Pattern Recognition Letters 133 (2020) 349–355

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Multi-criteria online frame-subset selection for autonomous vehicle videos

Soumi Das[a], Sayan Mandal[a], Ashwin Bhoyar[a], Madhumita Bharde[b], Niloy Ganguly[a], Suparna Bhattacharya[b], Sourangshu Bhattacharya[a,*]

[a] Indian Institute of Technology, Kharagpur, West Bengal, 721302, India
[b] Hewlett Packard Enterprise, Bangalore, 560xxx, India

ARTICLE INFO

Article history:
Received 22 August 2019
Revised 13 March 2020
Accepted 29 March 2020
Available online 4 April 2020

MSC:
41A05
41A10
65D05
65D17

Keywords:
Active learning
Video frame selection
Autonomous driving
Semantic segmentation

ABSTRACT

Data Subset selection for training learning models for a variety of tasks, has been widely studied in the literature of batch mode active learning. Recent works attempt to utilize the model specific signals in the deep learning context for computer vision tasks. Companies, in their bid to create safe autonomous driving models, train and test their models on billions of miles of driving data; not all of which may be valuable for a training task. In this paper, we study the problem of frame-subset selection from autonomous vehicle driving data, for the problem of semantic segmentation - which is a crucial component of the perception module in an autonomous driving system. We find that state of the art methods for deep active learning do not utilize pairwise similarity between incoming and existing frames. We explore both active learning settings, where labels for incoming points are not available, as well as frame selection settings and find that our method selects more valuable frames than only score-based frame subset selection, or frame subset selection without label information. We demonstrate the effectiveness of our method using DeeplabV3+ model on both benchmark as well as datasets generated by driving simulators. Our generated dataset and code will be made publicly available.

1. Introduction

Semantic segmentation is a critical sub-task of the perception stack in autonomous driving systems e.g. CARLA [1], and is also an intense area of research in computer vision [2]. However, performance of the state of the art models for semantic segmentation still lacks perfection, and benefits from high amounts of training data. Hence, car companies can utilize the driving videos for training semantic segmentation models. However, training state of the art models on such huge quantities of data is computationally expensive. In this paper, we study the problem of incrementally selecting a subset of video frames from driving videos, such that training on the subset produces a model with performance close to the one trained on the entire video. We consider the scenario where ground truth labels (semantically segmented images) are available for the entire video, e.g. when the videos are obtained from a driving simulator. However, some of the criteria developed here, and the overall method for subset selection is applicable to the active learning setting as well, where ground truth images are needed only for the selected subset.

Existing works on subset selection for computer vision tasks can be classified into the following two categories: (1) Video summarization [3] where explicit supervision for subset of examples to be selected is available; or (2) Active learning [4] where labels for the full set are not available. We find deep batch active learning techniques [5,6] to be the closest to our setting and build on it. Most active learning techniques define acquisition functions which measure the importance of each new datapoint [4]. The main idea in this paper is to use combinations of multiple criteria for selecting datapoints using the convex subset selection framework described in [3]. We propose two types of criteria: *pairwise*, which depends on pairs of datapoints, and *pointwise* which only depends on each datapoint. Both types of criteria may or may not depend on target label without affecting the overall framework, thus allowing it to be used for active learning or data subset selection. While a related concept of "informativeness" and "representativeness" has been explored in batch mode active learning [7,8], they have not been applied to deep learning models. The key advantage of our formulation is not having any positive semi-definiteness constraint on the pairwise criteria.

* Corresponding author.
E-mail address: sourangshu@cse.iitkgp.ac.in (S. Bhattacharya).

https://doi.org/10.1016/j.patrec.2020.03.031
0167-8655/

Another important dimension of designing successful criteria is the ability to incorporate state of the art models - DeeplabV3+ [2] in our case. This has been a recent area of study for various computer vision models, e.g. CNNs [6], and other models such as ResNet101, Inception, etc [5]. We extend definitions of two criteria: *distinctiveness* and *uncertainty*, proposed in [5], which were originally designed for image classification, to the problem of semantic segmentation, terming them as *PW-distinctiveness* and *PW-uncertainty* respectively where PW denotes pixel-wise. We use these along with two other criteria: *SIFT* [9] which is a pairwise criteria and *loss* which is a pointwise criteria which also requires labels, for selecting frames subsets.

We perform a detailed experimental comparison of the various criteria using the proposed framework, on the benchmark dataset of Camvid [10] and a dataset of driving video generated by us using the CARLA driving simulator [1]. The driving video contains a variety of situations comprising of over 4 hours of driving with more than 4000 keyframes. We observe **100:4** compression ratio in frames (selecting 4 out of every 100 frames on average) with only 0.74% reduction in accuracy and ~ 3% reduction in MIoU over the original dataset. Also, the pairwise criterion proposed here improves the performance over state-of the art point-wise criteria proposed in [5], which is further improved over, by incorporating label information using loss function. Finally, we show that our methods behave predictably w.r.t. variation in hyperparameters controlling the size of selected subset.

2. Related work

Traditional active learning seeks to select informative examples in three different ways - Membership Query Synthesis, Stream based Selective Sampling and Pool Based Active Learning [4]. Pool based Active Learning [11] considers the scenario where there is a set of labelled instances besides a large number of unlabelled examples to choose from. Recently, [12] introduced an approach based on min-max view of active learning by measuring both informativeness and representativeness of instances. Their method constructs a semi-supervised SVM-like objective for squared-loss. In [8], the authors proposed a general framework where measures like informativeness and representativeness were not liable to any input data constraints. Similarly, [7] adopted Maximum Mean Discrepancy (MMD) to measure the difference in distribution between labelled and original datasets. Wang et al. [13] proposed an approach to select samples of queries by minimizing the α - relative Pearson divergence (RPE) between the labelled and queried datasets and showed the superiority over the use of MMD. However, all of these works revolve around non-deep frameworks where they use SVM to classify their data and prove their hypothesis.

Recently deep active learning frameworks have been explored. Gal et al. [14] developed a modified active learning model by combining Bayesian deep learning into active learning framework and showed that their model achieves significant improvement on the existing active learning approaches. However, this requires the underlying model to be a deep bayesian network. Huang et al. [5] proposed a method in which a pre-trained model can be adapted to a new task using the measures of distinctiveness and uncertainty which estimate the contribution of the datapoint in optimizing feature representation as well as improving the performance for the new task. Sener and Savarese [6] poses the problem of active learning as core-set selection, which iteratively minimizes the difference between loss over labelled points and that over unlabelled points. Here, we extend the factors proposed by [5] and [6] for semantic segmentation. We develop a formulation for subset selection designed towards the end-goal of semantic segmentation. The problem of subset-selection for the task of video summarization has been well studied in recent literature [3,15]. However, these methods have not been applied in the domain of active learning for performance enhancement. Besides, they keep a positive semi-definite constraint on the pairwise similarity. We intend to modify the formulation proposed by Elhamifar and Kaluza [3] by incorporating various other factors and performing subset-selection for the purpose of semantic segmentation.

3. Methods

3.1. Problem formulation

We consider a dataset $\mathcal{D}$ comprising of frames of a video x_i, and labels y_i corresponding to various tasks on video frames such as semantic segmentation. Hence, $\mathcal{D} = \{(x_i, y_i),\ i = 1, \ldots, n\}$ where n is the number of frames in the dataset. The data may arrive in either a streaming manner, or in batches. We process the data in batches $X_t,\ t = 1, \ldots, T$, where batch size $|X_t| = m$. Hence $mT = n$. We also define the cumulative sets $C_t = \cup_{i=1}^{t} X_i$. We are interested in constructing representative sets $R_t \subseteq C_t$, such that semantic segmentation models trained on R_t perform similarly to those trained on C_t and size of R_t is minimized. Here, performance is the generalization error, measured as the error on a common test set.

Online subset selection formulation: We pose the problem of selecting representative sets R_t as an online subset selection problem, and follow the formulation in [3]. Given a datapoint i from the set X_{t+1} and another datapoint j from the set R_t, R_{t+1} is constructed by solving the problem:

$$\begin{aligned} \min_{z^o_{ij}, z^n_{ij}} \quad & \sum_{i=1}^{m}\sum_{j=1}^{|R_t|} z^o_{ij} d_{ij} + \sum_{i,j=1}^{m} z^n_{ij} d_{ij} + \lambda \sum_{j=1}^{m} \|[z^n_{1,j} \ldots z^n_{m,j}]\|_p \\ s.t. \quad & \sum_{j=1}^{|R_t|} z^o_{i,j} + \sum_{j=1}^{m} z^n_{i,j} = 1 \\ & z^n_{i,j}, z^o_{i,j} \in [0, 1] \end{aligned} \tag{1}$$

where d_{ij} denotes the notion of dissimilarity between frames i and j, z^o_{ij}, z^n_{ij} are relaxed versions of binary assignment variables. $z^o_{ij} = 1$ denotes that the representative of ith new example is jth old example, and $z^n_{ij} = 1$ denotes that representative of ith new example is the jth new example. Under the binary assumption, the linear constraints ensure that every new example gets exactly one representative. Another interpretation can be that, z^n_{ij}, z^o_{ij} are the probabilities that ith new point is represented by jth new or old point. Also, note that the first two terms denote representation error for all the new points, represented with either old or new points, respectively.

The last term penalizes the inclusion of many new points as representatives, hence effectively reducing the number of representatives. λ is a user supplied parameter which controls the penalty for choosing a larger representative set.

3.2. Multiple-criteria-based representative selection

The above formulation relies on (dis)similarity between the examples in the new set (X_{t+1}) and existing representative set (R_t), for selecting the new representatives. However, this does not take care of the possibility that in many application scenarios, one is interested in selecting points which provide better performance towards a given task. For example, one could be interested in selecting video frames (or keyframes) from a self driving car video, which are helpful towards training a semantic segmentation model. In such scenarios, it is important to also consider other signals, e.g. loss, uncertainty, etc., which are dependent on models trained on old representative datapoints (frames). In this section,

we develop a novel formulation for using multiple such criteria. Note that for frame subset selection where labels for the not selected frames are available, we can compute the exact loss, instead of approximating it [6], whereas if the labels are not available, we develop other "uncertainty" criteria (see Section 3.3). Hence, our criteria are of two types, depending on the nature of input data available: active learning and subset selection type, a combination of which can be used for subset selection.

Let M_t be the model trained on R_t. For our formulation, we further define two types of selection criteria: *pairwise and pointwise criteria*. Pairwise dissimilarity criteria $d_{ij}(t) = d((x_i, y_i), (x_j, y_j)|M_t)$ are evaluated on pairs of examples $((x_i, y_i), (x_j, y_j))$ at any point in time. Note that $d_{ij}(t)$ may or may not depend on the current model. For example, in this paper we use SIFT [9] as a measure of dissimilarity, which is related to the perceptual dissimilarity between two images and hence independent of the model. However, one could have also used Fisher kernel [16], which is dependent on an underlying generative model.

Pointwise "dis-quality" criteria $q_i(t) = q((x_i, y_i)|M_t)$ measure an inverse goodness score for a datapoint (x_i, y_i) given a trained model at time t.

In this study, we consider three different dis-quality metrics: negative log-loss denoted as $L_i(t) = L(x_i, y_i|M_t)$, and modified versions of distinctiveness and uncertainty (originally proposed in [5]), called *PW-distinctiveness* $D_i(t) = D(x_i|M_t)$, *PW-uncertainty* $U_i(t) = U(x_i|M_t)$. Note that $L_i(t)$ utilizes the label y_i while $D_i(t)$ and $U_i(t)$ do not. Hence, a combination of SIFT features $d_{ij}(t)$, PW-uncertainty $U_i(t)$ and PW-distinctiveness $D_i(t)$ can be also used for active learning. We define $U_i(t)$ and $D_i(t)$ in Section 3.3. Hence, we define combined pointwise dis-quality score as:

$$q_i(t) = \eta_1 L_i(t) + \eta_2 D_i(t) + \eta_3 U_i(t) \tag{2}$$

where $\eta_i > 0$ and $\sum \eta_i = 1$. η_i are the weights signifying importance of the corresponding metric. For most experiments, we use $\eta_i = \frac{1}{3}$ for $i = 1, 2, 3$. Next, we define the joint pairwise and pointwise dis-quality criteria as:

$$Q_{ij}(t) = \rho d_{ij}(t) + (1 - \rho) q_j(t) \tag{3}$$

where, $0 \leq \rho \leq 1$ is the weightage given to pairwise dissimilarity criteria.

We have setup the joint dis-quality criteria above in a manner such that new datapoints with lower dis-quality score results in better candidates for selection. Hence we pose the final selection problem as:

$$\min_{z^o_{ij}, z^n_{ij}} \sum_{i=1}^{m} \sum_{j=1}^{|R_t|} z^o_{ij} Q_{ij}(t) + \sum_{i,j=1}^{m} z^n_{ij} Q_{ij}(t) + \lambda \sum_{j=1}^{m} \|[z^n_{1,j} \dots z^n_{m,j}]\|_p$$

$$s.t. \quad \sum_{j=1}^{|R_t|} z^o_{i,j} + \sum_{j=1}^{m} z^n_{i,j} = 1$$

$$z^n_{i,j}, z^o_{i,j} \in [0, 1] \tag{4}$$

Here $z^o_{ij} = 1$ or $z^n_{ij} = 1$ implies that jth old or new frame is the representative of ith new frame. For pairwise score, as explained before, a representative j should have the minimum dissimilarity with a new frame i. For pointwise score, the contribution to score function is $\left[\sum_{j=1}^{|R_t|} z^o_{i,j}(1 - \rho) q_j(t) + \sum_{j=1}^{m} z^n_{i,j}(1 - \rho) q_j(t)\right]$. An interesting observation is that when $\rho = 0$, either 0 or 1 new representative points are selected. The new point with lowest value of $q_j(t)$ is selected only when $q_j(t) < \lambda \|[z^n_{1,j} \dots z^n_{m,j}]\|_p$. Hence, these hyperparameter values produce trivial results. In the next section, we describe the modified versions of distinctiveness and uncertainty scores.

3.3. Pointwise distinctiveness and uncertainty

Designing effective criteria for measuring the importance of new incoming datapoints, in the context of a deep model for computer vision has recently been an active area of study [5,6]. Sener and Savarese [6] hypothesizes using an estimate of the loss for a given example as a surrogate. In our case, since the label information is available, we can exactly calculate the loss at a given datapoint. We use $L_i(t)$ as the negative log loss score output by the Deeplabv3+ model [2], since examples with higher loss are more important.

Huang et al. [5] describe two metrics: distinctiveness and uncertainty, for active learning applied to the problem of image categorization. Uncertainty is defined as the sum of Bernoulli variances of class probabilities: $\sum_k p_k(1 - p_k)$,. Datapoints with higher uncertainty are more important. For the semantic segmentation problem, each pixel is associated with a class. So, for a given image x, we define the class probability for a class k as given a trained model M_t: $P_k(x|M_t) = \frac{\text{no. of pixel with predicted class k}}{\text{total no. of pixels in x}}$. Hence, following [5] we define PW-uncertainty as:

$$U_x(t) = -\sum_k P_k(x|M_t)(1 - P_k(x|M_t)) \tag{5}$$

Distinctiveness [5] measures the difference between transformation patterns between model layers A and B evaluated at data point x, $S^{A\to B}_x$, and the corresponding estimate of transformations patterns using representatives based on predicted class, $\hat{S}^{A\to B}_x$. Specifically, for each class k, its representative, c_k is calculated as the centroid of all training images in class k. $S^{A\to B}_x = S^A_x - S^B_x$, where $S^A_x = [\|x - c^A_1\|^2, \dots, \|x - c^A_K\|^2]^T$. $\hat{S}^{A\to B}_x$ is calculated as $\hat{S}^{A\to B}_x = \sum_k P_k(x) S^{A\to B}_{c_k}$.

In case of semantic segmentation, since each pixel is associated with a label, we calculate the centroid image c_k pixelwise, i.e. $c_k[i, j] = \frac{\sum_{(x,y)} \mathcal{I}(y[i,j]=k) x[i,j]}{\sum_{(x,y)} \mathcal{I}(y[i,j]=k)}$. Here we take pixel specific classes since otherwise, all images will have most of the class labels, thereby adding almost all images to all the class specific centroids. Hence, there will not be much discrimination between scores of different classes. $S^{A\to B}_x$ and $\hat{S}^{A\to B}_x$ are calculated as described above. We calculate PW-distinctiveness (following [5]) as:

$$D_x(t) = \frac{\tau(S^{A\to B}_x, \hat{S}^{A\to B}_x) - 1}{2} \tag{6}$$

where, $\tau(x, y)$ is the Kendall's tau coefficient between score lists x and y. Algorithm 1 shows the overall scheme of incremental subset selection. In the next section, we report the experimental results.

4. Experimental results

In this section, we compare the performances of various proposed subset selection criteria, as well as baselines, in the framework of proposed algorithm (Algorithm 1).

4.1. Experimental setup

We use DeepLabV3+ [2] as the semantic segmentation model to be trained on the set of instances. We modify the loss function such that every instance is given weightage proportional to number of points it is representing in the original training set. This ensures that all the incoming sets get equal weightage during training, irrespective of the number of points selected. We train our models for 300 epochs using Adam Optimizer with learning rate varying between 10^{-3} to 10^{-4}.

We report results with the following combinations of criteria for comparison:

Cumulative: The benchmark results with no data reduction. The model is trained on C_t

Algorithm 1 : Incremental Subset Selection Framework.

Input:
R_0: Existing Set of Instances
X_t: Incoming Set of Instances at time t
ρ, λ: Set of parameters for the optimization
M_0: Trained Model on R_0
T: Number of incoming batches
Process:
for t = 1,2, ..., T **do**
  Calculate negative loss function $L_i(t), i \in X_t, R_{t-1}$ using M_{t-1}
  Calculate $D_i(t), i \in X_t, R_{t-1}$ using M_{t-1} (Eqn 6)
  Calculate $U_i(t), i \in X_t, R_{t-1}$ using M_{t-1} (Eqn 5)
  Calculate SIFT distance $d_{ij}, \forall (i,j) \in X_t \times (R_{t-1} \cup X_t)$ (refer Lowe (1999))
  Calculate $q_j(t) \forall j \in X_t, R_{t-1}$ (Eqn 2)
  Calculate $Q_{ij}(t) \forall (i,j) \in X_t \times (R_{t-1} \cup X_t)$(Eqn 3)
  Solve optimization problem of Eqn 4 to get $z^o_{ij} \forall (i,j) \in X_t \times R_{t-1}$ and $z^n_{ij} \forall (i,j) \in X_t \times X_t$
  For each $i \in X_t$ calculate index of representative frame as $\arg\max_j [z^o_{ij} z^n_{ij}]$. If max come from z^n_{ij}, add it to R_{t-1}
  $R_t = R_{t-1}$
  Train the model M_t using R_t
end for
Output:
M_T : Final trained model
R_T : Final reduced dataset

Random: We select the same number of random instances from incoming set as obtained through our formulation, reported as a lower bound.

DU [5]: This is a baseline method described in [5], using PW-distinctiveness and PW-uncertainty criteria designed here for semantic segmentation.

SIFT: Using only SIFT [9] dissimilarity metric d_{ij} and the proposed framework.

SL : Using SIFT, d_{ij}, and Loss $L_i(t)$.

SDU: Using SIFT, d_{ij}, PW-distinctiveness $D_i(t)$, and PW-uncertainty $U_i(t)$.

SDUL: Using all the four criteria: SIFT, PW-distinctiveness $D_i(t)$, PW-uncertainty $U_i(t)$, and loss $L_i(t)$.

Cumulative is the benchmark method, while Random and DU are baseline active learning methods. SIFT and SDU are proposed active learning methods, while SL and SDUL are subset selection methods.

Datasets: We use two datasets for the experiments. The first is a benchmark dataset, Camvid [10], and the second (called CARLA) is prepared using the open-source driving simulator CARLA[1]. CARLA simulates an urban driving scenario where there is an agent vehicle and several other non-player agents like other vehicles, pedestrians, traffic lights and speed signs. We used the CARLA autopilot mode to generate driving video using 14 simulation episodes, at 10 frames per seconds for about 4 h of driving (total 144, 000 frames). For each simulation run, we collect: (1) images from camera attached to the agent vehicle, and (2) the corresponding ground semantically segmented image which comes from the simulator after post-processing the original image. CARLA provides us with 13 classes of objects in the semantically segmented image, e.g. vehicle, pedestrian, roads, roadlines, etc. In order to make computation manageable, we extracted the key frames from the entire data of resolution 256 × 256 and conducted our experiments on them. We collected a total of 4400 frames, out of which we used 52% for the first group of experiments (Section 4.2), 95% for the second group (Section 4.3) and 5% as a common test set for both the groups.[1]

Metrics reported: We have reported two metrics: Global Accuracy and Mean Intersection Over Union (MIoU) computed over the test set for the task of semantic segmentation [17].

Global Accuracy is defined as $\frac{\sum_i N_{ii}}{\sum_i p_i}$ where N_{ii} is the number of pixels of class i correctly predicted to belong to class i and p_i is the total number of pixels belonging to class i.

Mean IoU is defined by $\frac{1}{N_c} \sum_i \frac{N_{ii}}{p_i + \sum_j N_{ji} - N_{ii}}$ where N_c is the number of classes of objects defined for the dataset and N_{ji} is the number of pixels of class j predicted to belong to class i.

4.2. Comparison of methods: Single incoming set

In this section, we compare the performances of different combinations of criteria noted above (also called methods), using subset selection from a single incoming set. Table 1 reports the variation in global accuracy and MIoU under different methods for different compression ratios, evaluated on the two datasets CARLA(top) and CamVid(bottom). For the CARLA dataset, we used an existing set (R_0) comprising of 900 frames, and an incoming set (X_1) of 1000 frames. For the CamVid dataset, we used an existing set (R_0) of 200 frames, and an incoming set (X_1) of 300 frames. We report results for 3 compression ratios (ratio of original to selected frames) – 100:4, 100:9 and 100:15. For random and baseline, we directly select the appropriate number of frames, while for SIFT, SDU, SL and SDUL, we select appropriate number of examples by varying the parameter λ (Eqn. 4), whose values are also reported in the table.

We observe from Table 1 that the method SDUL outperforms all other methods with various metrics. The difference in performance between the methods 'SIFT' and 'SDUL' shows that providing perceptual dissimilarity (using SIFT) alone does not lead to a significant subset, hence leading to lower Accuracy and MIoU. We also notice that the method SDUL performs the close to Cumulative in terms of the reported metrics (3% and 5% difference in terms of MIoU, even for compression ratio of 100:4). Hence we conclude that it is possible to find a significantly smaller subset as replacement for a large set of data for the task of semantic segmentation.

We also compare the proposed active learning methods with the baseline method 'DU' proposed in [5]. We observe that both SIFT and SDU perform better than DU, which involves only PW-distinctiveness and PW-uncertainty. This is due to the absence of consideration of pairwise similarities which is well captured in SIFT and hence combination of all of them surpasses the performance of the baseline method DU. We also note that, as expected, the selection criteria perform better than random.

4.3. Comparison of methods: incremental subset selection

In this section, we compare the best of our proposed methods – *SDUL* with *Random* and *DU*, for incremental subset selection. We perform our experiments in a setup, where the existing set at $t = 1$ had 900 instances, and a batch of new 1000 instances arrive at every timestep. We keep a fixed compression ratio of 100: 15 for each timeframe, leading to increase in count of frames using subset methods, by a factor of 150. Table 2 shows that the performance of Cumulative set increases with time, since number of frames also increases. Alongside, the frames selected by our method SDUL also stays effective. Even though the gap between the performance metrics using Cumulative set and that using our method exists, yet our method proves to be significantly better compared to the Random and DU([5]). Table 2 demonstrates the same where for each

[1] Described in [5].

Table 1
Comparison of different methods for varying compression ratio for CARLA dataset (Top) and Camvid dataset (Bottom): Random and DU[1] are baselines; SIFT and SDU are active learning methods;SL and SDUL are subset selection methods;λ is the hyperparameter for controlling the compression ratio; the numbers in parenthesis denote the difference of Accuracy or MIoU with the benchmark method (Cumulative) .

Method	CARLA data								
	100:4			100:9			100:15		
	λ	%Accuracy	%MIoU	λ	%Accuracy	%MIoU	λ	%Accuracy	%MIoU
Cumulative	-	85.84	51.65	-	85.84	51.65	-	85.84	51.65
Random	-	83.45(2.39)	44.38(7.27)	-	84.86(0.98)	48.45(3.2)	-	84.64(1.2)	46.84(4.81)
DU [1]	-	84.29(1.55)	46.27(5.38)	-	84.99(0.85)	48.37(3.28)	-	85.05(0.79)	48.64(3.01)
SIFT	1.1	84.20(1.64)	46.12(5.53)	0.95	84.60(1.24)	47.17(4.48)	0.91	84.92(0.92)	48.47(3.18)
SDU	0.8	84.77(1.07)	48.32(3.33)	0.67	85.02(0.82)	49.26(2.39)	0.635	85.25(0.59)	49.50(2.15)
SL	0.8	84.83(1.01)	48.11(3.54)	0.7	85.09(0.75)	49.08(2.57)	0.65	85.03(0.81)	49.19(2.46)
SDUL	0.8	**85.10**(0.74)	**48.58**(3.07)	0.68	**85.24**(0.60)	**49.46**(2.19)	0.64	**85.34**(0.50)	**49.58**(2.07)
Method	Camvid data								
	100:4			100:9			100:15		
	λ	%Accuracy	%MIoU	λ	%Accuracy	%MIoU	λ	%Accuracy	%MIoU
Cumulative	-	87.69	43.76	-	87.69	43.76	-	87.69	43.76
Random	-	82.84(4.85)	36.13(7.63)	-	84.32(3.37)	37.62(6.14)	-	84.72(2.97)	37.46(6.3)
DU[1]	-	79.69(8.00)	32.01(11.75)	-	79.56(8.13)	31.56(12.2)	-	82.32(5.37)	32.46(11.3)
SIFT	1.3	82.08(5.61)	37.02(6.74)	1.08	84.92(2.77)	39.97(3.79)	1.03	85.60(2.09)	40.96(2.8)
SDU	0.92	83.47(4.22)	38.70(5.06)	0.77	85.16(2.53)	40.49(3.27)	0.725	85.44(2.25)	40.68(3.08)
SL	0.88	83.51(4.18)	38.84(4.92)	0.738	85.42(2.27)	40.89(2.87)	0.706	85.81(3.27)	41.01(2.75)
SDUL	0.92	**83.51**(4.18)	**39.04**(4.72)	0.762	**85.81**(1.88)	**41.29**(2.47)	0.722	**86.14**(1.55)	**41.37**(2.39)

Table 2
Comparison of different methods in an incremental framework: t denotes the timestep at which the instances arrive; # is the number of instances obtained using the respective method; DU [1] is the baseline method; Random does not use label information during subset selection and hence an active learning paradigm; SDUL is a non-active learning paradigm; numbers in parenthesis denote the difference of Accuracy or MIoU with that of Cumulative Set.

Method	t = 2			t=3			t=4		
	#	%Accuracy	%MIoU	#	%Accuracy	%MIoU	#	%Accuracy	%MIoU
Cumulative	1900	85.84	51.65	2900	86.75	54.53	3900	91.98	58.13
Random	1050	84.64(1.2)	46.84(4.81)	1200	85.69(1.06)	50.93(3.6)	1350	87.26(4.72)	50.22(7.91)
DU [1]		85.05(0.79)	48.64(3.01)		85.71(1.04)	50.76(3.77)		87.40(4.58)	50.98(7.15)
SDUL		**85.34**(0.5)	**49.58**(2.07)		**85.80**(0.95)	**51.57**(2.96)		**90.01**(1.97)	**53.69**(4.44)

timeframe t, we have the cardinalities (#) of the cumulative sets and subsets, followed by the performance metrics obtained after training them using DeepLabV3+ [2].

4.4. Variation w.r.t. ρ and compression ratio

In this section, we observe the change in performance of subset selection by varying the parameters ρ and λ as described in Section 3. Note that change in λ is essentially used to vary the compression ratio. We used the data collected from the CARLA [1] simulator for the following experiments.

Change with compression ratio: Fig. 1-TOP reports the change in Compression ratio, and simultaneously the Global Accuracy and Mean IoU with varying λ. We observe that as the value of λ increases, the number of elements being selected from the incoming set gets reduced, leading to higher compression ratio. We select the same number of instances obtained using SDUL, for finding the subsets using Random and DU [5]. We also train the DeepLabV3+ [2] model on the subsets obtained using the three different methods - SDUL, Random and DU[5] and notice that the instances selected using our method perform significantly better compared to the other two methods.

Change with ρ : Fig. 1-BOTTOM reports the same as above with varying ρ. We observe that as the value of ρ increases, the number of elements selected from incoming set also increases, leading to a lower compression ratio. This is due to the fact that as the weightage factor ρ for SIFT [9] increases, the optimization (Eq. (4)) gets inclined to make an incoming frame, its own representative, leading to a large number of selections. We train the DeepLabV3+ ([2]) model on the selected subsets and observe that our proposed method SDUL performs better compared to the other two methods.

Hence, we observe that with increasing λ, compression ratio increases with a trade-off in global accuracy and mean IOU. Alongside, with increasing ρ, compression ratio decreases and correspondingly, accuracy and mean IOU get affected. Our aim is to have a high compression ratio, with relatively low difference margin to that of Cumulative Set. One can try to keep only SIFT as the metric for optimization, since with higher value of ρ, the accuracy and IOU are almost equivalent to that of the Cumulative Set. However, once we fix a compression ratio and compare all of them, the other three methods with varying combination of metrics (Loss, PW-distinctiveness and PW-uncertainty) perform better than SIFT as observed in Table 1.

4.5. Anecdotal examples

In this section, we show illustrative examples of images taken from CARLA and CamVid datasets. In Fig. 2, we show examples of images from each dataset (belonging to incoming set X_t) and show their representative images (belonging to existing set R_{t-1}). We notice that irrespective of the change in light or weather, a proper representative image gets selected by solving the optimization problem (Eq. (4)) involving all the metrics. In Fig. 3, we show illustrative ground truth semantic segmentation images from CARLA dataset along with its predictions using DeepLabV3+ [2] model.

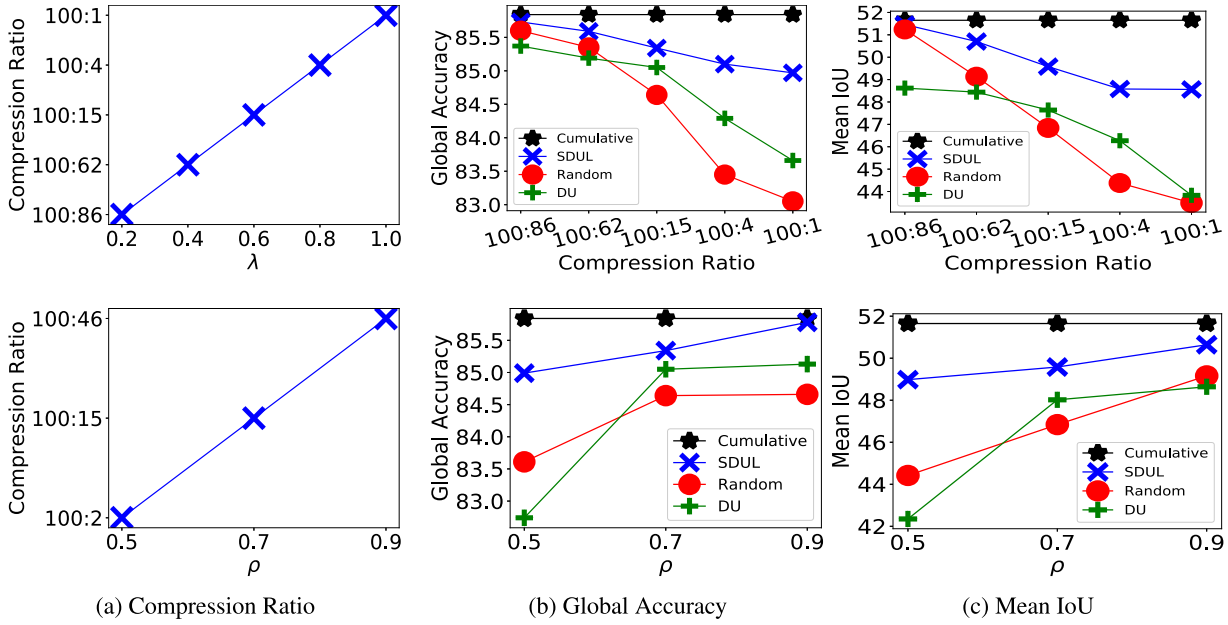

Fig. 1. Change in (a) Compression Ratio (b) Global Accuracy and (c) Mean IoU on varying Compression Ratio (TOP) and ρ (BOTTOM).

Fig. 2. Examples of representative images: (Left to right - Image (a), (b), (c), (d)). Images (a,b) and (c,d) are pairs of original and representative images from CARLA and CamVid datasets..

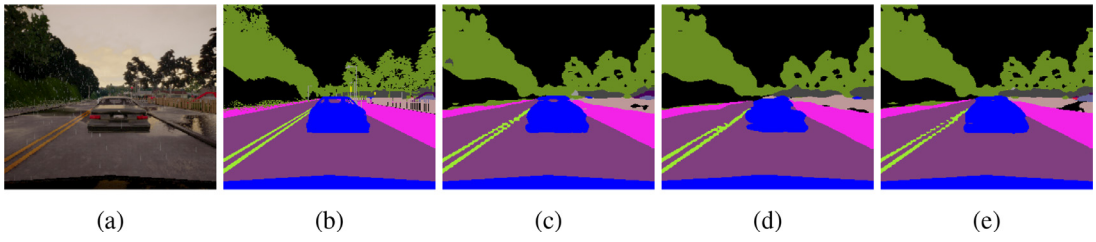

Fig. 3. Examples of semantic segmentation of Image (a): Ground truth (Image b) and predicted semantic segmentation from Cumulative model (Image c) and SDUL-100: 4 (Image d) and SDUL-100: 15 (Image e)..

5. Conclusion

In this paper, we propose a methodology which involves multiple criteria for the subset selection framework. We use two types of criteria in our formulation - pairwise and pointwise, which depend on pairs of datapoints and individual datapoint respectively. We use SIFT [9] as pairwise criteria, and PW-distinctiveness,PW-uncertainty which are the modified versions of distinctiveness and uncertainty proposed in [5] and model loss as pointwise criteria for selection of significant instances. The results show that combination of all the criteria have proved beneficial and perform close to the setting where the entire set is considered.

Declaration of Competing Interest

None.

References

[1] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, V. Koltun, Carla: an open urban driving simulator, arXiv:1711.03938 (2017).
[2] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, ECCV, 2018.
[3] E. Elhamifar, M.C.D.P. Kaluza, Online summarization via submodular and convex optimization., in: CVPR, 2017, pp. 1818–1826.
[4] B. Settles, Active learning literature survey, Technical Report, University of Wisconsin-Madison Department of Computer Sciences, 2009.
[5] S.-J. Huang, J.-W. Zhao, Z.-Y. Liu, Cost-effective training of deep cnns with active model adaptation, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, 2018, pp. 1580–1588.
[6] O. Sener, S. Savarese, Active learning for convolutional neural networks: A core-set approach, in: International Conference on Learning Representations, 2018.
[7] Z. Wang, J. Ye, Querying discriminative and representative samples for batch mode active learning, ACM Transactions on Knowledge Discovery from Data (TKDD) 9 (3) (2015) 17.
[8] B. Du, Z. Wang, L. Zhang, L. Zhang, W. Liu, J. Shen, D. Tao, Exploring representativeness and informativeness for active learning, IEEE Trans. Cybern. 47 (1) (2017) 14–26.

[9] D.G. Lowe, Object recognition from local scale-invariant features, in: iccv, Ieee, 1999, p. 1150.
[10] G.J. Brostow, J. Fauqueur, R. Cipolla, Semantic object classes in video: a high--definition ground truth database, Pattern Recognit. Lett. 30 (2) (2009) 88–97.
[11] D.D. Lewis, W.A. Gale, A sequential algorithm for training text classifiers, in: SIGIR94, Springer, 1994, pp. 3–12.
[12] S. Huang, R. Jin, Z. Zhou, Active learning by querying informative and representative examples., IEEE Trans. Pattern Anal. Mach. Intell. 36 (10) (2014) 1936.
[13] H. Wang, L. Du, P. Zhou, L. Shi, Y.-D. Shen, Convex batch mode active sampling via α-relative pearson divergence, in: Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.
[14] Y. Gal, R. Islam, Z. Ghahramani, Deep bayesian active learning with image data, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 1183–1192.
[15] E. Elhamifar, G. Sapiro, S.S. Sastry, Dissimilarity-based sparse subset selection, IEEE Trans. Pattern Anal. Mach. Intell. 38 (11) (2016) 2182–2197.
[16] T. Jaakkola, D. Haussler, Exploiting generative models in discriminative classifiers, in: Advances in Neural Information Processing Systems, 1999, pp. 487–493.
[17] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.