Sequential Bayesian Model Update under Structured Scene Prior for Semantic Road Scenes Labeling

Evgeny Levinkov Mario Fritz

Max Planck Institute for Informatics, Saarbrücken, Germany
{levinkov, mfritz}@mpi-inf.mpg.de

Abstract

Semantic road labeling is a key component of systems that aim at assisted or even autonomous driving. Considering that such systems continuously operate in the real-world, unforeseen conditions not represented in any conceivable training procedure are likely to occur on a regular basis. In order to equip systems with the ability to cope with such situations, we would like to enable adaptation to such new situations and conditions at runtime.

Existing adaptive methods for image labeling either require labeled data from the new condition or even operate globally on a complete test set. None of this is a desirable mode of operation for a system as described above where new images arrive sequentially and conditions may vary.

We study the effect of changing test conditions on scene labeling methods based on a new diverse street scene dataset. We propose a novel approach that can operate in such conditions and is based on a sequential Bayesian model update in order to robustly integrate the arriving images into the adapting procedure.

1. Introduction

Driving assistance systems have been rapidly evolving lately due to a constantly increasing interest in real-world application as well as studies conducted in the field of computer vision. An important task of such systems is road scene labeling in order to derive the semantic structure of the observed scenes. One of the big challenges is making such systems robust so that they can reliably operate in a wide range of conditions. However, capturing and training every possible condition a car can encounter throughout years of driving seems to be an impossible task.

Recently, there has been an increased interest in approaches of domain adaptation [11, 9] in computer vision that are able to adapt existing classifiers to new domains and conditions. These require supervision from the target domain, that can not be provided by the envisioned systems that continuously operate in the real-world. Existing adaptive methods [1] allow the use of machine generated labels in order to refine the classifier and help it to adapt to changing conditions. However, they perform only global adaptation, for which they require access to the whole test set. Again, this is against the idea of a continuously operating system.

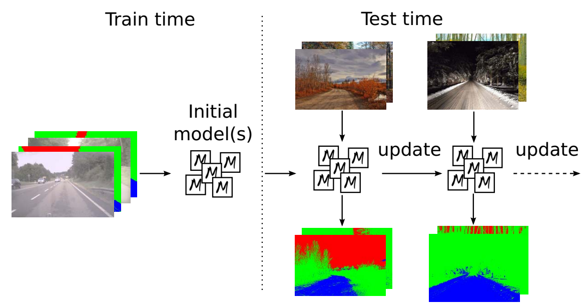

Figure 1: Given an initial model, trained on existing data with groundtruth labels, our algorithm simultaneously labels images as they arrive and updates the model in a robust manner.

In contrast, we aim at an adaptive algorithm that is able to perform adaptation on the fly. Therefore, this paper proposes a sequential bayesian update strategy that pursues multiple model hypothesis for semantic scene labeling. Figure 1 presents an overview of how our algorithm works. In order to circumvent typical problems of online learning by a "self-training" procedure, we perform model updates under the assumption of a stationary label distribution.

The main contributions of this paper are: (1) We present a new dataset of diverse road scenes that allows us to study the effect of drastic changes between training and test feature statistic for semantic scene labeling. (2) We evaluate state-of-the-art scene labeling techniques to provide an initial benchmark on this new challenge set. (3) We propose a novel method for sequential model update in a continuous learning and prediction setting. It is based on a Bayesian update under structured scene prior. The evaluation on our new challenge dataset shows performance improvements of up to 10% compared to non-adaptive baselines.

2. Related Work

Road scenes labeling has been studied for a long time and is predominantly addressed as a labeling problem modeled by Conditional Random Fields (CRFs) [10]. Lots of work has gone into improving the unary potentials [4, 14, 7] as well as the connectivity [8]. Recent advances include: Wojek *et al*. [15] who used appearance-based features and dynamic CRF, Brostow *et al*. [4] who used structure from motion point clouds, Alvarez *et al*. [2] used appearance features based on illuminant invariance, and Structured Class-Labels [7] that model unaries of label patches rather than individual pixels. While exciting progress has been achieved in this domain, these technique do not have the ability to adapt to changing visual conditions.

Domain adaptation techniques [11, 9] can help to solve this problem, but they require at least some sample instances with ground truth labels from the target domain. In contrast, we are aiming for an algorithm which is able to perform adaptation without any possible access to ground truth labels at test-time.

Alvarez *et al*. [1] considered the setting of using machine generated labels at test time for street scene segmentation, but their approach requires access to the whole test set and the quality heavily depends on acceptance threshold parameter. The latter defines which new samples are accepted or rejected based on their likelihood and it cannot be chosen automatically or optimized for. It can be chosen by hand or grid search and therefore suffers from being tuned to a particular dataset. In contrast, our method is targeted at dealing with a stream of incoming images, pursues multiple model hypothesis simultaneously and proposes a more principled way of dealing with the acceptance threshold of new samples for the model update.

3. Sequential Model Update for Semantic Image Labeling

As we aim for vision systems that continuously operate in the real-world, unforeseen conditions not represented in the training set are likely to occur. In order to equip systems with the ability to cope with such situations, we would like to enable adaptation to such new situations and conditions.

There is a large body of work on adaptive learning method which allow the update of models at test time. However, predominantly the availability of labeled data is assumed. If the new data is assumed as unlabeled, we enter the regime of semi-supervised or transductive learning. In such settings, the availability of the full test set is assumed, which is not practical for any continuously operating system.

Therefore we investigate ways how to achieve a sequential model update based on lately arrived, unlabeled data. Such approaches are often associated with the term "self-training". They are troubled with effects of "model drift", which denote effects that occur when erroneous predictions on the test data are used to update the model. Due to these problems and their unprincipled nature, they can diverge and instead of benefiting from the new data, deteriorate in performance. In this section, we first describe how such "self-training" method are typically formulated, then describe how to exploit scene priors and finally propose a new method that improves on "self-training" by a Bayesian model update.

3.1. Naïve Model Update

Typical self-training approaches are based on a two step procedure. First, a lately arrived batch of images is labeled using the current model. Second, after an optional threshold on a confidence rating, these samples are used to update/re-train the model. In more detail, we get an output probability distribution $P(x_{(i,j)})$ from our classifier for each pixel (i, j) and the predicted class-label for it

$$c^* = \operatorname*{argmax}_{c \in \mathcal{Y}} P(x_{(i,j)} = c). \tag{1}$$

Then, as in such setting there is no way of checking whether the given labeling is correct or not, we take features of only those pixels, for which the following holds

$$P(x_{(i,j)} = c^*) > \lambda, \tag{2}$$

where λ is a acceptance threshold parameter. High probability $P(x_{(i,j)} = c^*)$ should indicate high confidence of the classifier in the predicted label. This is a completely heuristic approach, as the classification of the test data is only an approximation to the un-accessible groundtruth. The previously described problems of model drift stem from this approximation.

3.2. Model Update under Scene Prior

It was mentioned in the previous section, that taking new samples with the predicted labels which have high confidence is not necessarily a reliable way of updating the model due to inaccuracies in the intermediate models. While we want to be robust w.r.t. changes in the feature distribution, stationarity of the label distribution is a milder assumptions in many scenarios. We adopt ideas from J. Alvarez *et al*. [1] who employ a pixel-wise, normalized class-histogram on the off-line data as a prior distribution to weight the output probability distribution of the classifier at testing time.

In detail, we compute histogram for each pixel and after per-pixel L_1-normalization we get a prior $P_{pr}^{(i,j)}$ for each pixel $(i,j), i = 1, \ldots, W_{pr}, j = 1, \ldots, H_{pr}$. In our experiments images in the testing dataset all have various dimensions, so we perform nearest-neighbor sampling from the prior distribution $P_{pr}^{(i,j)}$. Then at testing time output probability distribution $P(x_{(i,j)})$ for all pixels $(i,j), i = 1, \ldots, W, j = 1, \ldots, H$ from our classifier for an image with dimensions $W \times H$ is element-wised multiplied with the corresponding prior

$$\tilde{P}(x_{(i,j)}) \propto P(x_{(i,j)}) P_{pr}^{(\lfloor i \frac{H_{pr}}{H} \rfloor, \lfloor j \frac{W_{pr}}{W} \rfloor)}. \tag{3}$$

This is used for accepting or rejecting new training examples on a per-pixel-basis

$$\tilde{P}(x_{(i,j)} = c^*) > \lambda, \tag{4}$$

where c^* is given by (1) and λ is some predefined threshold parameter. We take the corresponding pixel's features together with the predicted label c^* as a new sample if (4) holds.

3.3. Sequential Bayesian Model Update under Structured Scene Prior

We propose a new model to leverage unlabeled data for a sequential model update for scene labeling. Our approach is based on a Bayesian model update. We maintain a population of models (particles) that approximate the distribution over the model-space $p(h_t|L_t)$, instead of relying on a single model, as in the previous formulations. The required integration over the model-space is solved by a Monte-Carlo method – just like in Condensation and Particle Filters that are well known from tracking applications [6, 5]. Consequently, scene labeling at test time will be performed by marginalization over the model distribution

$$p(X|L_t) = \int p(X|h_t) p(h_t|L_t) \, \mathrm{d}h_t, \tag{5}$$

where X is the labeling of a test image for which we want to do prediction.

While the above-mentioned tracking formulations have a measurement step that evaluates image evidence, we measure the compatibility with the scene prior S. This is again based on the assumption of a stationary label distribution $P_{pr}^{(i,j)}$ as for the previous method.

Bayesian Model Update We are interested in modeling an evolving target distribution over models in order to account for the uncertainty in the unobserved scene labels. Therefore, we model the unobserved scene labels l_t of the unlabeled data u_t at time step t as a latent variable. Rather

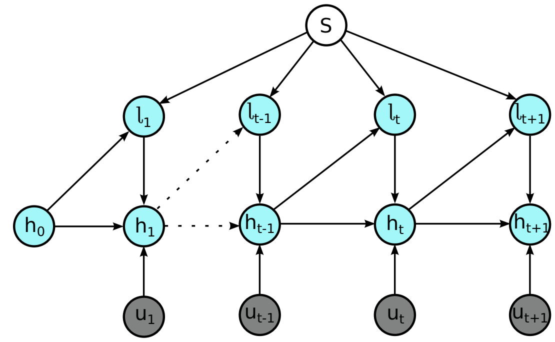

Figure 2: Bayesian network for the proposed model. Adaptation is done at discrete time-steps t. h_t is a set of model hypothesis at time-set t (unobserved), l_t is a set of unknown labelings (unobserved), u_t denotes the set of unlabeled raw images (image features) at time-step t (observed), and S is a statical parameter, an average labeling over the training set.

than sticking to a single model hypothesis, we seek to model a distribution over model hypothesis h_t. Therefore we update a distribution over model hypothesis given labels $p(h_t|L_t)$. Here $L_t = \{l_0, l_1, \ldots, l_{t-1}, l_t\}$.

We describe the incorporation of the unlabeled examples in a Bayesian framework by integrating over all model hypothesis

$$p(h_t|L_{t-1}) = \int p(h_t|h_{t-1}, u_t) p(h_{t-1}|L_{t-1}) \, \mathrm{d}h_{t-1}. \tag{6}$$

In the measurement step, we apply the Bayes' rule in order to get the updated distribution over model hypothesis

$$p(h_t|L_t) = \frac{p(l_t|h_{t-1}, S) p(h_t|L_{t-1})}{p(l_t|L_{t-1})}, \tag{7}$$

with

$$p(l_t|h_{t-1}, S) = p(l_t|h_{t-1}) p(l_t|S), \tag{8}$$

where $p(l_t|h_{t-1})$ is the probability of a certain scene labeling prediction given a model hypothesis h_{t-1} and $p(l_t|S)$ is a scene labeling prior. Figure 2 gives an overview of our model.

Sampling We perform inference with a Monte-Carlo approach [6]. At each time step the model distribution $p(h_t|L_t)$ is represented by a set of particles $s_t^{(N)}$ with weights $\pi_t^{(N)}$. Next, the particles are propagated to the next time step via $p(h_t|h_{t-1}, u_t)$ that takes into account the existing models and the unlabeled data. In traditional tracking application this transition is modeled with a deterministic part and a stochastic component. In our setting, we propose to do model propagation by randomly choosing a subset of images which are provided to a particular classifier to retrain as well as picking a randomized acceptance threshold λ per particle. The benefits are twofold. First, a diverse set of models is generated for the next iteration. Second, parameters like the acceptance thresholds are dealt with within the model and no hard choices have to be made.

In summary, our particle filter over model space works as follows. For each particle i out of N:

1. Pick a particle s_t^i from $s_t^{(N)}$, which represents $p(h_t|L_t)$, according to the weights $\pi_t^{(N)}$
2. Sub-sample set of unlabeled images u_t to $\hat{u}_t$
3. Predict labels $\hat{l}_t = \text{argmax}_l\, p(l|h_t)$ for subset $\hat{u}_t$
4. Accept or reject samples based on some threshold λ
5. Retrain model using $(\hat{u}_t, \hat{l}_t)$ and L_{t-1}

Traditional tracking approaches would now follow up with a measurement in order to update the weights $\pi_t^{(N)}$. Similarly, we update the weight $\pi_t^{(N)}$ of each sample (model hypothesis) according to (7). In this equation $p(h_t|L_{t-1})$ is the distribution represented by our particles after the propagation step from above and $p(l_t|h_{t-1}, S)$ is the product of the likelihood of the labeling times the likelihood of the labeling given the scene labeling prior. We don't compute the denominator - but rather directly normalize the weights of the particles $\pi_t^{(N)}$ to sum to 1.

Implementation details It is important to note, that the update of weights happens at the next time step. We have to do this in order to get a faithful estimation of performance of each of the retrained particles on the same data, which was not in turn used in the retraining of any of the particles. In our implementation we pick the acceptance threshold randomly from the interval 1/3 to 0.9. In all our experiments we use 16 particles – each being a Random Forest classifier. In fact, Figure 3 shows that already a small number of particles allows to get considerable improvements. In each step t we process a batch of 10 images in order to pick a subset as described in the sampling procedure. For the Naïve adaptive approach we set the acceptance thresholding parameter $\lambda = 0.8$. and for the Model Update under Scene Prior we set $\lambda = 0.5$. These are the best performing parameters we found for those two baselines. Code and the new dataset are available on the following website: `https://www.d2.mpi-inf.mpg.de/sequential-bayesian-update`.

4. Diverse Road Scenes Dataset

In order to study the problem of adaptation we need a dataset, which exhibits considerable amount of appearance variation between the training and test set. Typical road scene datasets like [15, 2] (Figure 4, first column) already exhibits some visually difficult situations like changes in

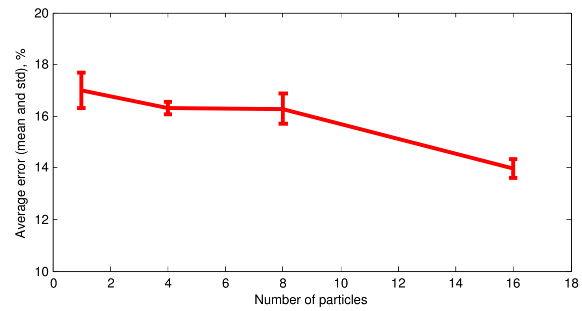

Figure 3: Dependency of average class error on the number of tracking particles.

object appearances due to motion blur effect, deep shadows which appear and disappear suddenly, changes in lightning conditions like over- or under-saturated regions, but the overall statistics stays similar between training and test.

Therefore, we have collected a new dataset which exhibits much richer appearance variation. In order to get the diversity we are aiming for, we turned to Internet resources. We searched over the Internet, and particularly considered Flickr®, looking for images depicting roads mostly in conditions which we called "autumn" and "winter" – weather conditions that are typically avoided in existing datasets. We used the search engine of Flickr® and used tags "dirty roads", "autumn roads", "roads with mud", "winter roads". This resulted in a collection of 220 images, about half of which represent roads in autumn conditions and another half – roads in winter conditions. We performed pixel-wise hand labeling of the gathered images into three classes: road (blue), sky (red), and background (green).

Figure 4 shows examples from each of the "seasons" in our dataset. The dataset expose a much stronger appearance variation than previous datasets. Typical challenges include roads covered in autumn leaves or snow as well as different types of roads such as dirt and gravel roads and even images taken at night, although we leave out such issues like bad lighting, low contrast, or rain.

5. Experimental Results

In our experiments we establish a baseline on our new diverse road scene dataset and compare different non-adaptive techniques for scene segmentation that have different features to increase robustness. Then we evaluate our novel sequential Bayesian update scheme and compare it to different baselines and state-of-the-art in adaptive scene segmentation [1].

Setup and features In our implementations we employed a Random Forest [3] classifier consisting of 10 trees each having depth of at most 15 with 20% bagging of the training

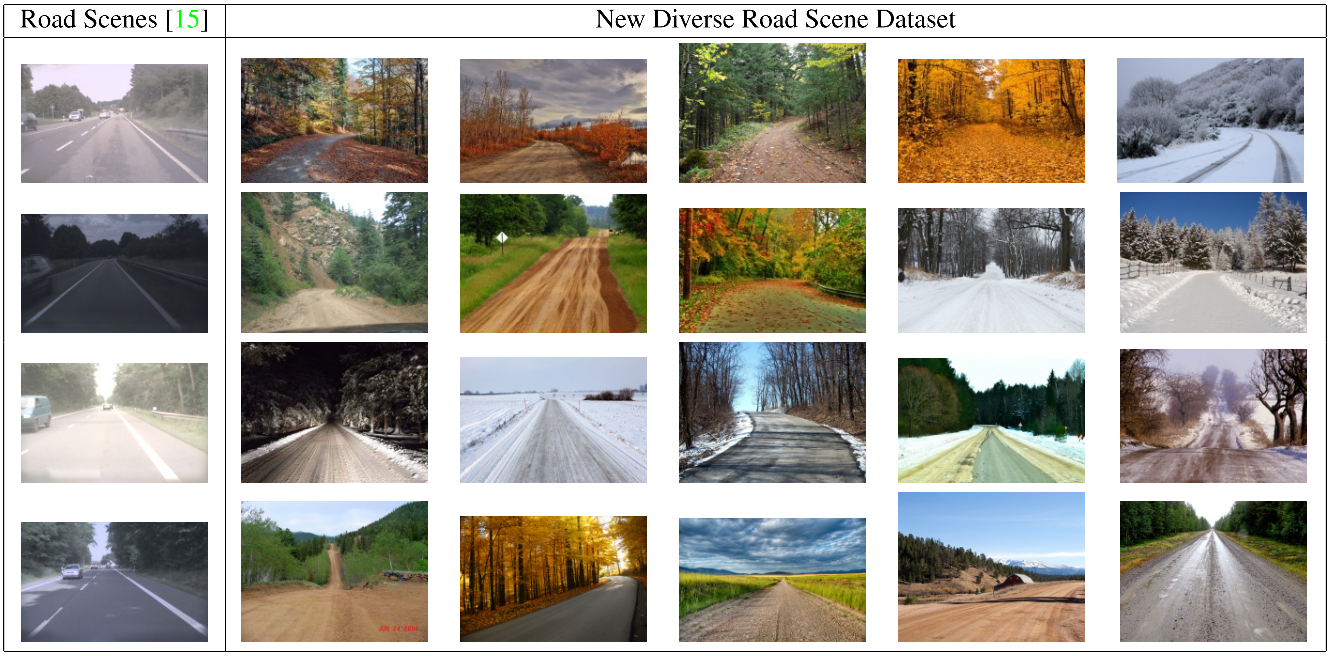

Figure 4: First column shows examples of road scene dataset from [15]. Other columns show examples of the new diverse road scene dataset exhibiting very different appearances and a wider range of conditions.

Method	Error, %			
	Road	Background	Sky	Average
Random Forest	43.8	**10.5**	29.1	27.7
Random Forest+FC-CRF	40.8	11.0	**22.2**	**24.7**
Structured class-labels	**38.6**	11.1	33.1	27.6

Table 2: Comparison of different non-adaptive techniques of enhancing labelings. Bold font highlights the best numbers.

Test set	Fully connected CRF error, %			
	Road	Background	Sky	Average
Old	0.7	2.2	2.7	1.9
New	52.7	6.5	35	31.4

Table 1: Comparison of Krähenbühl *et al*. [8] semantic image labeling algorithm on the old and the new test test.

set, *i.e*. each tree sees at most 20% of the training set to decrease correlation among different trees. This classifier has good accuracy [14], is robust to noisy labels, and its update can be parallelized [13] and be very efficient in terms or running time [12].

Unless stated otherwise, we used the training set from [15] for training (Figure 4, first column) and performed testing or adaptation on the new test set. Groundtruth annotation of the test set is not used in any way, other than for computing error rates. If not noted otherwise, we employ the features from [15]. The resulting feature vector consists of 194 values including: first 16 coefficients of the Walsh-Hadamard transform, grid point's coordinates within the image, raw color features. The features are extracted at multiple scales from all channels of the input image in CIE-Lab color space. As a preprocessing step, a and b channels are normalized by means of gray world assumption to cope with varying color appearance. The L channel is mean-variance normalized to fit a Gaussian distribution with a fixed mean to cope with global lighting variations.

Non-adaptive methods In order to show that non-adaptive methods have a limited capability of generalizing to a different and strongly varying feature distribution as presented in our new dataset, we took one of the state-of-the-art methods for semantic image labeling of Krähenbühl *et al*. [8], and trained it on the training set and tested on both the old and the new test set (Table 1). The old test set has a similar appearance as the training set (Figure 4, first column), so the resulting numbers are very strong. But when we test on the new test set, the method shows strong accuracy degradations caused by the changed feature distribution. Particularly, the road recognition rate gets more than 50 times worse, because background and sky have more or less similar appearance as in the training set, while appearance of the road usually does not resemble the one in the training set.

We have also tried a number of algorithms, which can help non-adaptive methods to perform better in situations when appearance changes considerably by imposing certain additional constrains. Table 2 shows results for: a Random Forest with the above described features, a fully connected CRF (FC-CRF) [8] applied on top of the Random Forest output, and Structured Class-Labels [7]. We use Random Forest as an initial classifier in our implementations of adaptive methods.

Fully connected CRF allows to enhance labeling and make it less noisy by enforcing consistent labels of the neighboring pixels. This method allows to get lower errors for road (3%) and sky (around 7%). We used publicly available implementation of the inference algorithm.

We also tried out our implementation of Structured Class-Labels [7] with the same Random Forest classifier as above. This approach takes also label statistics into account at training time. In the presence of a stationary label statistic (as it is in case for road scenes), it allows for a certain degree of compensation for the changing feature distribution by enforcing an expected label structure for the unseen data. In fact, it shows more than 5% improvement for the road class, although decreasing the detection rate of the sky class, so the average error almost doesn't change.

Global adaptive methods Global adaptive methods consider the whole test set at once and try to adapt to it. The main restriction of such methods is that they require access to the whole test set. In the real world setting, when new images constantly arrive, global algorithms would have to deal with a constantly increasing test set.

Recently, Alvarez *et al*. [1] proposed such an globally adaptive scheme for road scene segmentation. Table 3 (first row) presents resulting numbers for their original method, which the authors kindly agreed to run on our test set. Their method uses different features and a different training set, but their training set also consists of road scenes representing comparable appearance with our training set. The main algorithmic difference is that their method performs adaptation to the whole test set at once, while our Sequential Bayesian Model Update performs sequential updates in a Bayesian formulation allowing real world application, when a fixed "test" set simply does not exist.

Their method doesn't perform well on the new dataset, as we think, because it is a global adaptive method and it considers the whole test set at once and suffers from many false positives. This gives insight to the weaknesses of global adaptive methods in contrast to our sequential method, which updates on small batches and can therefore adapt to an evolving appearance distribution.

Sequential adaptive methods As initial model for this set of experiments we use the Random Forrest model with fully connected CRF from the non-adaptive methods presented above which showed an overall error of 24.7%. This classifier is used as the initial point for an adaptive algorithm and refined during the process of adaptation on the test set.

Table 3 shows resulting numbers for adaptive methods after they have processed the whole test set. The algorithms were run 3 times and the results were averaged over. The Naïve method shows a considerable improvement of over 7% over its initial model. This method doesn't perform any checking of the new samples it accepts, setting λ reasonably high should provide a good indicator that the predicted label is likely to be true.

In order to provide another point of reference to previous work, we compare to a sequential update by using a method in the style of Alvarez *et al*. [1] as described in Section 3.2 based on our features and training set. This results in an improvement over the Naïve (around 1.5%) approach in labeling the road (around 5%) and sky (around 3%). The average error also decreases by more than 4%. This method shows better performance, because it uses prior information to reweight the output of a classifier which allows to decrease the number of false positive samples that are added into the classifier. But still it is worth mentioning that both methods have a considerable variance depending on the initialization and the randomized nature of the algorithm.

We found the algorithm mentioned above to be quite sensitive to the correct choice of the acceptance threshold parameter λ. In contrast, our Bayesian model picks the threshold for each particle at random. Our method allows to get even better labeling for road (around 2%) and sky (around 2%) over the Naïve Model Update under Scene Prior. The overall performance improves to 13.9%, which improves by around 3% over the Naïve Model Update and around 1.5% over the Naïve Model Update under Scene Prior. We would also like to highlight small variance of our algorithm compared to the two previous approaches.

It is interesting to note the inferior performance of our method on the background class, which can be misleading and can be easily explained by the following observation: there are almost no images with the asphalt road in the new dataset (as it is in the set of [15]), so for all other algorithms it is always easier and safer to predict "background" due to the class bias. Basically, for all other algorithms (both non- and adaptive) output labelings often consisted of just background, which is undesirable. While we were interested in treating all classes equally, so low Average error is a good indicator, that our algorithm has improved the labeling qual-

Update type	Method	Error, %			
		Road	Background	Sky	Average
global	Alvarez *et al.* [1]	76.2	12.7	25.5	38.2
sequential	Naïve	26 ±1.4	**15.4±0.4**	9.3±1.4	17±0.7
	Naïve + Scene Prior	21±2.7	18.5±0.6	6.5±0.9	15.5±1.4
	Bayesian Model	**19±0.6**	18.3±0.6	**4.5±0.4**	**13.9±0.3**

Table 3: Comparison of different adaptive approaches after processing the whole test set (mean plus std). Bold font highlights the best numbers.

ity.

Figure 5 shows some example of how labelings for certain images evolve as our Bayesian Model Update method processes one batch of consequent images from the test set after another. The last row represents a situation when an image from the test set looks much like from the train set, therefore our algorithm performs correct labeling in the very beginning and does it throughout the run-time. This shows that our algorithm is stable and does not drift. It is remarkable how our approach can recover from initially poor segmentation results and adapts to the new conditions. We also show the results of the method of Alvarez *et al.* [1], over which we show quantitative as well as qualitative improvements.

6. Conclusion

Today's semantic scene labeling methods show good performance if the training distribution is representative for the test scenario. But when this feature distribution does change, such techniques deteriorate in performance quickly. We collected a challenging dataset of images which has very different appearance statistic compared to the established scene segmentation datasets. A state-of-the-art segmentation algorithm by Krähenbühl *et al.* [8] shows up to 50 times worse recognition rate of scene classes, when tested on the new set over a set of images with appearance similar to the training one.

We showed that even Naïve sequential model update allows to benefit considerably from the new information at test time. Although such method shows high variance of the convergence results which depend on the initialization as well as the choice of the acceptance threshold parameter. In order to cope with this challenge, we propose a Bayesian Model Update that sequentially updates the segmentation model as new data arrives. In contrast to previous algorithm, it gains robustness by maintaining a distribution over models and avoids model drift by exploiting a scene prior. The resulting method shows strong improvements over state-of-the-art non-adaptive baselines as well as recently proposed adaptive approaches.

Acknowledgements

We would like to cordially thank Christian Wojek for advices and providing us with his code for feature extraction and José M. Álvarez for agreeing to run his method [1] on our new dataset.

References

[1] J. M. Álvarez, T. Gevers, Y. LeCun, and A. M. López:. Road scene segmentation from a single image. In *ECCV*, 2012. 1, 2, 4, 6, 7, 8

[2] J. M. Álvarez and A. M. López. Road detection based on illuminant invariance. In *ITS*, 2010. 2, 4

[3] L. Breiman. Random forests. 2001. 4

[4] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, 2008. 2

[5] F. Dellaert, W. Burgard, D. Fox, and S. Thrun. Using the condensation algorithm for robust, vision-based mobile robot localization. In *CVPR*, 1999. 3

[6] M. Isard and A. Blake. Condensation - conditional density propagation for visual tracking. In *IJCV*, 1998. 3

[7] P. Kontschieder, S. R. Bulò, H. Bischof, and M. Pelillo. Structured class-labels in random forests for semantic image labelling. In *ICCV*, 2011. 2, 6

[8] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *NIPS*, 2011. 2, 5, 6, 7

[9] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 2011. 1, 2

[10] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001. 2

[11] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010. 1, 2

[12] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof. On-line random forests. In *ICCV*. 5

[13] T. Sharp. Implementing decision trees and forests on a gpu. In *ECCV*, 2008. 5

[14] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *CVPR*, 2008. 2, 5

[15] C. Wojek and B. Schiele. A dynamic conditional random field model for joint labeling of object and scene classes. In *ECCV*, 2008. 2, 4, 5, 6

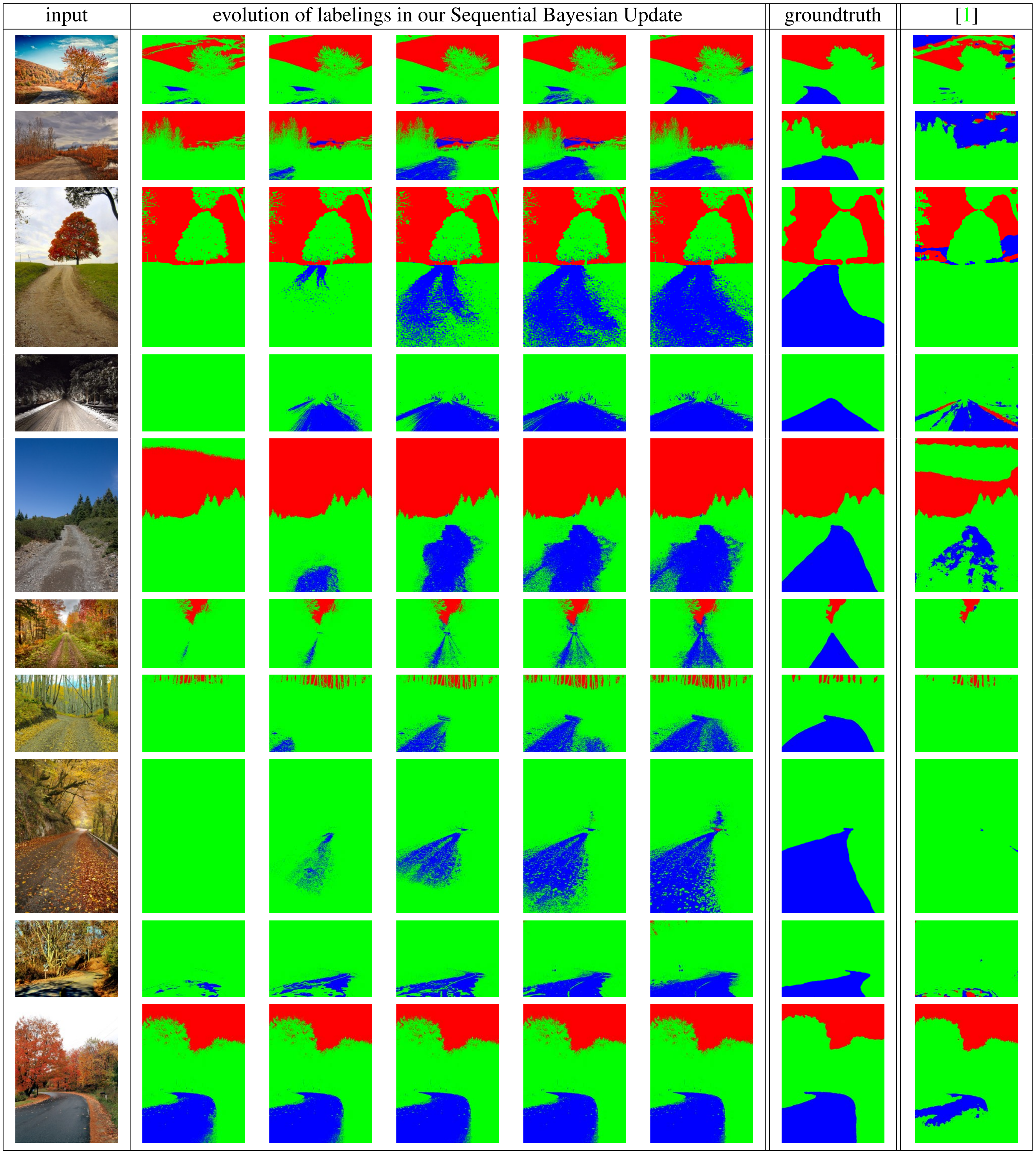

Figure 5: Example results showing the input image, evolution of the labelings through the proposed Sequential Bayesian Update method. The last two columns show the corresponding ground truth annotation and the output of the global adaptive method of Alvarez *et al*. [1]. Green color denotes background, red - sky, and blue - road.