for our method and other baselines. The horizontal solid line (p100) at the top represents model performance with the entire training set labeled. The dashed line (p95) represents 95% of that performance. We observe that our method outperforms all other methods and is able to achieve 95% of maximum model performance with just 7%, 17% and 24% labeled data on SceneNet-RGBD [29], ScanNet [9], and Matterport3D [3] datasets. Note that we omit the results with the seed set here since all methods have the same performance on it.

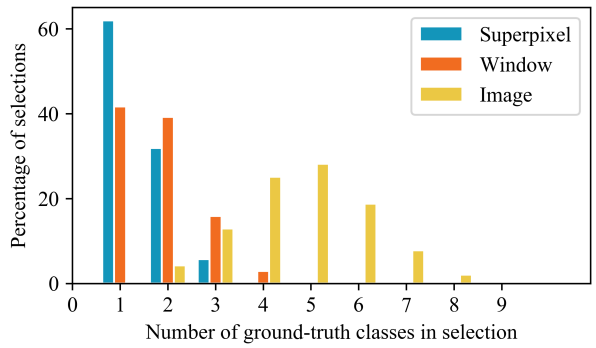


Figure 8: Distribution of number of unique ground truth classes in selections made by different approaches. The majority of the selections made by our superpixel approach have only a single ground truth label in them. Compared to the fixed window approach which has an expected 1.83 unique classes per selection, our approach has 1.40 expected classes per selection. Here, the average number of pixels per window and per superpixel was taken to be the same.

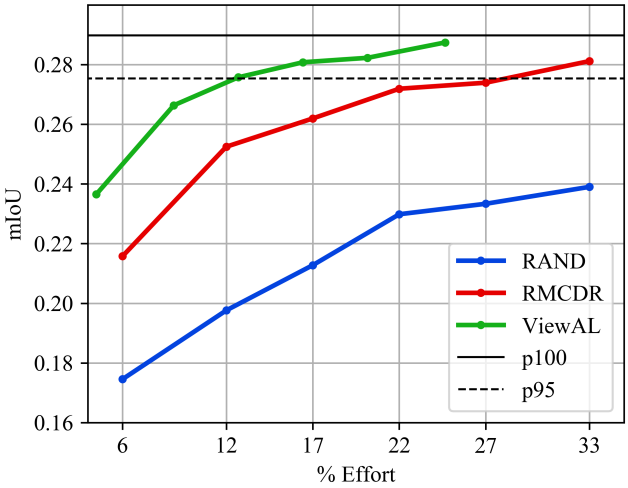


Figure 9: mIoU vs Labeling Effort on the ScanNet dataset. One unit of effort is the expected time taken to label 1% of the dataset using a LabelMe [39]-like interface. Our method delivers 95% of maximum model performance while requiring only 13% effort.

maximum model performance with just 13% effort, compared to 27% effort by the runner-up method on the ScanNet dataset.

4.4. Ablation Studies

Effect of view entropy in isolation. In order to show that view entropy provides a helpful signal for active selection, we evaluate the performance of a non-regional variant (i.e., no superpixel selection) without the use of MC dropout or view divergence. In this case, $P_i^{(u,v)}(c)$ from Eq. 14 is just the softmax probability of the network (instead of average across MC dropout runs). That means that images are only selected on the basis of the view entropy score (Section 3.2.1), which is averaged across all pixels of the image. Fig. 10 shows that view entropy in isolation significantly helps selection while outperforming both **RAND** and **ENT** methods.

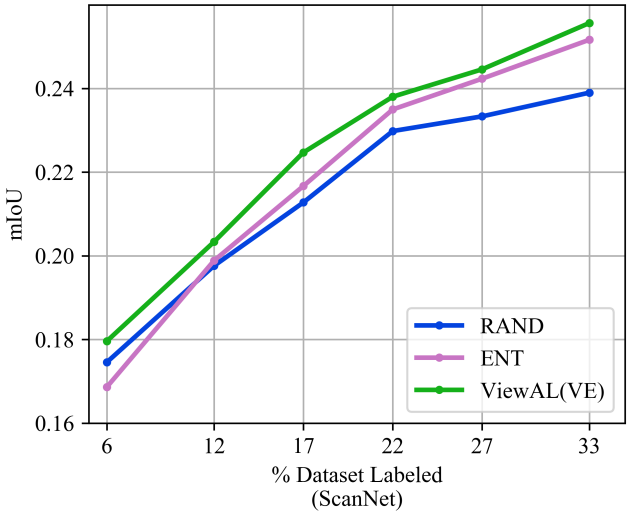


Figure 10: Results using only view entropy (i.e., without superpixels, MC dropout, or view divergence) which outperforms both random (RAND) and softmax entropy (ENT) methods.

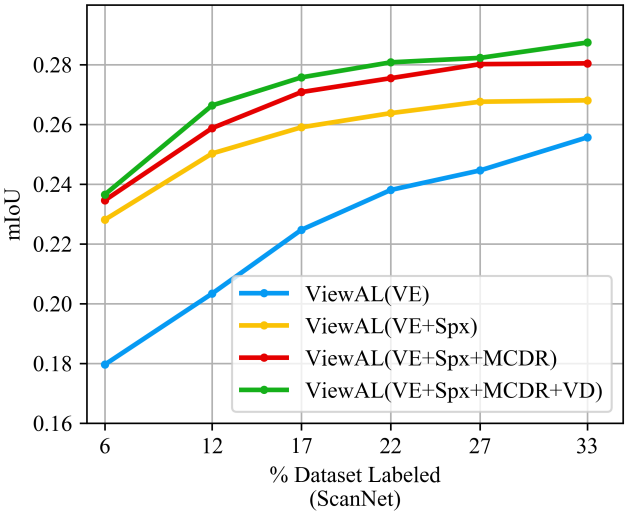


Figure 11: Ablation study of our method: ViewAL(VE) is our method without superpixels, MC dropout, and view divergence. When superpixels are used for selection over entire images, we see significant improvements as shown by the curve ViewAL(VE+Spx). Adding MC dropout improves performance further as indicated by ViewAL(VE+Spx+MCDR). Our final method, ViewAL(VE+Spx+MCDR+VD) improves over this further by adding view divergence.

Effect of superpixels. We evaluate the effect of using superpixel selection in combination with view entropy scores. In Fig. 11, the curve ViewAL(VE+Spx) shows the effectiveness of selecting superpixels rather than entire images. Active learning performance significantly improves as superpixel-based region selection facilitates focus only on high scoring regions while ignoring less relevant ones.

Effect of using MC dropout. Instead of using the plain softmax probabilities as in the last two paragraphs, we use MC dropout to get an estimate of class probabilities as shown in Eq. 14. The curve ViewAL(VE+Spx+MCDR) in Fig. 11 shows that using MC dropout improves active learning performance by a considerable margin. This can be explained by MC dropout providing a better estimate of class posteriors than just using simple softmax probabilities.

Effect of using view divergence. Finally, we add view divergence score (Section 3.2.2), giving us our complete method. View divergence helps select the view for a superpixel with the most different class-wise probability distribution from the rest of views that superpixel appears in. It further improves the active learning performance of our method as shown by the curve ViewAL(VE+Spx+MCDR+VD) in Fig. 11.

5. Conclusions

We have introduced ViewAL , a novel active learning method for semantic segmentation that leverages inconsistencies in prediction across views as a measure of uncertainty. In addition to a new view entropy score, we propose to use regional selection in the form of superpixels, which we incorporate in a unified active learning strategy. This allows us to acquire labels for only the most promising areas of the images and at the same time reduce the labeling effort. While we have shown results on widely-used indoor datasets, our method is agnostic to the respective multi-view representation, and can easily be applied to a wide range of scenarios. Currently, our focus is on semantic segmentation; however, we believe that this work provides a highly promising research avenue towards other tasks in computer vision, including instance segmentation, object detection, activity understanding, or even visual-language embeddings.

Acknowledgements

This work is funded by Google (Daydream - AugmentedPerception), the ERC Starting Grant Scan2CAD (804724), and a Google Faculty Award. We would also like to thank the support of the TUM-IAS, funded by the German Excellence Initiative and the European Union Seventh Framework Programme under grant agreement n 291763, for the TUM-IAS Rudolf Mößbauer Fellowship. Finally, we thank Angela Dai for the video voice over.

References

[1] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017. 1

[2] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9368–9377, 2018. 2, 3

[3] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. 6, 7, 11, 13, 14

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 1

[5] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018. 3, 6

[6] Kashyap Chitta, Jose M Alvarez, and Adam Lesnikowski. Large-scale visual active learning with deep probabilistic ensembles. *arXiv preprint arXiv:1811.03575*, 2018. 3

[7] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 3

[8] Ido Dagan and Sean P Engelson. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier, 1995. 12

[9] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5828–5839, 2017. 6, 7, 11, 13, 14

[10] Dumitru Erhan, Christian Szegedy, Alexander Toshev, and Dragomir Anguelov. Scalable object detection using deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2154, 2014. 1

[11] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 3

[12] Alexander Freytag, Erik Rodner, and Joachim Denzler. Selecting influential examples: Active learning with expected model output changes. In *European Conference on Computer Vision*, pages 562–577. Springer, 2014. 2

[13] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059, 2016. 3, 4, 12

[14] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1183–1192. JMLR. org, 2017. 2, 3, 6, 12

[15] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 1

[16] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 1

[17] Marc Gorriz, Axel Carlier, Emmanuel Faure, and Xavier Giro-i Nieto. Cost-effective active learning for melanoma segmentation. *arXiv preprint arXiv:1711.09168*, 2017. 3

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 3

[19] Rebecca Hwa. Sample selection for statistical parsing. *Computational linguistics*, 30(3):253–276, 2004. 2, 6

[20] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*, pages 34–50. Springer, 2016. 1

[21] Ajay J Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class active learning for image classification. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2372–2379. IEEE, 2009. 2, 6

[22] Christoph Käding, Erik Rodner, Alexander Freytag, and Joachim Denzler. Active and continuous exploration with deep neural networks and expected model output changes. *arXiv preprint arXiv:1612.06129*, 2016. 2

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[24] David D Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*, pages 148–156. Elsevier, 1994. 2

[25] David D Lewis and William A Gale. A sequential algorithm for training text classifiers. In *SIGIR94*, pages 3–12. Springer, 1994. 2

[26] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[27] Radek Mackowiak, Philip Lenz, Omair Ghori, Ferran Diego, Oliver Lange, and Carsten Rother. Cereals-cost-effective region-based active learning for semantic segmentation. *arXiv preprint arXiv:1810.09726*, 2018. 2, 3, 6, 12

[28] Andrew Kachites McCallumzy and Kamal Nigamy. Employing em and pool-based active learning for text classification. Citeseer. 2

[29] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J.Davison. Scenenet rgb-d: 5m photorealistic images of synthetic indoor trajectories with ground truth. 2016. 1, 6, 7, 11, 13, 14

[30] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J.Davison. Scenenet rgb-d: Can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? 2017. 6, 11

[31] Ion Muslea, Steven Minton, and Craig A Knoblock. Active learning with multiple views. *Journal of Artificial Intelligence Research*, 27:203–233, 2006. 3

[32] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016. 1

[33] Hieu T Nguyen and Arnold Smeulders. Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 79. ACM, 2004. 2

[34] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4929–4937, 2016. 1

[35] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 1

[36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1

[37] Dan Roth and Kevin Small. Margin-based active learning for structured output spaces. In *European Conference on Machine Learning*, pages 413–424. Springer, 2006. 2, 6

[38] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 3

[39] Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77(1-3):157–173, 2008. 6, 7

[40] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018. 3

[41] Tobias Scheffer, Christian Decomain, and Stefan Wrobel. Active hidden markov models for information extraction. In *International Symposium on Intelligent Data Analysis*, pages 309–318. Springer, 2001. 12

[42] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016. 13

[43] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 519–528. IEEE, 2006. 13

[44] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017. 3, 6

[45] Ozan Sener and Silvio Savarese. Active Learning for Convolutional Neural Networks: A Core-Set Approach. 08 2017. 12

[46] Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013. 1

[47] Burr Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009. 2, 11

[48] Burr Settles and Mark Craven. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1070–1079. Association for Computational Linguistics, 2008. 2, 6

[49] Burr Settles, Mark Craven, and Soumya Ray. Multiple-instance active learning. In *Advances in neural information processing systems*, pages 1289–1296, 2008. 2

[50] H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 287–294. ACM, 1992. 2

[51] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1

[52] Qing Sun, Ankit Laddha, and Dhruv Batra. Active learning for structured probabilistic models with histogram approximation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3612–3621, 2015. 3

[53] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 1

[54] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001. 2

[55] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin de Capitani, and Luc Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *European conference on computer vision*, pages 13–26. Springer, 2012. 5

[56] Alexander Vezhnevets, Joachim M. Buhmann, and Vittorio Ferrari. Active learning for semantic segmentation with ex-

pected change. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3162–3169, 2012. 2

[57] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, 2016. 2, 3, 6, 11, 12

[58] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016. 1

[59] Zuobing Xu, Ram Akella, and Yi Zhang. Incorporating diversity and density in active learning for relevance feedback. In *European Conference on Information Retrieval*, pages 246–257. Springer, 2007. 2

[60] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 399–407. Springer, 2017. 3, 6, 12

[61] Donggeun Yoo and In So Kweon. Learning loss for active learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 93–102, 2019. 2, 3

[62] Xiaoyu Zhang, Jian Cheng, Changsheng Xu, Hanqing Lu, and Songde Ma. Multi-view multi-label active learning for image classification. In *2009 IEEE International Conference on Multimedia and Expo*, pages 258–261. IEEE, 2009. 3

[63] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 1

A. Dataset Statistics

We evaluate our approach on three public datasets, SceneNet-RGBD [30, 29], ScanNet [9], and Matterport3D [3]. SceneNet-RGBD is a synthetic dataset containing large-scale photorealistic renderings of indoor scene trajectories, with around 5M RGBD frames. ScanNet contains around 2.5M views in 1513 real indoor scenes. Matterport3D has around 200K RGBD views for 90 real building-scale scenes. We use a subset of images, as highlighted in Table 1, since active learning iterations on the entire datasets would be too expensive in terms of compute.

Statistic	SceneNet	ScanNet	Matterport3D
Train Sequences	2434	1041	968
Train Frames	72990	23750	25761
Validation Seqs.	500	465	214
Validation Frames	15000	5453	13702
Test Sequences	-	80	370
Test Frames	-	5320	22588
Semantic Classes	13	40	40

Table 1: Statistics of SceneNet-RGBD[29], ScanNet[9] and Matterport3D[3] dataset subsets used in our experiments.

B. Baseline Active Learning Methods

We compare our method against popular uncertainty and diversity based active learning approaches found in the literature. Here, we give a brief overview of these approaches. In terms of notation, D_U is the unlabeled dataset, D_L is the currently labeled dataset, M is the total number of target classes, K is the number of images from D_U requested to be labeled in each active selection iteration, n goes over pixels for image i and θ_{SEG} are the parameters of the segmentation network.

Random Selection (RAND) In random selection, in each active selection iteration, the next query for K samples is composed of randomly selected samples from the unlabeled dataset.

Softmax Confidence (CONF) The least confidence approach discussed in [47] can be adapted to deep convolutional networks by using softmax probability of the most probable class as confidence [57]. This selection strategy then selects the least K confident samples from D_U as the next query. For semantic segmentation, we calculate confidence for each pixel and use the sum across pixels as the confidence for the image. For each image i, the confidence

score is therefore given by Eq. 10, and K least scoring samples are selected for label acquisition.

$$S_i^{CONF} = \sum_n \max_j p(y_i^n = j \mid x_i; \theta_{SEG}) \quad (10)$$

Softmax Margin (MAR) Similar to CONF, this approach [41] ranks all the samples in order of the difference of softmax probabilities of the most probable label (j_1) and the second most probable label (j_2), and chooses the K samples which have the least difference (Eq. 11) [57]. The idea is that samples for which the network has a small margin between the top predictions means that the network is very uncertain between the two.

$$S_i^{MAR} = \sum_n (p(y_i^n = j_1 | x_i; \theta_{SEG}) - p(y_i^n = j_2 | x_i; \theta_{SEG})) \quad (11)$$

Softmax Entropy (ENT) In the case of semantic segmentation, the entropy value for each pixel in the image is summed to get the entropy score for the whole image (Eq. 12).

$$S_i^{ENT} = -\sum_n \sum_{j=1}^{M} p(y_i^n \mid x_i; \theta_{SEG}) \log p(y_i^n \mid x_i; \theta_{SEG}) \quad (12)$$

Entropy takes into account probabilities of all classes unlike CONF, which considers most probable class or MAR, which only considers the top two most probable classes.

CEAL Entropy (CEAL) CEAL [57] combines CONF, MAR, ENT methods with pseudo-labeling in their active learning framework. We only compare with their ENT variant since the results are quite identical for all the other measures. At the end of each active selection iteration, they propose not only adding samples labeled by the oracle, but also high confidence samples from D_U for which softmax entropy is less than the threshold δ. For these samples, the assigned labels are the predicted ones by the current model. The idea behind pseudo-labeling is that since the high confidence samples are close to the labeled samples in CNNs feature space, adding them in training is a reasonable data augmentation for CNN to learn robust features. Further, as the active iteration increase, the number of samples selected for pseudo-labeling increases since the network gets more and more confident. To prevent high amounts of pseudo-labeling, the threshold is decreased at the end of each selection iteration. Our implementation of CEAL only assigns pseudo-labels at pixel level instead of image level to account for locality of segmentation task.

Monte Carlo Dropout (MCDR) It has been argued in [13] that vanilla deep learning models rarely represent model uncertainty, and softmax entropy is not really a good measure of uncertainty. Instead of softmax probabilities [14] use Monte Carlo (MC) dropout to estimate model uncertainty.

$$S_i^{MCDR} = -\sum_n \sum_{j=1}^{M} p_{MC}(y_i^n \mid x_i; \theta_{SEG}) \log p_{MC}(y_i^n \mid x_i; \theta_{SEG}) \quad (13)$$

where p_{MC} is given by

$$p_{MC}(y_i^n \mid x_i; \theta_{SEG}) = \frac{1}{D} \sum_{d=1}^{D} p_{SM}(y_i^n \mid x_i; \theta_{SEG}^d), \quad (14)$$

with D being the total number of MC Dropout runs.

Regional MC Dropout (RMCDR) Proposed for semantic segmentation in [27], it follows the same approach as MCDR. However, instead of calculating scores for whole images, scores are calculated for fixed-size regions. The selection algorithm is then selecting as many highest entropy regions as it takes to make up K images. The original method of [27] uses Vote Entropy [8], however we use MC Dropout since it gives slightly better results. Further, the method of [27] uses cost estimates regressed from annotator click patterns, which we dont use since these are not available for any of the datasets we evaluate on.

Maximum Representativeness (MREP) Unlike the other approaches discussed until now which were only uncertainty based, MREP is a mixed approach that combines uncertainty and diversity. This method, proposed in [60] first choose points that are highly uncertain. From among these points, it further chooses points that best represent the rest of distribution based on some similarity measure. In our implementation, vote entropy is first used to select $2K$ samples, and then K most representative samples amongst those are selected to be labeled. We use the Euclidean norm for the similarity measure.

Core-Set Selection (CSET) Core-Set [45] is a purely diversity-based approach. The method aims to select a subset of K points such that the model trained on a subset of the points is competitive for the rest of the points. The K samples selected are the ones that have the smallest δ for the δ cover of the set. This means that the algorithm seeks to minimize the maximum distance between sample x_i in the remaining unlabeled dataset and its closest neighbor x_j in the selected subset. We use the simple greedy selection strategy proposed in [45] as it performs only slightly worse than the robust version.

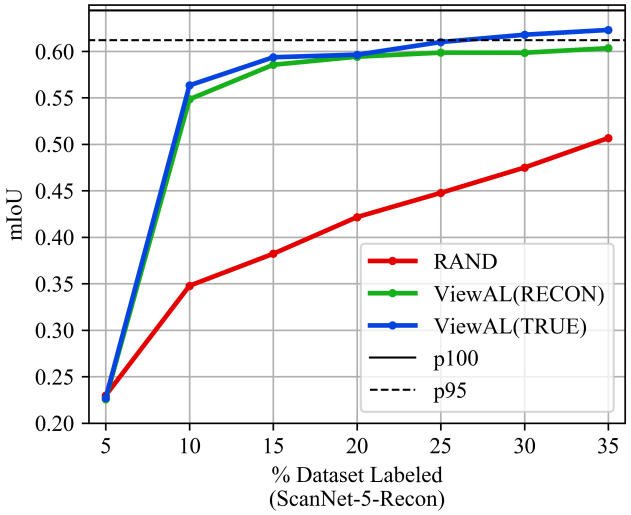


Figure 12: Performance with imperfect depth and pose. Our method using reconstructed depth and pose, ViewAL(RECON), outperforms the RAND baseline and performs only slightly worse than the variant using true depth and poses, ViewAL(TRUE).

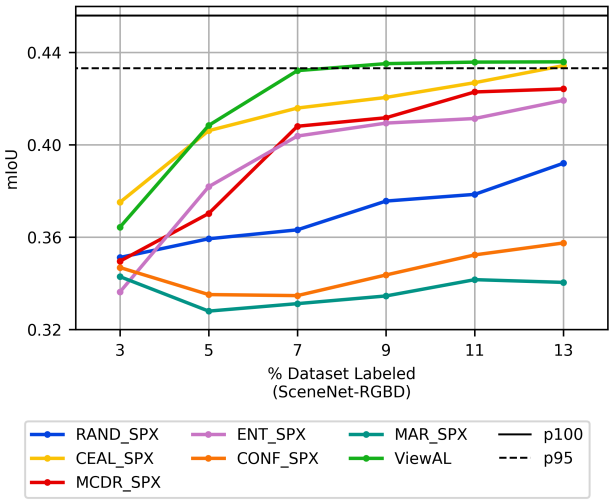


Figure 13: Active learning performance for our method and other baselines when all baselines use superpixels.

C. Performance with Imperfect Depth and Pose

Here we evaluate the performance of our method when the ground truth depth and pose are not available, i.e. only RGB frames are available. In such a case, one alternative for making associations between pixels across frames is to use structure from motion/multi-view stereo methods. We use COLMAP [42, 43] to first reconstruct the scenes from RGB frames and obtain depth and camera parameters. We use 5 scenes from ScanNet [9] for this. We keep to just 5 scenes as the time taken to reconstruct a scene using COLMAP is quite long, and since here we only want to compare the performance using ground truth depth and pose against reconstructions, these should be sufficient. We use 1000 frames from each scene, and split the total 5000 frames into 2000 training (unlabeled), 1000 validation and 2000 test frames. The seed set has 100 fully labeled frames. Each selection iteration chooses 100 more frames (or equivalent superpixels) from the training set to be labeled. We compare against random selection (**RAND**) and the variant of our method that uses true pose and depth (ViewAL(TRUE)).

Fig. 12 shows the results for this experiment. We observe that our method which uses reconstructed depth and pose still outperforms the **RAND** baseline and performs only slightly worse than the variant using true depth and poses.

D. Comparison with baselines allowed to select superpixels

Fig. 13 shows the scenario where other methods are allowed to use superpixel selection instead of window / image selection. It can be observed that most methods do benefit from superpixel selection.

E. Handling non-static data

For computation of view entropy and divergence scores, we need to associate superpixels between frames. In our experiments, we use frame depth and pose to get these associations. However, this can be done only in case of static scenes, i.e. when objects do not change positions across frames. A promising future direction could be to extend this work for the dynamic setting using, for instance, optical flow estimates or keypoint descriptor matching to achieve superpixel association across frames.

F. Result Tables

Due to limited space in the main paper, we present the experimental results here in tabular form. Table 2, Table 3, Table 4 list results for all the methods we compared on SceneNet-RGBD [29], ScanNet [9] and Matterport3D [3] datasets. Table 5 reports results for the ablation study.

% Labeled Data	RAND	RMCDR	MCDR	ENT	CONF	CSET	MAR	MREP	CEAL	ViewAL (Images)	ViewAL
1	0.2245	0.2124	0.2160	0.2261	0.2257	0.2259	0.2254	0.2168	0.2255	0.2159	0.2125
3	0.2612	0.3524	0.2427	0.2586	0.2584	0.2509	0.2558	0.2650	0.2624	0.2585	**0.3643**
5	0.2791	0.3776	0.2768	0.2864	0.2868	0.2767	0.2882	0.2980	0.3101	0.2854	**0.4084**
7	0.2991	0.4026	0.3038	0.3082	0.3029	0.3001	0.3038	0.3165	0.3376	0.3094	**0.4321**
9	0.3173	0.4092	0.3278	0.3292	0.3208	0.3234	0.3194	0.3345	0.3542	0.3385	**0.4352**
11	0.3290	0.4187	0.3409	0.3395	0.3334	0.3346	0.3313	0.3451	0.3580	0.3541	**0.4358**
13	0.3405	0.4226	0.3583	0.3541	0.3510	0.3467	0.3459	0.3644	0.3639	0.3649	**0.4359**
15	0.3509	0.4337	0.3716	0.3616	0.3630	0.3285	0.3522	0.3755	0.3781	-	**0.4383**
17	0.3587	0.4340	0.3737	0.3726	0.3731	0.3432	0.3688	0.3845	0.3807	-	**0.4412**

Table 2: Semantic segmentation performance in terms of mIoU when labeled data is selected using baseline active learning methods and our method on SceneNet-RGBD [29] dataset.

% Labeled Data	RAND	RMCDR	MCDR	ENT	CONF	CSET	MAR	MREP	CEAL	ViewAL (Images)	ViewAL
1	0.0998	0.0957	0.0950	0.0961	0.0958	0.0961	0.0999	0.0934	0.1001	0.0957	0.0953
6	0.1746	0.2158	0.1821	0.1686	0.1672	0.1741	0.1662	0.1843	0.1598	0.1895	**0.2365**
12	0.1976	0.2525	0.2083	0.1989	0.1972	0.2077	0.2003	0.2128	0.2035	0.2214	**0.2663**
17	0.2128	0.2619	0.2327	0.2167	0.2146	0.2286	0.2167	0.2349	0.2284	0.2353	**0.2757**
22	0.2298	0.2719	0.2480	0.2350	0.2321	0.2378	0.2291	0.2483	0.2437	0.2490	**0.2808**
27	0.2333	0.2739	0.2558	0.2423	0.2407	0.2444	0.2355	0.2523	0.2524	0.2580	**0.2823**
33	0.2390	0.2812	0.2654	0.2517	0.2469	0.2531	0.2470	0.2581	0.2619	0.2648	**0.2874**

Table 3: Semantic segmentation performance in terms of mIoU when labeled data is selected using baseline active learning methods and our method on ScanNet [9] dataset.

% Labeled Data	RAND	RMCDR	MCDR	ENT	CONF	CSET	MAR	MREP	ViewAL (Images)	ViewAL
1	0.0754	0.0797	0.0825	0.0765	0.0762	0.0778	0.0781	0.0807	0.0815	0.0802
5	0.1086	0.1589	0.1250	0.1141	0.1207	0.1053	0.1159	0.1254	0.1157	**0.1693**
9	0.1310	0.1831	0.1443	0.1424	0.1387	0.1254	0.1343	0.1512	0.1496	**0.1920**
13	0.1429	0.1905	0.1659	0.1590	0.1544	0.1481	0.1478	0.1644	0.1708	**0.2005**
17	0.1564	0.1991	0.1735	0.1692	0.1616	0.1609	0.1614	0.1749	0.1750	**0.2026**
20	0.1609	0.1994	0.1802	0.1787	0.1703	0.1680	0.1673	0.1845	0.1813	**0.2092**
24	0.1660	0.2007	0.1903	0.1836	0.1796	0.1826	0.1769	0.1945	0.1925	**0.2140**
27	0.1766	0.2042	0.1947	0.1826	0.1839	0.1850	0.1777	0.1971	-	**0.2148**
31	0.1823	0.2112	0.2032	0.1960	0.1915	0.1902	0.1869	0.2019	-	**0.2159**

Table 4: Semantic segmentation performance in terms of mIoU when labeled data is selected using baseline active learning methods and our method on Matterport3D [3] dataset.

% Labeled Data	ViewAL(VE)	ViewAL(VE+Spx)	ViewAL(VE+Spx+MCDR)	ViewAL(VE+Spx+MCDR+VD)
1	0.1004	0.1001	0.0952	0.0952
6	0.1795	0.2280	0.2345	**0.2365**
12	0.2033	0.2502	0.2587	**0.2663**
17	0.2247	0.2590	0.2708	**0.2757**
22	0.2380	0.2637	0.2754	**0.2807**
27	0.2445	0.2675	0.2801	**0.2822**
33	0.2556	0.2680	0.2804	**0.2873**

Table 5: Ablation Study Results. ViewAL(VE) is our method without superpixels, MC dropout, and view divergence. When superpixels are used for selection over entire images, we see significant improvements as shown by the curve ViewAL(VE+Spx). Adding MC dropout improves performance further as indicated by ViewAL(VE+Spx+MCDR). Our final method, ViewAL(VE+Spx+MCDR+VD) improves over this further by adding view divergence.