Traffic Scene Segmentation based on Boosting over Multimodal Low, Intermediate and High Order Multi-range Channel Features

Arthur D. Costea and Sergiu Nedevschi

Abstract— **In this paper we introduce a novel multimodal boosting based solution for semantic segmentation of traffic scenarios. Local structure and context are captured from both monocular color and depth modalities in the form of image channels. We define multiple channel types at three different levels: low, intermediate and high order channels. The low order channels are computed using a multimodal multiresolution filtering scheme and capture structure and color information from lower receptive fields. For the intermediate order channels, we employ deep convolutional channels that are able to capture more complex structures, having a larger receptive field. The high order channels are scale invariant channels that consist of spatial, geometric and semantic channels. These channels are enhanced by additional pyramidal context channels, capturing context at multiple levels. The semantic segmentation is achieved by a boosting based classification scheme over superpixels using multi-range channel features and pyramidal context features. A pre-segmentation is used to generate semantic channels as input for more powerful final segmentation. The final segmentation is refined using a superpixel-level dense conditional random field. The proposed solution is evaluated on the Cityscapes segmentation benchmark and achieves competitive results at low computational costs. It is the first boosting based solution that is able to keep up with the performance of deep learning based approaches.**

I. Introduction

Semantic segmentation is the task of pixel-wise image labeling with semantic classes. It provides a high-level representation for an image and can have a significant role in the semantic perception and understanding of the environment by intelligent vehicles. High accuracy and precision are crucial for enabling advanced driver assistance or autonomous driving. This has to be achieved at low computational costs for real-time use.

The state of art on semantic segmentation is rapidly evolving. Due to the recent advances in deep learning, current benchmarks are dominated by approaches that use convolutional neural networks to learn feature representations directly from raw RGB data and also to provide pixel level multiclass predictions. Unfortunately, most approaches have high execution times and hardware requirements, and only few of them are practical for real-time applications [1] [2] [3].

Most intelligent vehicles are equipped with laser scanners or stereo cameras in order to achieve a more powerful perception that also uses depth. The Cityscapes dataset [4] is currently one of the most active segmentation benchmarks and enables the use of depth data by providing stereo image pairs and precomputed disparity images. Currently, from the 40 evaluated solutions, there are only two solutions that use depth data [5] and [6]. Earlier boosting based solution were capable of handling a large variety of features consisting of color, edge, texture or depth features [7] [8], but these are outperformed by RGB input only deep convolutional networks on the CamVid segmentation benchmark [9].

In this work we propose a fast boosting based approach using low, intermediate and high order features from color and depth modalities. The boosting scheme enables easy integration of different feature types that can be handcrafted or learning based. Being the first boosting based solution that is capable of keeping up with the robustness of deep learning based solutions, we hope to encourage research also into this direction. The main contributions of this paper consist of:

- Multimodal and multi order image channels:
  - Low level: multimodal and multiresolution filtered image channels
  - Intermediate level: deep convolutional channels
  - High level: spatial, geometric and semantic channels
- Pyramidal context channels
- Simplified multi-range classification features
- Semantic context from pre-segmentation
- Optimized boosting classification scheme

II. Related Works

Extensive research has been carried out in the field of semantic segmentation, by exploring a variety of features and classification schemes. The availability of segmentation benchmarks such as PASCAL VOC [10], CamVid [9], SYNTHIA [11], Cityscapes [4] have a significant impact on the evolution of segmentation approaches and enable the analysis and better understanding of key factors for achieving more robust segmentations. The Cityscapes dataset is currently the most challenging benchmark considering that it consists of video sequences captured in traffic environments from 50 different cities. 5000 images were fully annotated and 25000 images were only coarsely annotated using 19 semantic classes. The size and quality of training data is a crucial factor for achieving robust results.

One of the first baseline solutions was the Texton-boost approach proposed by Shotton *et al.* in [7]. Segmentation was achieved using a boosting based classification scheme over color and texton features, and Conditional Random Field (CRF) based refinement. Further improvements were

Arthur D. Costea is with the Image Processing and Pattern Recognition Group, Computer Science Department, Technical University of Cluj-Napoca, Romania, e-mail: arthur.costea@cs.utcluj.ro.

Sergiu Nedevschi is the head of the Image Processing and Pattern Recognition Group, Computer Science Department, Technical University of Cluj-Napoca, Romania, e-mail: sergiu.nedevschi@cs.utcluj.ro.

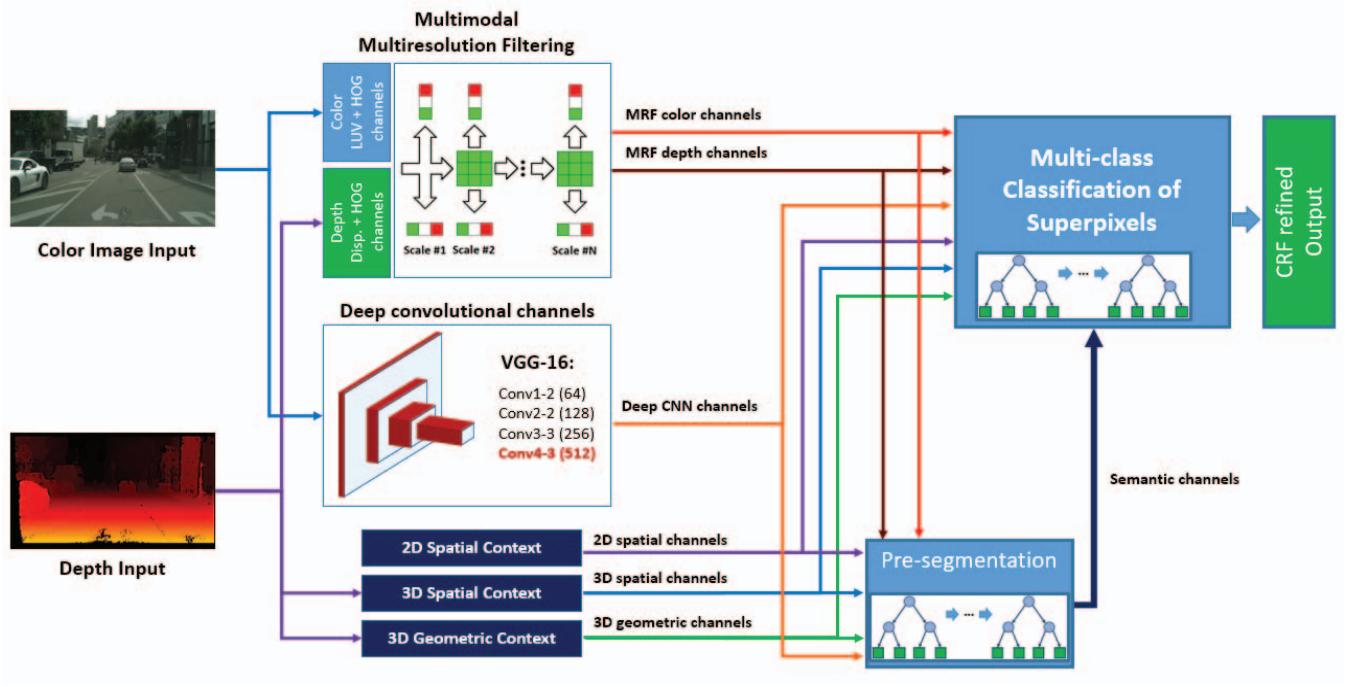

Figure 1. System overview

achieved using more complex CRFs, such as hierarchical CRFs [12] or dense CRFs [13].

Baseline features were improved by Ros *et al.* using a global color transfer strategy [14] to address illumination changes during daytime or between the training and test dataset. The 2D color and texture features have been extended by 3D depth features from structure from motion [15] and dense stereo reconstruction [8] [16]. Cordts *et al.* proposed encode-and-classify trees at pixel level and a multi-cue segmentation tree at the superpixel level in [17], achieving real-time segmentation due to the low computational costs. The employed cues included color, texture, depth, motion and object detection. Fast execution times have been achieved also by using Wordchannel features [18] and multiresolution filtered channel features [19], adopting a boosting classification scheme from sliding window type pedestrian detection.

Due to the recent advances of deep learning networks in image classification the current state of art is dominated by convolutional neural network (CNN) based solutions also in the field of semantic segmentation. The fully connected layer from image classification was replaced by the convolutional layer, resulting in fully convolutional networks (FCNs) [20]. FCNs enabled end-to-end training and became a popular choice for segmentation approaches. Dilated convolutions were introduced in [21] in order to enhance the receptive field of different layers. To address the low resolution prediction of FCNs the DeepLab-CRF [22] approach introduced the atrous convolution and was further extended by CRF-RNN [23] employing recurrent layers. A different strategy was employed in [24] relying on a multi-resolution architecture based on a Laplacian pyramid.

Unfortunately, the drastic improvement of segmentation robustness by deep CNN based solutions comes with a high computational cost. Most approaches have an execution time of multiple seconds for a single image or the time is not reported at all. Solutions such as [1], [2] or [3] are outperformed by the current top performing approaches, but these are the only approaches that managed to reduce computational costs for real-time applications.

There is a high necessity to explore ways to improve results at low computational costs. In this work we introduce the first boosting based solution that is capable of keeping up with deep learning based solutions and show that boosting can be a powerful tool for fusing a large variety of features types.

III. Proposed Solution Overview

Multiple key concepts are introduced in order to achieve robust segmentation at low computational costs. The solution takes advantage of both color and depth perception. Depth is perceived using stereo reconstruction. Multimodal features are captured at three different levels – low, intermediate and high level – and are computed in the form of image channels. The feature channels are extended by pyramidal context channels, capturing context at multiple scales. To achieve semantic segmentation superpixels are classified based on multi-range channel features and pyramidal context features using boosted decision trees.

A smaller scale of this classification scheme, trained on a different training subset, is used to achieve a presegmentation of the input. The presegmentation is decomposed into higher order semantic channels and is used as input for the final segmentation.

The final segmentation is refined using a dense Conditional Random Field (CRF) defined at superpixel-level. An overview of the proposed solution is illustrated in Fig. 1.

Figure 2. Color and depth (disparity) gradient.

IV. Multi-level Features

In the following, we define the employed multimodal and multi-level image features. The computed features are stored in the form of image channels and the classification features will be sampled from these channels. The boosting classification scheme has the task of fusing these different order features for achieving a robust segmentation. For each feature type we also focus on reducing computational costs.

A. Low level features

As low level features we consider features that capture basic color, texture or edge information. These pixel level features consider only a small neighborhood of surrounding pixels resulting in a small receptive field. In order to capture edges at multiple scales and multiple orientations, we employ the fast multiresolution filtering scheme proposed in our previous work [25]. In this work we extend the filtering scheme also with depth data. In previous works that employed depth data [15] [8] [16], depth was mostly used as a source for 3D spatial context for semantic pixel classification. Dense depth data can be a significant source also for object boundaries and structure elements, considering that it is invariant to texture, as seen in Fig. 2. The multimodal multiresolution filtered channels will represent a pool of basic building blocks for learning more complex structures by the boosted decision trees.

In the case of color modality, we compute 10 image channels consisting of LUV color channels, gradient magnitude and the magnitude at 6 orientations (HOG). These channels are filtered iteratively using a 3 × 3 box filter as a low-pass filter to achieve features at multiple scales. The resulting channels are high-pass filtered using simple vertical and horizontal differences in order to capture high frequency edge elements at multiple scales and orientations.

In the case of depth modality we use the dense disparity image as the source for gradient magnitude and orientation channels, resulting in 8 channels. These channels are also filtered using the low-pass and high-pass filtering scheme.

In order to have a lower number of filtered channels, we use only 4 filtering scales, resulting in one low-pass and two high pass filtered channels at each scale of the 10 color and 8 depth input channels (a total of 216 filtered channels). Due to the computational simplicity of the filtering scheme, all channels can be computed in less than 2 ms for a 1024 × 512 pixel image using a GPU implementation.

B. Intermediate level features

As intermediate features we consider larger structures or components that build up the objects that have to be recognized. Deep CNNs are particularly good at learning complex structures in a hierarchical manner. Zeiler and Fergus provide an analysis in [26] and also illustrate the features that are learnt by the different layers of a deep CNN. The first layers learn basic edge and color features while the upper layers learn more and more complex features having also larger receptive fields. Yang *et. al* proposed in [27] the use of boosted decision trees over convolutional channel features for pedestrian sliding window classification. Extensive experiments have been done in evaluating different deep net architectures and the layers of each net. The best results were achieved using traditional LUV + HOG image channels together with the last 512 convolutional filters of the 4th layer of the ImageNet pretrained VGG16 net [28]. Interestingly, the use of higher layers reduced detection robustness, concluding that the boosting based classification and feature pooling scheme is more efficient at combining intermediate level features. Due to the necessity of recomputing image features for mutliscale detection, the execution time of the approach was of 13 seconds for an image.

In the case of semantic segmentation, deep CNNs are usually applied at a single scale. The computation of the 512 convolutional filtering results of the 4th convolutional layer group (conv4_3) of VGG16 is achieved in around 50 ms using a GPU for an image of 1024 x 512 pixels. In this work we propose the extension of multimodal multiresolution

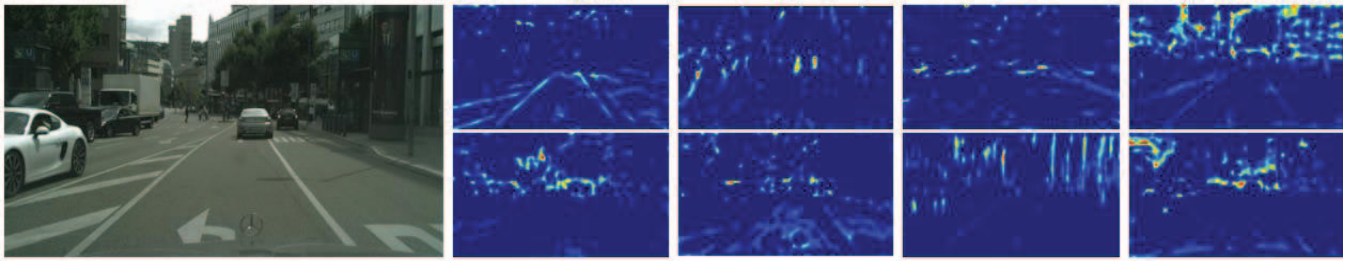

Figure 3.VGG16 Deep CNN convolutional kernel response examples from the 512 conv4_3 filters

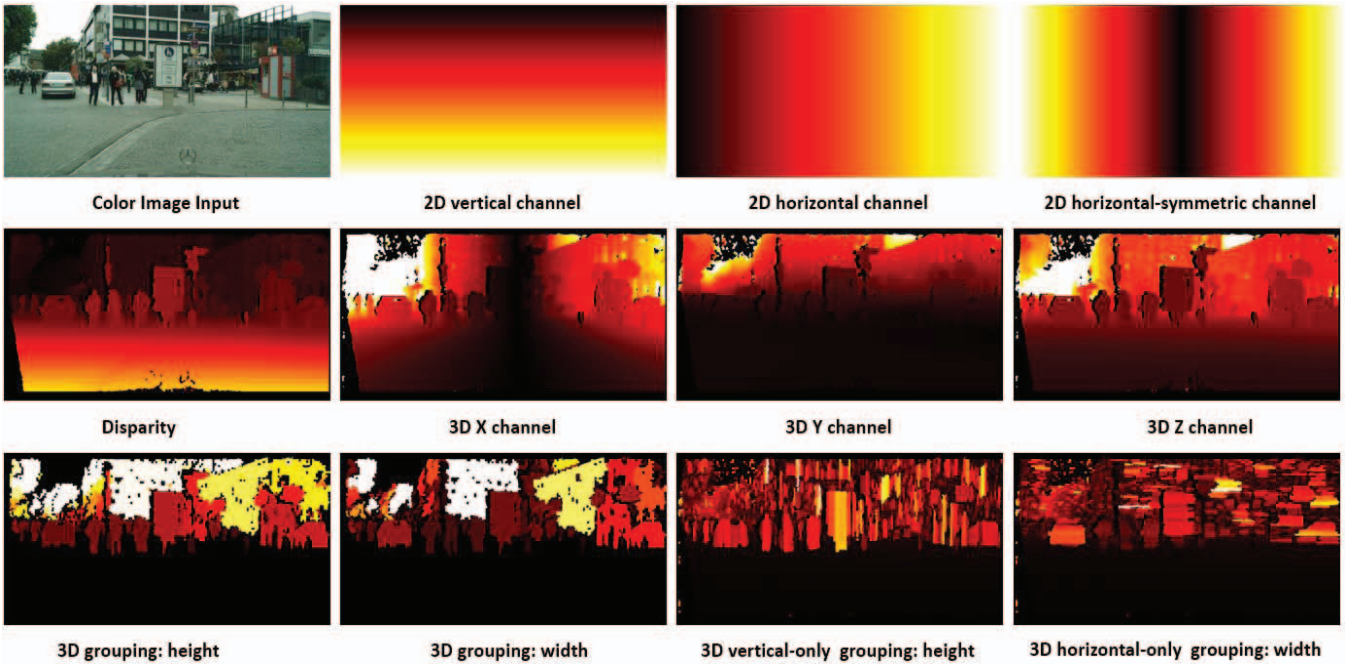

Figure 4. 2D spatial, 3D spatial and 3D geometric channels.

filtered channels with deep convolutional filtered channels that were pretrained on the largescale ImageNet dataset [29]. The downsampling ratio of the additional channels is of 8, resulting in channels of 128 × 64 pixels.

C. *High level features*

The high level features enable reasoning over the 2D spatial, 3D spatial, 3D geometry and semantic context during the semantic classification of pixels. These features are encoded as normalized values in image channels and the boosted decision trees can learn different constraints over these values for different classes.

In our previous work [19], we employed 2D spatial context channels to permit the learning of the 2D context for different classes. These channels consist of vertical, horizontal and symmetric-horizontal channels and represent normalized 2D image coordinates at each pixel position. For example, the vertical channel represents the normalized 2D vertical coordinate. As an example, the boosting classifier can learn that the road pixels tend to have a high value in this channel, being closer to the bottom of the image. Similarly, the 3D context can be learnt using 3D spatial channels representing the normalized X, Y and Z 3D coordinates. These channels are illustrated in Fig. 4.

In this work we propose the use of geometric channels. We apply a grouping over superpixels to identify cohesive structures and determine the height, width and size of each such structure. We integrate this information in the form of normalized values in image channels in order to allow boosting classifiers to learn the geometric constraints of different classes. For example, it is difficult to differentiate a car from a truck at pixel level, but if the 3D height channel tells that the pixel is part of an object that has a height of over 2 meters, then the probability of a truck will be much higher.

To identify cohesive structures, first we segment the image into 16000 superpixels. This allows us to capture narrow structures, such as light poles or traffic lights. Using an approximation of the SLIC [30] segmentation approach and GPU implementation, the segmentation is achieved in around 2.5 ms. We use superpixel-level region growing to identify cohesive structures. Two superpixels are merged if the absolute difference in 3D (on the longitudinal axis) is less than 2.5% of the distance to camera. A relative threshold is used due to the decrease of depth precision with distance. We ignore superpixels that have a height above ground of less than 0.5 meters and ignore clusters with a size of only 1 or 2 superpixels. Finally, we determine the height, width and area for each group. These properties are encoded as normalized values in the form of image channels, called geometric channels. This way we enable boosting classifiers to take into consideration the 3D size of the objects for pixel level classification.

When applying region growing, an object can be easily merged with background if there is a single superpixel with erroneous 3D data next to the whole object boundary (which can be a large region). To enable at least the partial extraction of objects for difficult cases, we also generate groupings that are limited to only vertical or horizontal merging. In the case of vertical merging the representation is similar to the Stixel representation used in [31]. In the case of vertical merging we generate a geometric channel representing the 3D height of each group and in the case of horizontal merging the 3D width. We generate additional channels where the grouping size is determined by the number of pixels, instead of 3D size, to enable the perception of regions where depth reconstruction is missing or is not reliable. Example illustrations of geometric channels are provided in Fig. 4. The high level feature channels are computed in less than 1 ms.

V. PYRAMIDAL CONTEXT CHANNELS

Context has been proved to be an important factor for achieving robust segmentation in several works [32] [33] [34]. The context can be determined at multiple levels of larger and larger pixel neighborhoods. In the case of global context, the whole image is considered. As shown in [34], simple average pooling over larger regions from deep convolutional filter outputs can provide a significant contextual boost.

In this work we propose the partitioning of the feature channels, described in the previous section, into multi-level pyramid cells as illustrated in Fig. 6. Similar partitions, called spatial pyramids [35], are used frequently in the context of bag of features based image classification. The largest cell includes the whole image and represents the first layer. The second layer partitions the first cell into two square cells. From this layer on, the next layer is achieved by partitioning into cells that are two times smaller. We compute the average value for each cell in each image channel, to generate classification features, resulting in the pyramidal context channels. For computational efficiency, we use 8 levels for low level and high level feature types, and 5 levels for the intermediate, convolutional channels (due to the smaller resolution). In the case of smoothed filtered channels and 2D spatial channels the average values do not provide additional information and the context pyramid is not computed.

VI. CLASSIFICATION

In the following we define a classification scheme that can be used for the semantic classification of an individual pixel. Multiclass semantic segmentation is achieved by training individual binary classifiers for each class.

A. Multi-range classification features

We use the feature channels described in Section IV to generate classification features. A pixel is classified based on the channel values of the pixel neighborhood. The relevant region can be large due to the variety of semantic classes and object sizes, resulting in a high number of potential classification features. In our previous work [19] we proposed the use of multi-range features. The classification features were sampled in a grid-wise manner from feature channels using grids of 13 x 13 pixel locations. Four grids were considered using different pixel step rates, covering four different pixel ranges. The denser near range was responsible for capturing local structure while the sparser far ranges were responsible for capturing the context.

In this work we opt for a simpler grid of classification features. Instead of sampling the features in a grid-wise manner from the rectangular regions of interest, we sample only horizontally and vertically, as illustrated in Fig. 6. This way, the number of potential features is reduced by an order of magnitude and there is only a small decrease in classification performance, as shown in the experimental results section. The main advantage is that we are able to use more training samples (limited by memory), which has a larger impact on classification performance. In our experiments we sample the classification features for training using a one dimensional vertical and a one dimensional horizontal grid of 7 samples with step rates of 2, 8 and 32

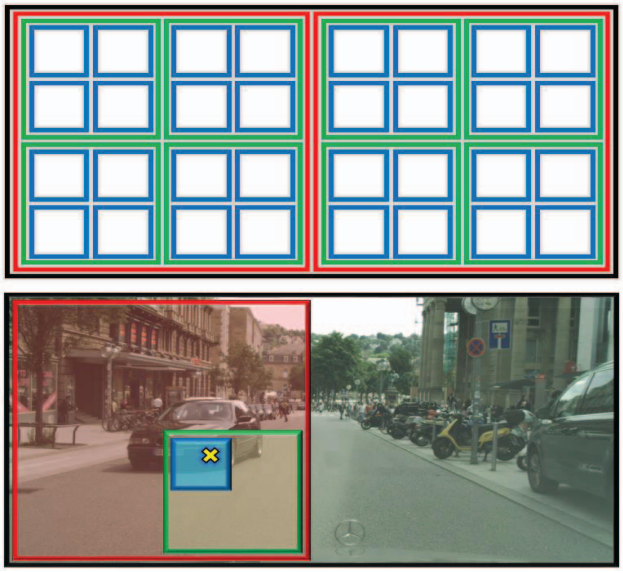

Figure 5. First four levels of the context pyramid and the cells associated to a pixel location. The average value is computed as feature for each cell.

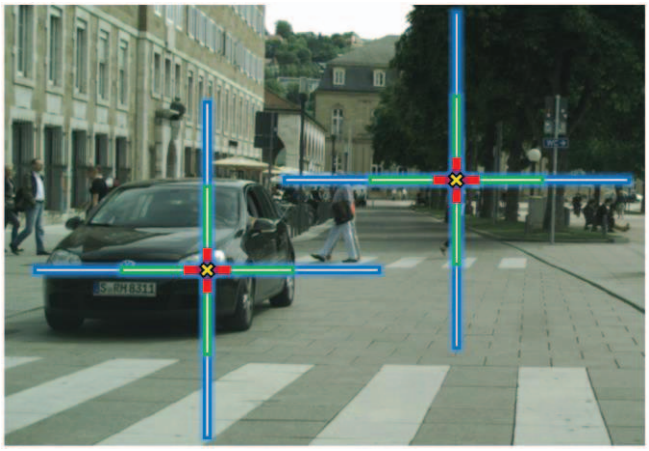

Figure 6. Multi-range classification feature sampling using three ranges

pixels. The 3 grids have different ranges: short range - 14 pixels; middle range - 52 pixels; long range - 208 pixels. In case of samples that would be located outside the image bounds (for pixels close to the margins) we take the closest computed location. To further reduce the number of features, we ignore the long range features in the case of low level feature channels (due to the focus on local structure) and the short range features in the case of intermediate, convolutional channels (due to the downsampled channels' size).

B. Contextual classification features

Another type of classification features is extracted from the pyramidal context channels. The pixel that has to be classified is associated to each pyramid cell it belongs to. Each image feature channel and each pyramid level results in a single classification feature representing the average channel value of the pyramid cell. The highest level represents the global average of the whole channel, while the lowest level represents the average of the smallest cell. Five classification features are obtained in the case of intermediate deep convolutional channels and eight in the case of other channels.

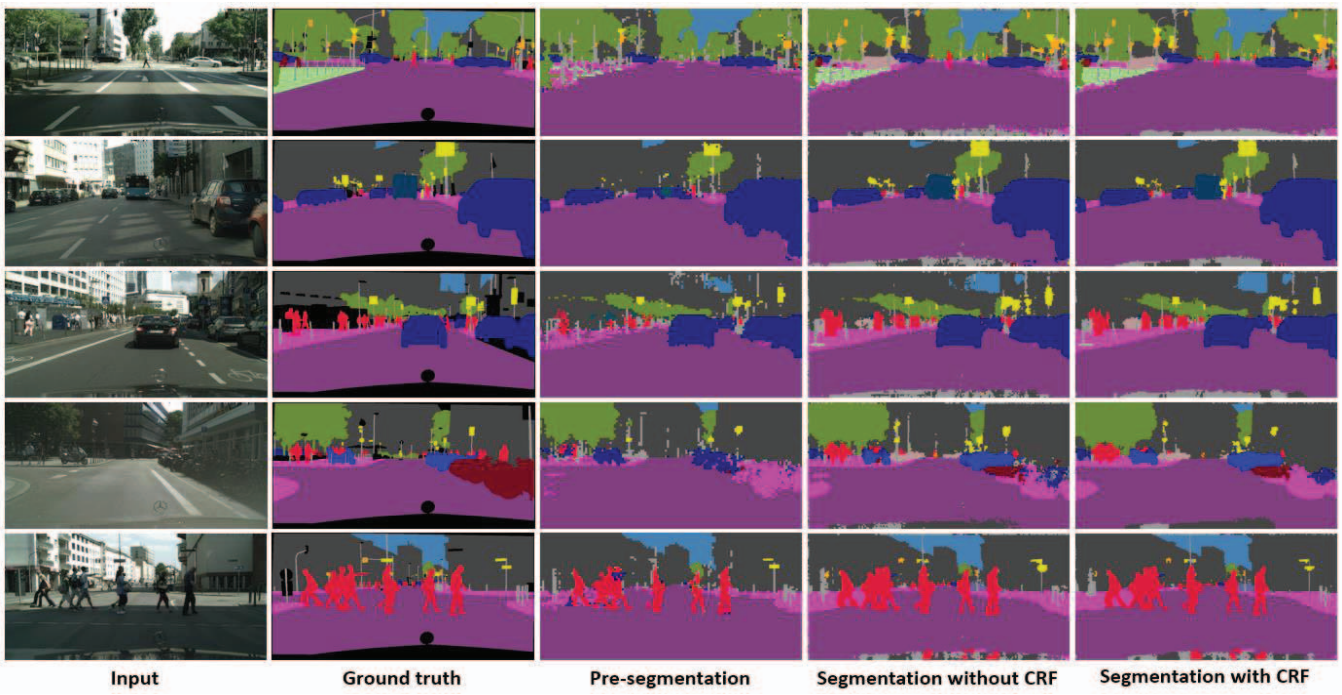

Figure 7. Semantic segmentation results using pre-segmentation and CRF refinement.

C. *Classifier training*

We train a binary boosting classifier for each individual semantic class. For each class we use 400000 training samples, consisting of 100000 positive samples and 300000 negative samples. The number of samples is limited only by the available memory. The training instances are sampled randomly from the whole training database based on the ground truth segmentation. In the case of negative samples, we use an equal number of samples for each negative class. This way, we avoid having too few negative samples for semantic classes that are underrepresented in the training set.

For training robust boosting classifiers, we follow the insights from boosting based pedestrian sliding window classification [25]. A boosting classifier is trained consisting of 2048 boosted 7 level decision trees. The number of levels is increased due to the higher number of training samples. Only a random subset of 1% is considered from all classification features when training the decision stump nodes of the decision trees resulting in a significant acceleration of the training process. This also reduces overfitting and has a minimal effect on classification performance.

D. *Semantic segmentation*

Training multiple binary boosting classifiers enables multi-class semantic classification of individual pixels. To reduce the number of necessary classifications for a full image segmentation, we classify only the center pixel of each superpixel. The class with the highest prediction score.is selected and the prediction is retained at superpixel level.

We do a simple classifier calibration in order to decrease the false positive and false negative rate of each class. The addition of an offset to the prediction score of a class increases false positive rate and decreases false negative rate, while the subtraction has the opposite effect. We find the offset for each class using binary search. The offsets assure an equal false negative and false positive rate on a validation set. In this case the class accuracy and precision is equal.

The boosting classifiers of the semantic classes are evaluated in a sequential order. We use a validation set to learn minimum and maximum prediction score thresholds. The evaluation of the boosted decision trees is halted if the prediction score drops below the minimum threshold or increases above the maximum, resulting in faster prediction without affecting the classification accuracy.

Several approaches apply CRF based refinement [13] [5] [22] in order to achieve a more robust and smoother segmentation. Similarly, to our previous work [19], we use a dense CRF defined at superpixel level to infer the final segmentation. Due to the employment at superpixel level, dense CRF inference improves the segmentation at low computational costs.

VII. SEGMENTATION FEEDBACK

In this work, we propose the computation of an initial segmentation and use this to refine the final segmentation. For training the boosting based classifiers for the pre-segmentation module, we employ a smaller individual training dataset. In the case of Cityscapes dataset, we use the training sequences from the first three cities (Aachen, Bochum and Bremen), consisting of 586 annotated images. These images are excluded from the training set of the final classifier, in order to avoid overfitting to training data. We follow the training protocol described in the previous section. Due to the smaller training size, we use four times fewer training samples and train 512 boosted 5-level decision trees for each semantic class. To reduce computational costs, we classify only each 16^{th} pixel (2048 classifications) instead of each superpixel. As will be shown in the experimental results, the pre-segmentation module is able to provide

relatively good performances, considering the size of the training set and the subsampled prediction.

We use the pre-segmentation module to generate the prediction of each individual semantic class and store it in the form of semantic channels. These semantic channels are integrated into the final segmentation module in the form of additional high level feature channels. The final boosting classifiers have the ability to take into consideration the pre-classification and the semantic context of each pixel. Pre-segmentation and final CRF-refined segmentation is illustrated in Fig. 7.

VIII. Experiments

In order to evaluate the performance of the proposed solution we consider the Cityscapes [4] traffic scene segmentation dataset, being currently one of the most challenging and most active benchmarks. The segmentation performance can be evaluated on 19 semantic classes or 7 semantic categories. The standard Jaccard Index, also known as intersection-over-union (IoU), is used as main evaluation metric. The IoU is computed for each individual class or category and the mean value is reported.

First, we evaluate iteratively the contribution in performance of the individual feature types. We train only the 7 category classifiers and use only each 5th training image, due to the necessity of retraining the classifiers for each class and each configuration. We evaluate the mean IoU and mean accuracy at category level, and the global pixel classification accuracy. The evaluation is done on the standard validation set. In Table I we show the segmentation performance after adding iteratively the multi-level multi-range features and pyramidal context features. Then, we show the performance of the pre-segmentation module and the final segmentation using also semantic channels. It can be seen that each feature type provides an important boot in performance.

In Table II we illustrate the advantage of using the proposed vertical and horizontal-only multi-range features. First, we show the performance for using 2D grid multi-range features and using 25000 positive and 75000 negative samples for training each semantic classifier. Retraining the classifiers using the same training samples, but employing 1D grid multi-range features, provides only slightly lower results. Due to the lower number of classification features, we can use more training samples and achieve better results with the same training memory requirement.

Finally, we train a classifier for each of the 19 classes using all proposed features. We also use the extended training set (with coarse annotation), but we sample only 300 additional images for each of the 6 classes with the lowest number of training samples. In Table III, we provide a comparison with the current state of art based on mean class-IoU, category-IoU and execution time (only published works with available execution time are shown). The proposed solution is the first boosting based approach that provides competitive results. In Table IV we provide the IoU for each class. Additional details are available on the Cityscapes benchmark webpage (MultiBoost approach).

The training of a boosting classifier for a single class takes around 30 minutes with an i7-5960x CPU. The execution time for semantic segmentation with 19 classes is around 240 ms using an NVidia GTX 980Ti GPU. Feature computation takes around 60 ms, pre-classification 30 ms, final segmentation 120 and superpixel-level CRF 30 ms. The solution can be easily adapted for real-time applications.

TABLE I. Segmentation performance using different features

	Mean IoU	Mean Acc.	Global Acc.
Low level: multimodal MRFC	71.5	81.8	90.8
+ Intermediate level: CNN channels	73.2	82.7	91.2
+ High level: 2D + 3D channels	75.1	84.6	92.1
+ Pyramidal Context	76.8	86.7	92.5
Pre-segmentation	61.2	76.8	85.4
Final segmentation	78.8	87.3	93.6
Final segmentation + CRF	79.9	88.6	94.3

TABLE II. Multi-range feature grid type evaluation

	Mean IoU	Mean Acc.	Global Acc.
2D grid multi-range features & 100000 training samples	76.1	86.3	92.3
1D grid multi-range features & 100000 training samples	75.9	86.1	92.2
1D grid multi-range features & 400000 training samples	78.8	87.3	93.6

TABLE III. Cityscapes test set results: comparison

	Class Mean IoU	Category Mean IoU	Runtime [s]
Dilation10	67.1	86.5	4.0
Adelaide	66.4	82.8	35.0
FCN8s	65.3	85.7	0.5
DeepLab LargeFOV StrongWeak	64.8	81.3	4.0
DeepLab LargeFOV Strong	63.1	81.2	4.0
CRFasRNN	62.5	82.7	0.7
SQ	59.8	84.3	0.06
ENet	58.3	80.4	0.013
Segnet basic	57.0	79.1	0.06
Segnet extended	56.1	79.8	0.06
MultiBoost - ours	59.2	81.8	0.25

IX. Conclusion

In this paper we introduced a novel boosting based solution for semantic segmentation of traffic scenarios. It relies on multiple key features that enable robust segmentation at low computational costs. Multisensorial perception is exploited by computing low-level and high level features from color and depth modalities. These features are further enhanced by intermediate level deep convolutional channel features and high level semantic channels obtained from a pre-segmentation. The semantic classification of superpixels is achieved by boosting classifiers employing multi-range channel features and pyramidal context features.

TABLE IV. CITYSCAPES TEST SET RESULTS: PER-CLASS AND MEAN IOU (%)

Road	Sidewalk	Building	Wall	Fence	Pole	Traffic light	Traffic sign	Vegetation	Terrain	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle	Mean
95.8	69.4	87.2	34.4	32.7	40.4	54.8	58.6	89.2	65.2	90.2	68.4	42.5	89.0	22.5	51.8	40.8	36.5	55.6	59.2

The proposed solution is evaluated on the Cityscapes segmentation benchmark and achieves competitive results in comparison with deep learning based solutions. We show that although deep learning approaches may dominate the field, boosting based channel feature classification can be a powerful tool in the context of real-time applications, enabling novel improvement possibilities.

ACKNOWLEDGMENT

This work was supported by the EU H2020 project UP-Drive under grant nr. 688652.

REFERENCES

[1] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. “ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation”. In *arXiv*, 2016.

[2] V. Badrinaralyanan, A. Kendall and R. Cipolla. “SegNet: a deep convolutional encloder-decoder architecture for scene segmentation”. In *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2017.

[3] M. Treml, J. Arjona-Medina, T. Unterthiner, R. Durgesh, F. Friedmann, P. Schuberth, A. Mayr, M. Heusel, M. Hofmarcher, M. Widrich, B. Nessler, and S. Hochreiter. “Speeding up Semantic Segmentation for Autonomous Driving”. In *Conference on Neural Information Processing Systems (NIPS) Workshop*, 2016.

[4] 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 *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] I. Kreso, D. Causevic, J. Krapac, and S. Segvic. “Convolutional Scale Invariance for Semantic Segmentation”. In *German Conference on Pattern Recognition (GCPR)*, 2016.

[6] J. Uhrig, M. Cordts, U. Franke, and T. Brox. “Pixel-level Encoding and Depth Layering for Instance-level Semantic Labeling”. In *German Conference on Pattern Recognition (GCPR)*, 2016.

[7] J. Shotton, J.Winn, C. Rother, and A. Criminisi. “Textonboost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context”. In *International Journal of Computer Vision (IJCV)*, 2009.

[8] C. Zhang, L. Wang, and R. Yang. “Semantic segmentation of urban scenes using dense depth maps”. In *European Conference on Computer Vision (ECCV)*, 2010.

[9] G. J. Brostow, J. Fauqueur, and R. Cipolla. “Semantic object classes in video: A high-definition ground truth database”. In *Pattern Recognition Letters (PRL)*, 2009.

[10] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. “The pascal visual object classes (VOC) challenge”. In *International Journal of Computer Vision (IJCV)*, 2010.

[11] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and Antonio Lopez. “The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes”. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[12] P. Kohli, and P. H. Torr. “Robust higher order potentials for enforcing label consistency”. In *International Journal of Computer Vision (IJCV)*, 2009.

[13] P. Krahenbuhl and V. Koltun. “Efficient inference in fully connected CRFs with gaussian edge potentials”. In *Conference on Neural Information Processing Systems (NIPS)*, 2011.

[14] G. Ros, and J. M. Alvarez. “Unsupervised image transformation for outdoor semantic labelling”. In *Intelligent Vehicles Symposium (IV)*, 2015.

[15] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. “Segmentation and recognition using structure from motion point clouds”. In *European Conference on Computer Vision (ECCV)*, 2008.

[16] T. Scharwachter, and U. Franke. “Low-level fusion of color, texture and depth for robust road scene understanding”. In *Intelligent Vehicles Symposium (IV)*, 2015.

[17] M. Cordts, T. Rehfeld, M. Enzweiler, U. Franke, and S. Roth. “Tree-Structured Models for Efficient Multi-Cue Scene Labeling”. In *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2016.

[18] A. D. Costea, and S. Nedevschi. “Multi-class segmentation for traffic scenarios at over 50 fps”. In *IV*, 2014.

[19] A. D. Costea, and S. Nedevschi. “Fast Traffic Scene Segmentation using Multi-range Features from Multi-resolution Filtered and Spatial Context Channels”. In *Intelligent Vehicles Symposium (IV)*, 2016.

[20] J. Long, E. Shelhamer, and T. Darrell. “Fully Convolutional Networks for Semantic Segmentation”. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[21] F. Yu, and V. Koltun. “Multi-Scale Context Aggregation by Dilated Convolutions”. In *ICLR*, 2016.

[22] G. Papandreou, L. C. Chen, K. Murphy, and A. L. Yuille. “Weakly- and Semi-Supervised Learning of a DCNN for Semantic Image Segmentation”. In *International Conference on Computer Vision (ICCV)*, 2015.

[23] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr. “Conditional Random Fields as Recurrent Neural Networks”. In *International Conference on Computer Vision (ICCV)*, 2015.

[24] G. Ghiasi, and C. C. Fowlkes. “Laplacian Pyramid Reconstruction and Refinement for Semantic Segmentation”. In *European Conference on Computer Vision (ECCV)*, 2016.

[25] A. D. Costea, and S. Nedevschi. “Semantic channels for fast pedestrian detection”. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[26] M. D. Zeiler, and R. Fergus. “Visualizing and Understanding Convolutional Networks”. In *European Conference on Computer Vision (ECCV)*, 2014.

[27] B. Yang, J. Yan, Z. Lei, and S. Z. Li. “Convolutional Channel Features”. In *International Conference on Computer Vision (ICCV)*, 2015.

[28] K. Simonyan, and A. Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In *International Conference on Learning Representations (ICLR)*, 2015.

[29] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. “ImageNet Large Scale Visual Recognition Challenge”. In *International Journal of Computer Vision (IJCV)*, 2015.

[30] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. “SLIC Superpixels Compared to State-of-the-art Superpixel Methods”. In *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2012.

[31] L. Schneider, M. Cordts, T. Rehfeld, D. Pfeiffer, M. Enzweiler, U. Franke, M. Pollefeys, and S. Roth. “Semantic Stixels: Depth is Not Enough”. In *Intelligent Vehicles Symposium (IV)*, 2016.

[32] A. Lucchi, Y. Li, X. B. Bosch, K. Smith, and P. Fua. “Are spatial and global constraints really necessary for segmentation”. In *International Conference on Computer Vision (ICCV)*, 2011.

[33] W. Liu, A. Rabinovich, and A. C. Berg. “Parsenet: Looking wider to see better”. In *CoRR*, 2015.

[34] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. “Pyramid Scene Parsing Network”. In *arXiv*, 2016.

[35] S. Lazebnik, C. Schmid, and J. Ponce. “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories”. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.