# Large scale labelled video data augmentation for semantic segmentation in driving scenarios

Ignas Budvytis[1], Patrick Sauer[2], Thomas Roddick[1], Kesar Breen[1], Roberto Cipolla[1]
[1] Department of Engineering, University of Cambridge, UK
[2] Toyota Motor Europe, Belgium
{ib255@eng.cam.ac.uk,patrick.sauer@toyota-europe.com,tr346@eng.cam.ac.uk,
kesar.breen@gmail.com,rc10001@eng.cam.ac.uk}

## Abstract

*In this paper we present an analysis of the effect of large scale video data augmentation for semantic segmentation in driving scenarios. Our work is motivated by a strong correlation between the high performance of most recent deep learning based methods and the availability of large volumes of ground truth labels. To generate additional labelled data, we make use of an occlusion-aware and uncertainty-enabled label propagation algorithm [8]. As a result we increase the availability of high-resolution labelled frames by a factor of 20, yielding in a 6.8% to 10.8% rise in average classification accuracy and/or IoU scores for several semantic segmentation networks.*
*Our key contributions include: (a) augmented CityScapes and CamVid datasets providing 56.2K and 6.5K additional labelled frames of object classes respectively, (b) detailed empirical analysis of the effect of the use of augmented data as well as (c) extension of proposed framework to instance segmentation.*

## 1. Introduction

Semantic segmentation is one of the most important subproblems of autonomous driving. Its progress has been accelerated by the developments in the state-of-the-art in image classification [15, 32] and advances in training and inference procedures (e.g. dropout or batch normalisation) as well as architectural innovation in deep learning in general. However, in contrast to image classification and some deep learning lead problems of computer vision, semantic segmentation (especially for autonomous driving) still operates on limited size datasets which do not exceed 5000 labelled frames [10]. As labelling by hand takes approximately 1 hour per single frame, alternative methods for obtaining dense labelled data for semantic segmentation must be employed.

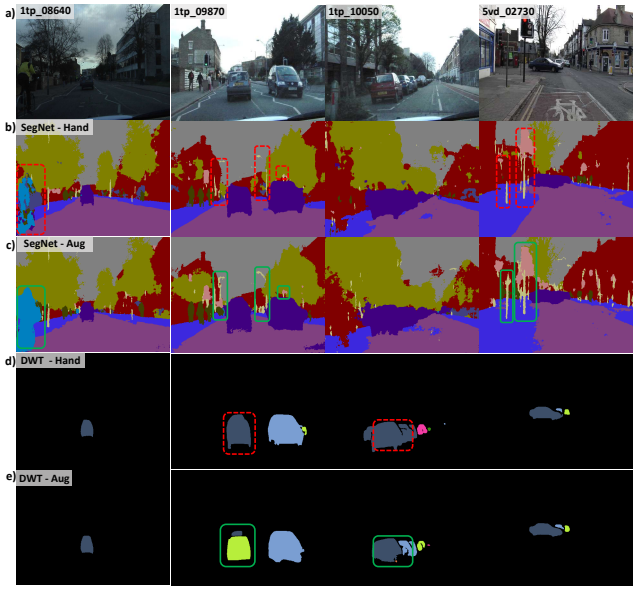


Figure 1. This figure illustrates the effect of training semantic class and instance segmentation networks on hand labelled and on augmented data. Row (a) contains four images from CamVid [6] dataset. Rows (b) and (c) show corresponding per pixel class predictions of SegNet [3] trained on original hand labelled data (b) and on augmented label data (c). Similarly the bottom two rows show instance predictions using the Deep Watershed Transform Network [2]) trained on hand labelled (d) and on augmented (e) data. Note the increased accuracy on small classes such as poles or road signs for class segmentation and the increased separation of cars for instance segmentation.

In this work we propose using a simplified version of the label propagation algorithm of [8] in order to increase the quantity of available ground truth labels by an order of magnitude. The chosen label propagation algorithm handles occlusions and label uncertainty elegantly, which is essential in order to avoid generating erroneous labelled data.

We evaluate the benefits of our proposed data augmentation procedure on two standard datasets for semantic segmentation: CityScapes [10] and CamVid [6]. We observe that for many models the use of augmented data leads to a significant increase in performance, and even in the worse case does not lead to a reduction larger than 2%. For some networks, such as E-Net [28], Dilation Network [36], SegNet [2] and PSPNet [37] an increase of 10.8%, 8.2%, 6.6% and 4.1% in averge classification accuracy and/or IoU score is observed - significantly larger than previously reported [26]. Qualitatively the most notable improvement is in increase of class accuracy of smaller classes as illustrated in Figure 1 and reported in Section 5.

We also extend our data augmentation framework and corresponding analysis to to instance segmentation (see Figure 1 and Section 5.3). As a part of our evaluation effort we include the CamVid-Instance dataset consisting of instance labels for people and cars from the original CamVid [6] dataset.

We proceed with a brief overview of related work in Section 2. We explain our data augmentation algorithm based on a simplified tree structured graphical model [8, 1] in Section 3. We then provide details of our experimental setup, results and analysis in Sections 4 and 5. Final remarks are provided in Section 6.

## 2. Related work

In this section we examine the pros and cons of various approaches aiming to directly or indirectly increase the quantity of label data.

**Larger datasets.** CamVid [6] and Leuven [18] datasets are among the earliest mid-sized datasets for semantic segmentation in autonomous driving, providing up to 701 hand labelled frames. Over the last four years we have seen a rapid increase of datasets covering driving scenarios such as the Ford Campus Vision and Lidar Dataset [27], KITTI Dataset [13], Daimler Urban Segmentation [31] with the largest so far being the CityScapes Dataset [10] providing 5000 hand labelled frames. The recently released Oxford Robotcar dataset [23] contains an unparalleled number of image frames on the order of 20 million, but no ground truth semantic segmentation labels are currently available.

**Label Propagation.** Video-based label propagation algorithms [8, 1, 26, 20] use fully or partially (e.g. paint strokes) labelled frames and propagate labels across the video via established frame to frame correspondences. Label propagation algorithms vary in (a) their methods of establishing correspondences between neighbouring frames, (b) the label inference method used and (c) the choice of unaries. Frame to frame correspondences are often established with optical flow [26] or patch matching [8]. However, as suggested in [1], unless the optical flow has high accuracy occlusion awareness, it is likely to propagate erroneous labels near occlusion boundaries. The most frequently used inference schemes include marginal posterior inference [8], sliding window inference [33], continuous MRF [14], shortest path calculation [4] and max-marginals [17]. Among these the marginal posterior inference produces the most intuitive segmentations by increasing predicted label uncertainty further away from labelled data in a chosen model (as explained in [7]). Finally, unaries are often provided via CRF [20], Random Forests [8] or other classifiers. However these require human interaction to provide highly accurate results. It is also important to note that while there are a plethora of label propagation algorithms, very few have evaluated the application of propagated labels for classifier training at scale. Exceptions include [26] and [1], however large gains in supervised training scenarios have not yet been demonstrated.

**Label Transfer.** Label transfer methods are very similar to label propagation methods in video, but they often use a different type of data as intermediate representation. For example, a 3D reconstruction of a scene [35, 34] can be used to achieve similar results to labels propagated in videos. Such an approach has benefits at occlusion boundaries (esp. if non-vison based 3D data is used), but may suffer lower accuracies at labelling small classes. The results of [35] are promising, however as no images or labelled frames have been disclosed, no comparison is possible. Another example of a label transfer approach is the leveraging of aerial images [25] where labelling some classes (e.g. side-walk or road) at scale turns out to be more efficient.

**Artificial Data.** Using artificial data [24, 9] provides another alternative for obtaining large amounts of high quality labelled data. Artificial datasets such as Synthia [30] and Virtual KITTI [12] seem to be on the rise. Despite the low cost, the direct value of such data for popular semantic segmentation benchmarks is yet to be proven. Key challenges remain obtaining photo-realistic images and modelling real world scenes, yet rapid progress has been demonstrated.

**Other.** Other approaches include the use of weak labels [22] as well as additional sensors such as 3D point clouds [11] or point supervision [5] by humans.

## 3. Label Propagation

In this section, we describe the model, inference and implementation details of the label propagation method used for data augmentation.

### 3.1. Model

We use a simplified version of the model of [8] in which parameter learning (including class unaries) and variational inference are avoided. We define a joint probability of pixel class labels as follows:

$$P(Z) \propto \prod_{\forall k,p,j} \Psi(Z_{k+d,T_{k+d,p}(j)}, Z_{k,p(j)}) \qquad (1)$$

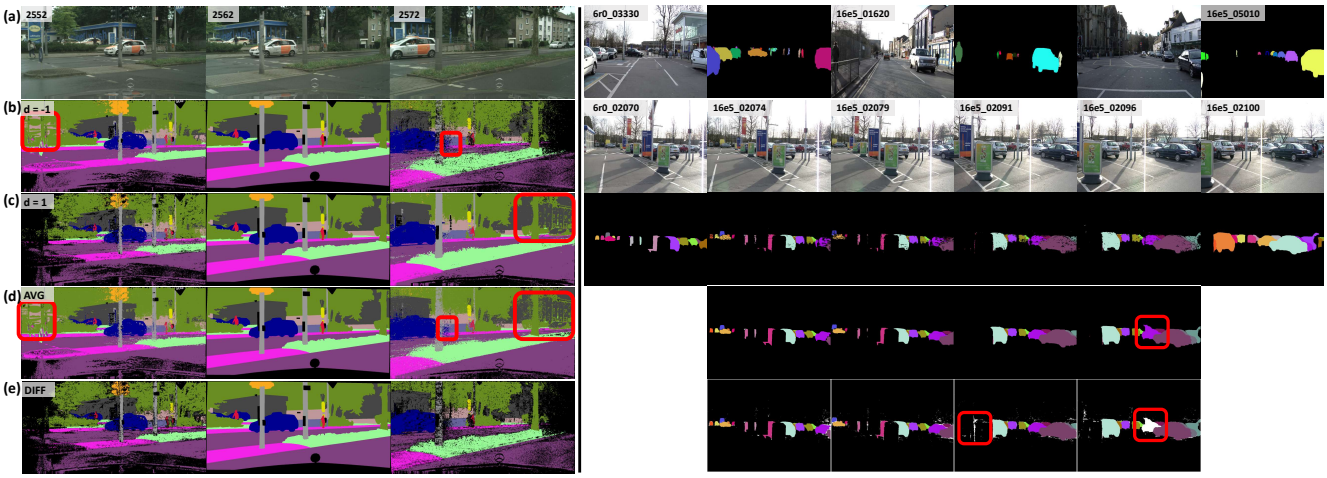


Figure 2. The left section of this figure illustrates the effect of uncertainty averaging and differencing of noisy propagated labels. Row (a) contains three images from the Bochum city sequence (CityScapes), of which the middle frame (2562) has ground truth labels provided. Rows (b) and (c) show propagation results for backward-built factor graph ($d = -1$) and forward-built factor graph ($d = 1$) respectively. Row (d) shows combined output produced by uncertainty averaging of (b) and (c). Row (e) shows combined output produced by label differencing of (b) and (c). Red squares indicate regions with erroneous pixel labels. Note how they are transferred to the output of uncertainty averaging. Row (a) on the right section of this figure contains three image and corresponding instance label pairs of people and car instances from CamVid-Instance dataset. Rows (b-e) show sequence of images from CamVid dataset (b), initial propagation output (c), dilated propagation output (d) and final labels obtained after manual clean up step (e). Note that white noise pixels correspond to the uncertain ("void") pixels from instance propagation which take neither background nor any of the instance labels.

where $Z$ is a set of discrete random variables $Z_{k,p(j)}$ taking values in the range $1..L$ corresponding to the class label of a pixel $j$ in a patch $p$ of frame $k$. Here $\Psi$ is a potential favouring same class prediction

$$\Psi(a,b) = \begin{cases} 1-\delta, & \text{if } a = b \\ \delta, & \text{otherwise.} \end{cases} \quad (2)$$

Furthermore $Z_{k+d,T_{k+d,p}(j)}$ corresponds to a class label of a pixel $j$ in a patch $T_{k+d,p}$ in frame $k+d$. Here $T_{k+d,p}$ corresponds to the best matching patch of frame $k+d$ to patch $p$ in frame $k$. Finally, $d$ is a constant which builds correspondences from the current frame to the previous frame or to the next frame when set to $-1$ and 1 respectively.

### 3.2. Inference

As each pixel is restricted to have exactly one best match for each pixel, the aforementioned joint distribution can be represented as a tree-structured factor graph. As a result, the exact inference of the marginal posterior for each variable $P(Z_{k,p(j)} = l)$ can be performed using message passing. The final per pixel class distributions are obtained by summing over distributions of overlapping pixels as follows

$$R(k,i,l) = \frac{1}{K} \sum_{s.t.p(j)=i} P(Z_{k,p(j)} = l) \quad (3)$$

where $K$ is a normalisation constant.

### 3.3. Implementation

The data augmentation algorithm is has two phases. During the first phase neighbouring frame correspondences are calculated by finding the highest cross-correlation score of a patch $p$ in window $W \times H$ around this patch in frame $k+d$ as detailed in [8]. During the second phase obtained correspondences are used to calculate per-pixel marginal posterior distributions as described in the previous section. Processing steps performed for semantic class labels and for instance label propagation are described below.

**Class label augmentation.** To obtain class labels for training, we perform three steps. First, for each pixel $i$ in frame $k$, we assign the most likely class label $\arg\max_{l'} R(k,i,l')$. For pixels where the most likely label has a probability lower than a threshold $\frac{1}{L} + 0.0001$ we assign the "void" label to avoid mislabelling caused by numerical instability. Examples of labels for $d = -1$ and $d = 1$ for one sequence from the Bochum dataset (CityScapes) are presented in rows (b) and (c) in Figure 2. Unlike in [8, 1], we produce the final result by taking a label image difference (i.e. assigning a class label if both frames agree and a "void" label if they disagree) as opposed to averaging the backward ($d = -1$) and forward ($d = 1$) built structures. Example comparisons between using image differencing and averaging can be found in rows (e) and (d) respectively of the left section of Figure 2. Although more pixel labels are obtained when using averaging, taking an image differ-

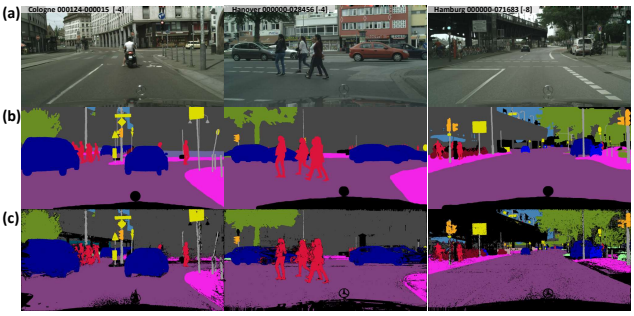


Figure 3. This figure provides a qualitative evaluation of augmented labels. In particular, row (a) contains three images extracted at varying distances (4 or 8) from a seed labelled frame provided in CityScapes dataset. Row (b) contains ground truth labels obtained by us. Row (c) illustrates corresponding augmented frames. A high qualitative match is achieved. See Table 2 for quantitative evaluation.

ence reduces erroneous labelling introduced by occlusions, dis-occlusions or erroneous patch correspondences.

**Instance label augmentation.** To obtain instance labels, we follow a similar procedure as in class label propagation. One notable difference is that we assign all pixels of non-instances to a background class and perform two steps of clean up (see Figure 2). During the first step we dilate all non-instance pixels which are surrounded by labels of one particular instance and fill small (less than 20 pixels) instance regions which reside within another instance with the labels of the surrounding region. During the second step we go through the generated labels and manually mark instances with severly wrong labels as "void". Note that clean-up of 6.5K of frames took no more than 4 hours of manual labour and can be significantly improved with more sophisticated tools.

# 4. Experiment Setup

In this section we describe experiments on three datasets: CamVid [6], CityScapes [10] and the novel CamVid-Instance dataset. For all datasets we use the experimental protocol described in Section 3 unless stated otherwise. Below we provide more details of each dataset and corresponding steps taken to produce augmented labels.

## 4.1. CamVid dataset

The CamVid Dataset [6] consists of 701 labelled images: 367 for training, 233 for testing and 101 for validation, covering a total of 10 minutes of 30Hz video of driving in Cambridge, UK. Labels are provided every 30 frames for the training and testing set and every 2 frames for the validation set. A total of 32 label types related to autonomous driving (road, side-walk, car, etc.) are provided, however due to low representation of rare classes, most studies have focused on evaluating classifiers on a subset containing 11 classes.

To obtain label propagation results we calculated the correspondences for every pair of neighbouring labelled frames. We then performed label propagation for the 11 classes, representing the void class via a uniform distribution. Inference took on average of 5 seconds per frame at full resolution of $960 \times 720$. The window size ($W \times H$) for establishing correspondences was set to $140 \times 100$, $\delta$ to $0.001$.

We obtained propagated labels for all the images in the train, test and validation datasets, however we use only the labels in the training dataset in the experiments described here.

## 4.2. CityScapes dataset

The CityScapes dataset consists of 5000 densely labelled frames and 20000 coarsely labelled frames of $2048 \times 1024$ resolution. Each densely labelled frame is surrounded by 30 unlabelled adjacent video frames. We performed label propagation for 20 surrounding frames resulting in a total of 62475 annotated frames using the protocol described in Section 4.1. 5% of the annotated frames were filtered manually. Key modes of failure included label leakage caused by large object displacement and ego-motion as well as sudden change in lighting.

To evaluate the quality of the labelling, we first sampled 9 frames from the CityScapes training dataset at random at four different locations (-8,-4,+4 and +8) from the seed label (see Figure 3). We then hand labelled the selected images following the original protocol of CityScapes [10]. As shown in Table 2, all the classes had accuracy higher than 70%, except for the *bus* class.

## 4.3. CamVid-Instance Dataset

For our third set of experiments we augmented the original CamVid [6] dataset with instance level annotation of people and cars. Several examples of instance labels obtained are shown in Figure 2.

In order to obtain instances, we used original boundaries of cars and people, only introducing new boundary annotations where two instances of the same class were overlapping. The number of instances in a single frame ranged from 0 to 27. More examples of CamVid-Instance dataset images can be found in the supplementary material.

## 4.4. Model Training

In this section we provide brief training details of various models for class and instance segmentation.

**Class segmentation.** For our experiments on the CamVid dataset we trained six commonly used segmentation architectures: FCN [21], SegNet [2], Bayesian SegNet [2], E-Net [28], Dilation Network [36] and PSPNet [37] on both

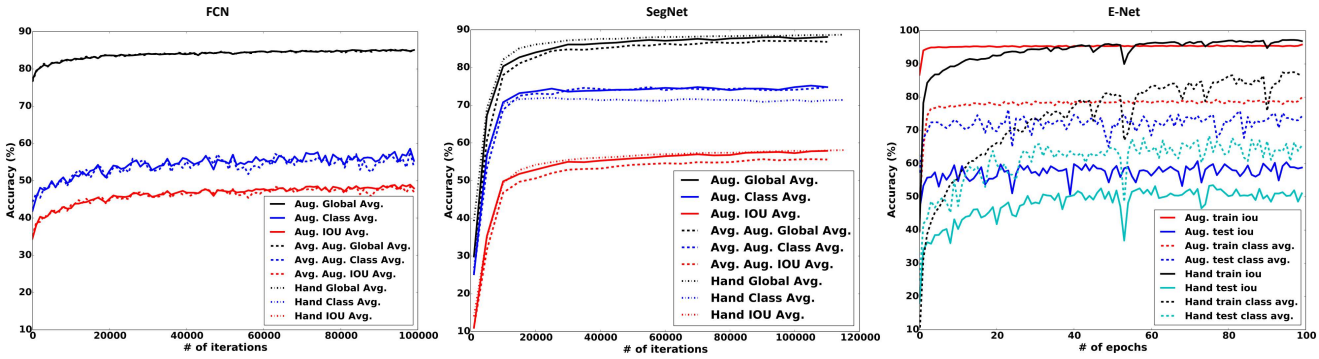


Figure 4. Three graphs in this figure compare segmentation network performance using global accuracy, class average accuracy and average IoU measures for results obtained when training on the original hand labelled CamVid data ("Hand") as well as on augmented labels obtained from uncertainty averaging ("Avg.") and label differencing ("Aug."), as explained in Section 3.3. The FCN network (left) does not benefit from additional hand labels in our experiments. Broadly this corresponds to the findings reported in [26]. On the other hand, Segnet [2] (middle) trained on augmented data produced using label differencing shows a significant increase in either class average or IoU when compared to Avg. or Hand labels. The graph on the right shows class accuracy and IoU scores on train and test datasets for E-Net [28] for the first 100 epochs. The high (and increasing) average class accuracy on training data for E-Net trained on hand labels indicates overfitting and explains the large class average and IoU score differences on the test data.

hand labelled as well as on augmented data. We chose the aforementioned models due to their varying design choices for upsampling layers, easily accessible code and high performance.

In order to reduce the complexity of setup we trained and tested Dilation Network without the context module. Similarly the input resolution of PSPNet [37] was halved ($353 \times 353$).

Except where mentioned otherwise, we used the original code and parameters provided by the authors of the corresponding networks. For a more fair comparison of networks no pre-training was used. For all the experiments on CamVid dataset we used input images of $480 \times 360$ resolution. Quantitative experiment evaluation and its analysis is presented in Section 5.1.

We repeated the same exercise for four models on the CityScapes dataset. We excluded Bayesian SegNet and FCN due to time constraints. Note that the aim of this work is to demonstrate the value of using augmented data and not to outperform state-of-the-art benchmarks.

**Instance segmentation.** For our experiments on instance segmentation we used the DeepMask [29] and Deep Watershed Transform [3] networks. We used original DeepMask implementation provided by the authors and implemented our own version of Deep Watershed Transform method, for which we used SegNet as a classifier at train and test time. We also simplified the post-processing steps to deleting the lowest watershed prediction level then growing the connected components back with dilation of one pixel radius. In order to account for a non-perfect classifier at train time we trained our watershed network to predict a watershed energy of zero on erroneously labelled background pixels.

## 5. Results

In this section we present the quantitative and qualitative results of our experiments on CamVid, CityScapes and CamVid-Instance datasets.

### 5.1. CamVid Dataset

Figure 4 illustrates the evolution of test accuracies during training of FCN [21], SegNet [2] and E-Net [28] on the CamVid dataset with and without data augmentation. It highlights the advantage of using augmented labels obtained by frame differencing as opposed to uncertainty averaging. It also indicates the potential of using augmented data to prevent overfitting. See the caption of Figure 4 for more details.

Table 1 compares six different state-of-the-art-models trained on hand labelled and on augmented data. Per class accuracies as well as average classification accuracy, global accuracy and mean intersection over union (mean IoU) scores are reported. Note that matching previously reported accuracies is not straight-forward hence we include an evaluation of our own. Each network was trained either until convergence or for an equal number of iterations with its counterpart.

Overall we have found that using augmented labels often led to a significant increase (4% or more) with only a slight decrease (less than 1.5%) in the worst case. A particularly large increase in both average class accuracy (10.8%) and IoU score (9.8%) was observed for E-Net and in average class accuracy (6.6%) for SegNet. The slight

| Method | Type | Data | # of iter. | Building | Tree | Sky | Car | Sign | Road | Pedestrian | Fence | Pole | Sidewalk | Bicyclist | Class avg. | Global acc. | Class IoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SegNet [2] | R | H | N/A | 88.0 | 87.3 | 92.3 | 80.0 | 29.5 | 97.6 | 57.2 | 49.4 | 27.8 | 84.8 | 30.7 | 65.9 | 88.6 | 50.2 |
| | T | H | 184K | 89.6 | 83.6 | 93.9 | 79.0 | 47.3 | 95.9 | 69.6 | 37.2 | 34.4 | 88.8 | 29.7 | 68.2 | 88.8 | 58.5 |
| | T | A | 184K | 83.7 | 86.8 | 94.6 | 82.4 | 59.0 | 92.8 | 83.9 | 48.7 | 46.2 | 93.5 | 51.1 | **74.8** | 88.1 | **59.8** |
| Bayesian SegNet [16] | R | H | N/A | 80.4 | 85.5 | 90.1 | 86.4 | 67.9 | 93.8 | 73.8 | 64.5 | 50.8 | 91.7 | 54.6 | 76.3 | 86.9 | 63.1 |
| | T | H | 129K | 76.8 | 89.1 | 92.2 | 87.4 | 74.7 | 94.7 | 90.4 | 63.8 | 64.5 | 90.9 | 74.0 | 81.7 | 87.3 | 59.1 |
| | T | A | 404K | 82.9 | 85.1 | 94.8 | 85.4 | 68.4 | 95.3 | 93.0 | 55.5 | 59.7 | 93.7 | 68.1 | 80.2 | 89.0 | **61.7** |
| E-Net [28] | R | H | N/A | 74.7 | 77.8 | 95.1 | 82.4 | 51.0 | 95.1 | 67.2 | 51.7 | 35.4 | 86.7 | 34.1 | 68.3 | N-P | 51.3 |
| | T | H | 40K | 75.5 | 82.9 | 95.9 | 79.5 | 43.8 | 95.2 | 28.8 | 37.5 | 39.2 | 85.1 | 31.9 | 63.2 | 84.7 | 51.1 |
| | T | A | 100K | 84.8 | 84.7 | 97.0 | 83.2 | 49.2 | 96.0 | 69.0 | 43.0 | 48.1 | 92.1 | 66.7 | **74.0** | 89.1 | **60.9** |
| FCN [21] | R | H | N/A | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | 64.2 | 83.1 | 52.0 |
| | T | H | 100K | 88.8 | 80.8 | 92.8 | 72.1 | 31.7 | 95.4 | 39.3 | 17.8 | 12.4 | 70.3 | 23.7 | 56.8 | 85.0 | 48.6 |
| | T | A | 100K | 89.9 | 80.8 | 93.3 | 69.3 | 29.5 | 95.3 | 32.3 | 16.4 | 11.8 | 69.7 | 20.4 | 55.4 | 85.1 | 48.0 |
| Dilation Network [36] | R* | H | N/A | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | 65.3 |
| | T | H | 100K | 88.4 | 83.8 | 94.7 | 84.6 | 49.5 | 96.2 | 59.9 | 36.7 | 19.7 | 84.3 | 43.1 | 67.3 | 88.3 | 55.6 |
| | T | A | 100K | 87.4 | 86.9 | 93.4 | 86.5 | 51.8 | 94.5 | 58.5 | 38.8 | 20.2 | 86.8 | 44.4 | 68.1 | 88.1 | 55.6 |
| PSPNet [37] | R | H | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P | N-P |
| | T | H | 300K | 88.8 | 89.0 | 93.7 | 88.9 | 44.3 | 97.0 | 73.2 | 40.7 | 29.7 | 89.6 | 59.4 | 72.2 | 90.0 | 62.3 |
| | T | A | 300K | 88.9 | 83.8 | 97.2 | 84.3 | 49.1 | 96.4 | 60.8 | 50.5 | 26.5 | 91.7 | 56.4 | 71.4 | 90.0 | 62.5 |

Table 1. This table provides a quantitative evaluation of six different methods trained on hand labelled (H) and augmented (A) data from CamVid [6] dataset. The number of iterations used to train each network is reported along with per-class segmentation accuracies, average classification accuracy, global accuracy and average per-class intersection over union (IoU) metric. All methods were trained on images of $480 \times 360$ resolution. Corresponding results of each network reported in the literature are also listed in rows marked with type R (reported). However, due to the different strategies followed in training and optimising deep networks, we encourage the reader to focus on the relative performance difference between the results we report. Large increases in class average and/or average IoU metrics are observed for E-Net and Segnet when trained on augmented data. The performance of Bayesian SegNet, FCN, Dilation Network and PSPNet is largely unaffected. * - note that a significant difference in previously reported accuracies [36] and the ones obtained by us of Dilation Network may be explained by the use of lower resolution images instead of $640 \times 480$.

| Method | Train Data | # of iter. | Road | Sidewalk | Building | Wall | Fence | Pole | Traffic Light | Traffic Sign | Vegetation | Terrain | Sky | Person | Rider | Car | Truck | Bus | Train | Motorcycle | Bicycle | Class avg. | Global acc. | Class IoU |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Augmented Labels | N/A | N/A | 97.0 | 87.4 | 95.5 | 80.3 | 73.7 | 72.8 | 86.3 | 86.9 | 89.3 | N/A | 95.4 | 88.7 | 85.4 | 94.0 | 93.9 | 42.4 | 92.5 | 77.0 | 72.5 | 84.0 | 89.9 | 71.7 |
| SegNet [2] | H | 182K | 96.4 | 83.8 | 88.6 | 40.5 | 38.6 | 62.2 | 52.6 | 63.7 | 89.0 | 69.8 | 97.2 | 84.7 | 45.2 | 92.8 | 48.7 | 57.8 | 33.1 | 25.7 | 78.0 | 65.7 | 89.7 | 50.6 |
| | A | 189K | 95.0 | 86.4 | 87.5 | 54.0 | 52.9 | 67.2 | 69.6 | 76.5 | 88.5 | 73.3 | 97.5 | 85.7 | 52.5 | 94.8 | 61.2 | 68.1 | 53.3 | 30.4 | 79.0 | **72.3** | 89.8 | **53.9** |
| E-Net [28] | H | 40K | 97.3 | 87.6 | 91.4 | 40.2 | 59.6 | 61.5 | 45.9 | 63.3 | 93.7 | 67.4 | 97.3 | 76.5 | 49.7 | 95.1 | 61.5 | 52.3 | 69.1 | 13.3 | 72.9 | 68.2 | 91.9 | 55.0 |
| | A | 100K | 96.2 | 86.7 | 91.9 | 48.8 | 56.3 | 53.0 | 53.9 | 67.3 | 94.9 | 74.1 | 97.3 | 77.0 | 40.8 | 95.4 | 47.4 | 80.1 | 9.31 | 12.0 | 75.3 | 66.2 | 91.8 | 53.8 |
| Dilation Network [36] | H | 24K | 96.8 | 73.7 | 86.4 | 34.0 | 63.4 | 24.7 | 0.8 | 13.8 | 93.4 | 63.2 | 91.4 | 47.9 | 15.5 | 88.1 | 48.7 | 49.8 | 58.4 | 3.3 | 57.2 | 53.2 | 87.6 | 41.7 |
| | A | 68K | 97.8 | 77.0 | 91.6 | 35.9 | 45.2 | 28.3 | 12.5 | 37.3 | 94.0 | 64.6 | 93.5 | 70.1 | 31.4 | 93.0 | 57.2 | 72.9 | 50.2 | 47.1 | 64.8 | **61.3** | 90.4 | **49.9** |
| PSPNet [37] | H | 400K | 98.3 | 84.0 | 94.3 | 8.8 | 41.3 | 46.2 | 52.9 | 63.0 | 91.9 | 58.8 | 93.2 | 75.2 | 57.5 | 93.5 | 43.2 | 66.1 | 0.6 | 4.4 | 82.0 | 60.8 | 91.5 | 51.9 |
| | A | 400K | 98.6 | 79.4 | 95.8 | 34.1 | 56.5 | 35.9 | 43.6 | 54.0 | 96.5 | 58.6 | 92.6 | 82.7 | 52.3 | 95.3 | 58.8 | 60.4 | 0.00 | 20.6 | 79.1 | **62.9** | 92.8 | **56.0** |

Table 2. The first row of this table contains the evaluation of augmented labels against hand labelled frames at various offsets from the seed label frame as illustrated in Figure 3. The rest of this table contains quantitative evaluation of four different methods trained on the hand labelled (H) and augmented (A) data from the CityScapes dataset. Similar metrics as in Table 1 are reported. The validation portion of the dataset is used for testing as the submission frequency of the CityScapes benchmark is highly limited. SegNet, Dilation Network and PSPNet show significant increase in class average accuracy and IoU metric scores when trained on augmented data.

decrease (less than 1.2%) in both metrics observed for the FCN network can be explained by its limited ability in handling small object classes at low resolution. A slight decrease in IoU matched by a higher class average observed for Bayesian SegNet is likely to have been caused by an unsaturated training requiring a significantly larger number of iterations for convergence due to the use of dropout in training.

## 5.2. CityScapes Dataset

Table 2 provides a quantitative analysis of SegNet [2], E-Net [28], Dilation Network [36] and PSPNet [37] tested on the validation portion of CityScapes dataset. SegNet and Dilation Network show a large increase in class average accuracy (6.6% and 8.1% respectively) and in IoU (3.3% and 8.2%). The increased performance of Dilation Network can be explained by existence of more pronounced small classes in CityScapes dataset. PSPNet shows a moderate increase

| Network | Data | Iter | AP-0.5-ALL-100 | AP-0.5:0.95-ALL-100 |
|---|---|---|---|---|
| DeepMask [29] | H | 50K | 0.139 | **0.095** |
| | A | 200K | **0.143** | 0.094 |
| DWN [3] | H | 188K | 0.207 | 0.082 |
| | A | 220K | **0.252** | **0.099** |

Table 3. This table compares DeepMask [29] and Deep Watershed [3] networks when trained on hand labelled (H) and on augmented (A) instance labels. Metrics of average precision at IoU score threshold of 0.5 and at multiple thresholds (0.5-0.95) are calculated using MSCOCO evaluation API [19].

in both class average (2.1%) and IoU (4.1%). Note while PSPNet does not reach reported [37] accuracy, maximising overall performance is out of the scope of this work. Multiple experimental changes could have contributed to lower than reported performance: no pre-training was used, network input and image resolution was halved, only 1 week of a single GPU training time had been dedicated. Finally, note that a slightly decrease in the IoU and average class accuracy of E-Net may indicate the limitations of this very compact model when trained on a significantly more complex dataset than CamVid.

### 5.3. Instance Segmentation

Table 3 and Figure 5 show quantitative and qualitative comparison of Deep Watershed [3] and DeepMask [29] instance segmentation networks when trained on hand labelled and on augmented data. Standard average precision metrics [19] under different IoU thresholds (as shown in Table 3) are used to perform the quantitative evaluation.

As with semantic segmentation experiments both metrics (*AP-0.5-ALL-100* and *AP-0.5:0.95-ALL-100*) show relatively larger increase (from 0.207 to 0.252) than decrease (from 0.095 to 0.094) when augmented data is used for training. Moreover a larger increase seen in both average precision metrics for Deep Watershed Transform Network translates to a qualitatively better separation of car instances shown in Figure 5.

## 6. Conclusion

In this paper we presented the analysis of the effect of large scale labelled video data augmentation for semantic segmentation in driving scenarios. For some networks, such as E-Net [28], SegNet [2] or Dilation Network [36] we observed increases of either average class accuracy or intersection over union (IoU) score in range from 6.6% to 10.8%. We also demonstrated the potential for using augmented data to improve the accuracy of instance level segmentation. The augmented data for CityScapes [10], CamVid [6] and CamVid-Instance datasets will be made available to the research community for further evaluation.

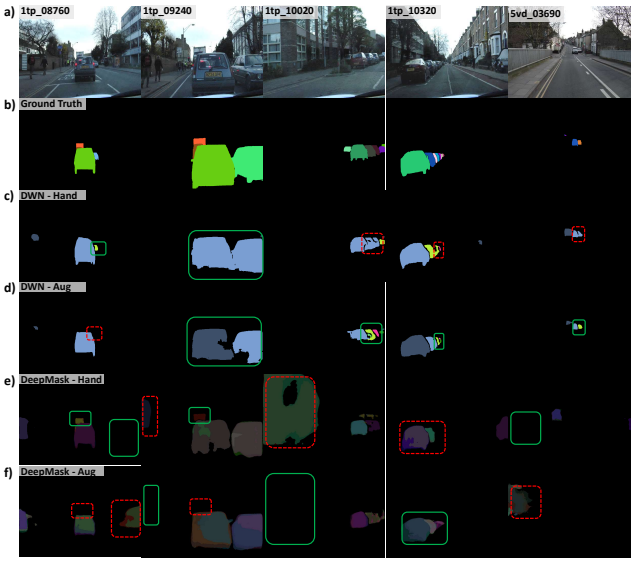


Figure 5. This figure provides a qualitative comparison of Deep Watershed [3] (rows c,d) and Deep Mask [29] (rows e,f) network predictions when trained on hand labelled and on augmented data. CamVid-Instance dataset is used for testing and training. Failures and successes in car separation as well as car segmentation are marked in green and red boxes correspondingly. Training the Deep Watershed Network [3] with augmented labels results in improved nearby car instance separation. The effect of augmented data is less pronounced for DeepMask [29] network which corresponds well to smaller quantitative differences in average precision metrics reported in Table 3.

## References


[1] V. Badrinarayanan, I. Budvytis, and R. Cipolla. Semi-supervised video segmentation using tree structured graphical models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(11):2751–2764, 2013.

[2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR*, abs/1511.00561, 2015.

[3] M. Bai and R. Urtasun. Deep watershed transform for instance segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[4] X. Bai and G. Sapiro. Geodesic matting: A framework for fast interactive image and video segmentation and matting. *Int. J. Comput. Vision*, 82(2):113–132, Apr. 2009.

[5] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei. What's the Point: Semantic Segmentation with Point Supervision. *European Conference on Computer Vision (ECCV)*, October, 2016.

[6] G. J. Brostow, J. Fauqueur, and R. Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 30(2):88–97, 2009.

[7] I. Budvytis. *Novel Probabilistic Graphical Models for Semi-Supervised Video Segmentation*. PhD thesis, University of Cambridge, 2012.

[8] I. Budvytis, V. Badrinarayanan, and R. Cipolla. Semisupervised video segmentation using tree structured graphical models. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2011.

[9] C. Chen, A. Seff, A. Kornhauser, and J. Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *In Proc. of the IEEE International Conference on Computer Vision (ICCV)*, December, 2015.

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

[11] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proc. of the IEEE International Conference on Computer Vision (ICCV)*, December, 2015.

[12] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[13] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[14] L. Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2016.

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

[17] P. Kohli and P. H. Torr. Measuring uncertainty in graph cut solutions - efficiently computing min-marginal energies using dynamic graph cuts. *European Conference on Computer Vision (ECCV)*, May, 2006.

[18] B. Leibe, N. Cornelis, K. Cornelis, and L. V. Gool. Dynamic 3d scene analysis from a moving vehicle. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2007.

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, September 2014.

[20] B. Liu and X. He. Multiclass semantic video segmentation with object-level active inference. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[21] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2015.

[22] Z. Lu, Z. Fu, T. Xiang, P. Han, L. Wang, and X. Gao. Learning from weak and noisy labels for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2016.

[23] W. Maddern, G. Pascoe, C. Linegar, and P. Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, to appear.

[24] J. Marin, D. Vázquez, D. Gerónimo, and A. M. López. Learning appearance in virtual scenarios for pedestrian detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2010.

[25] G. Máttyus, S. Wang, S. Fidler, and R. Urtasun. HD maps: Fine-grained road segmentation by parsing ground and aerial images. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2016.

[26] S. K. Mustikovela, M. Y. Yang, and C. Rother. Can ground truth label propagation from video help semantic segmentation? *CoRR*, abs/1610.00731, 2016.

[27] G. Pandey, J. R. McBride, and R. M. Eustice. Ford campus vision and lidar data set. *International Journal of Robotics Research*, 30(13):1543–1552, 2011.

[28] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. ENet: A deep neural network architecture for real-time semantic segmentation. *CoRR*, abs/1606.02147, 2016.

[29] P. O. Pinheiro, R. Collobert, and P. Dollar. Learning to segment object candidates. In *Advances in Neural Information Processing Systems 28*, pages 1990–1998. 2015.

[30] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. Lopez. The SYNTHIA Dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June, 2016.

[31] T. Scharwächter, M. Enzweiler, U. Franke, and S. Roth. Efficient multi-cue scene segmentation. In *German Conference on Pattern Recognition*, 2013.

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[33] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg. Motion coherent tracking using multi-label mrf optimization. *International Journal of Computer Vision*, 100(2):190–202, 2012.

[34] J. Xiao and L. Quan. Multiple view semantic segmentation for street view images. In *In Proc. of the IEEE International Conference on Computer Vision (ICCV)*, October, 2009.

[35] J. Xie, M. Kiefel, M.-T. Sun, and A. Geiger. Semantic instance annotation of street scenes by 3D to 2D label transfer. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[36] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016.

[37] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.