Pulling Things out of Perspective

Ľubor Ladický
ETH Zürich, Switzerland
lubor.ladicky@inf.ethz.ch

Jianbo Shi
University of Pennsylvania, USA
jshi@seas.upenn.edu

Marc Pollefeys
ETH Zürich, Switzerland
marc.pollefeys@inf.ethz.ch

Abstract

The limitations of current state-of-the-art methods for single-view depth estimation and semantic segmentations are closely tied to the property of perspective geometry, that the perceived size of the objects scales inversely with the distance.

In this paper, we show that we can use this property to reduce the learning of a pixel-wise depth classifier to a much simpler classifier predicting only the likelihood of a pixel being at an arbitrarily fixed canonical depth. The likelihoods for any other depths can be obtained by applying the same classifier after appropriate image manipulations. Such transformation of the problem to the canonical depth removes the training data bias towards certain depths and the effect of perspective. The approach can be straight-forwardly generalized to multiple semantic classes, improving both depth estimation and semantic segmentation performance by directly targeting the weaknesses of independent approaches. Conditioning the semantic label on the depth provides a way to align the data to their physical scale, allowing to learn a more discriminative classifier. Conditioning depth on the semantic class helps the classifier to distinguish between ambiguities of the otherwise ill-posed problem.

We tested our algorithm on the KITTI road scene dataset and NYU2 indoor dataset and obtained obtained results that significantly outperform current state-of-the-art in both single-view depth and semantic segmentation domain.

1. Introduction

Depth estimation from a single RGB image has not been an exhaustively studied problem in computer vision, mainly due to its difficulty, lack of data and apparent ill-posedness of the problem in general. However, humans can still perform this task with ease, suggesting, that the pixel-wise depth is encoded in the observed features and can be learnt directly from the data [20, 1, 10, 16]. Such approaches typically predict the depth, the orientation, or fit the plane for the super-pixels using standard object recognition pipeline, consisting of the calculation of dense or sparse features, building rich feature representations, such as bag-of-words, and application of a trained classifier or regressor on them. The responses of a classifier or a regressor are combined in a probabilistic framework, and under very strong geometric priors the most probable scene layout is estimated. This process is completely data-driven and does not exploit known properties of the perspective geometry, most importantly, that the perceived size of the objects scales with an inverse distance (depth) from the centre of projection. This leads to severe biases towards the distributions of depths in the training set; it is impossible to estimate the depth of an object if a similar object has not been seen at the same depth during the training stage. These short-comings of the algorithm can be partially resolved by jittering or very careful weighting of the data samples, however, the trained classi-

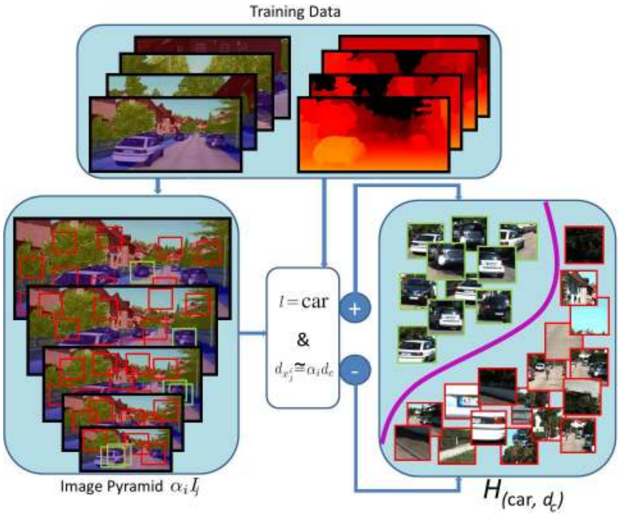


Figure 1. Schematic description of the training of our semantic depth classifier. Positive training samples for each semantic class, projected to the canonical depth d_c using the ground truth depth, are trained against other semantic classes and against samples of the same class projected to other than the canonical depth. Such a classifier is able to predict a semantic class and **any** depth by applying appropriate image transformations.

fier would still not be intrinsically unbiased.

A typical argument against the data-driven depth estimation is, that to successful perform this task, we need to be able to recognize and understand the scene. Thus one should wait, until sufficiently good recognition approaches are developed. For some recognition tasks this is already the case. Recent progress in computer vision and machine learning led to the development of algorithms [15, 27, 29, 28], that are able to successfully categorize images into hundreds [4] or even thousands [3] of different object classes. Further investigation reveals, that the success of these methods lies in how the problem is constrained; the objects of interest are typically scaled to the size of an image and under this setting carefully designed feature representations become much more discriminative. Based on this observation it is apparent, that the devil that limits the performance of algorithms for computer vision tasks, lies in the scale misalignment of the data due to the perspective geometry.

Standard semantic classifiers are trained to be discriminative between semantic classes, but robust to the change of scale. Such dissonance between the desired properties makes learning unnecessarily hard. For an object detection problem [2, 5, 27] the varying 2D scale of an object is typically handled by scaling the content of bounding boxes tightly surrounding the objects to the same size, and building a feature vector for each individual bounding box after this transformation. Without this crucial step, the performance of the detection methods decreases dramatically. In case geometry of the scene is known, or it could be reliably estimated, the location of bounding boxes can be constrained to be on the specific location, such as on the ground plane [11]. However, these approaches can be used only for foreground objects with specific spatial extent, shape and size, "*things*". For semantic segmentation task with background classes, "*stuff*", such as a road, a building, grass or a tree, such an approach is not suitable. Scaling bounding boxes surrounding lawns of grass to the same size does not make the feature representations more discriminative. However, there still exists a concept of scale for *stuff* classes tied to their real-world physical size. The physical size of a blade of grass, a window of a building or a leaf of a tree varies much less than the size of a lawn, of a building or of a tree. Consequently, the most suitable alignment, applicable to both *things* and *stuff*, is the normalization to the same physical size. This has been recognized in the scenarios with the Kinect camera, where the depth is known. The classifier using features normalized with respect to the measured depth [22] typically perform significantly better.

The mutual dependencies of visual appearance of a semantic class and its geometrical depth suggest that the problems of semantic segmentation and depth estimation should be solved jointly. It has been shown in [16] that conditioning depth on the semantic segmentation in a two-stage algorithm leads to a significant performance improvement. For stereo and multi-view images, [14, 9] have demonstrated that joint semantic segmentation and 3D reconstruction leads to a better result than performing each task in isolation. In these approaches, the rather weak source of mutual information utilized is the distribution of height [14] or surface normals [9] of different semantic classes.

In this paper we show, that using the properties of the perspective geometry we can reduce the learning of a pixel-wise depth classifier to a much simpler classifier predicting only the likelihood of a pixel being at an arbitrarily fixed *canonical* depth. The likelihoods for any other depths can be obtained by applying the same classifier after appropriate image manipulations. Such transformation of the problem to the canonical depth removes the training data bias towards certain depths and the effect of perspective. The approach can be straight-forwardly generalized to multiple semantic classes, improving both depth estimation and semantic segmentation performance by directly targeting the weaknesses of independent approaches. Conditioning the semantic label on the depth provides the way to align the data to their physical size, and conditioning depth on the semantic class helps the classifier to distinguish between ambiguities of the otherwise ill-posed problem.

We perform experiments on the very constrained street-scene KITTI data set [6] and the very challenging NYU2 indoor dataset [25], where no assumptions about the layout of the scene can be made. Our algorithm significantly outperforms independent depth estimation and semantic segmentation approaches, and obtains comparable results in the semantic segmentation domain with methods, that use full RGB-D data. Our pixel-wise classifier can be directly placed into any competing recognition or depth estimation frameworks to further improve the results; either as a unary potential for CRF recognition approaches [13] or as predictions for fitting the planes to the super-pixels [20].

2. An unbiased depth classifier

First, we define the notation. Let I be an image and $\alpha * I$ an image I geometrically scaled by a factor α. Let $W^{w,h}(I, x)$ be the (sub-)window of an image I of the size $w \times h$, centered at given point x. Any translation-invariant classifier $H_d(x)$, predicting the likelihood of pixel x being at the depth $d \in \mathcal{D}$ has to be a function of the arbitrarily large fixed size $w \times h$ sub-window centered at the point x:

$$H_d(x) := H_d(W^{w,h}(I, x)). \tag{1}$$

The perspective geometry is characterized by the feature that objects are scaled with their inverse distance from the observer's centre of projection. Thus, for any unbiased depth classifier $H_d(x)$ the likelihood of the depth d of any pixel $x \in I$ should be the same as the likelihood of the depth d/α of the corresponding pixel of the scaled image $\alpha * I$:

$$H_d(W^{w,h}(I,x)) = H_{d/\alpha}(W^{w,h}(\alpha * I, \alpha x)). \quad (2)$$

This property is crucial to keep the classifier robust to the fluctuations in the training data, which are always present for small and average-sized data sets. It seems straightforward, but it has not been used in any previous data-driven depth estimation approach [20, 1, 16].

This property implies, that the depth classification can be reduced to a much simpler prediction of whether the pixel $x \in I$ is at any arbitrarily fixed *canonical* depth d_c. The response for any other depth d can be obtained by applying the same classifier H_{d_c} to the appropriately scaled image by a factor d/d_c as:

$$H_d(W^{w,h}(I,x)) = H_{d_c}(W^{w,h}(\frac{d}{d_c} * I, \frac{d}{d_c}x)). \quad (3)$$

Thus, the problem of depth estimation is converted into the estimation of which transformation (scaling) would project the pixel into the canonical depth. The special form of the classifier directly implies how it should be learnt. In the training stage, a classifier should learn to distinguish the training samples transformed to the canonical depth from the training samples transformed to the depths other than the canonical depth. The details largely depend on the framework chosen; e.g. in the classification framework the problem is treated as a standard 2-label positives vs negatives problem, in the ranking framework the response for a training sample transformed to the canonical depth should be larger (by a sufficient margin if appropriate) than the response for a sample transformed to any other than the canonical depth.

Our classifier has several advantages over the direct learning of the depth from the feature representations of pixels. First, to predict a certain object (for example a car) at a certain depth d does not require a similar object to be seen at the same depth during the training stage. Second, our classifier does not have a problem with unbalanced training data, which is always present for a multi-class classifier or regressor. Intuitively, closer objects consist of more points and some object may have appeared at certain depth in the training data more often just by chance. These problems of a multi-class classifier or regressor could partially be resolved by jittering the data, using suitable sampling or reweighting of the training points; however, a direct enforcement of the property (2) is bound to be a better and more principled solution.

3. Semantic depth classifier

Single-view depth estimation is in general an ill-posed problem. Several ambiguities could be potentially resolved if the depth classifier was conditioned on the semantic label.

Training a classifier, which should be on one hand discriminative between semantic classes, but on the other robust to the change of scale, is unnecessarily hard. The problem would be much easier if the training samples were scale-aligned. The most suitable alignment, applicable to both *things* and *stuff*, is the normalization according to the physical size, and that is exactly, what the projection to the canonical depth (4) does.

The generalization of the depth classifier to multiple semantic classes can be done straight-forwardly by learning a joint classifier $H_{(l,d_c)}(W^{w,h}(I,x))$, predicting whether a pixel x takes a semantic label $l \in \mathcal{L}$ and is at the canonical depth d_c. By applying (4), the response of the classifier for any other depth d is:

$$H_{(l,d)}(W^{w,h}(I,x)) = H_{(l,d_c)}(W^{w,h}(\frac{d}{d_c} * I, \frac{d}{d_c}x)). \quad (4)$$

The advantage of our classifier being unbiased towards certain depths is now more apparent. An alternative approach of learning a $|\mathcal{D}||\mathcal{L}|$-class classifier or a $|\mathcal{L}|$ depth regressors would require a very large amount of training data, sufficient to represent a distribution for each label in a cross product of semantic and depth labels. In the training stage, our classifier should learn to distinguish the training samples of each class transformed to the canonical depth from the training samples of other classes and samples transformed to the depths other than the canonical depth. The transformation to the canonical depth is not applied for the *sky* class (for outdoor scenes) and the depth during test time is automatically assigned to ∞.

4. Implementation details

Transforming the window around each training sample to the canonical distance independently with a consequent calculation of features is computationally infeasible in practice. Thus, we discretize the problem of depth estimation during test stage into a discrete set of labels $d_i \in \mathcal{D}$. The error of a prediction based on the scale of objects is expected to grow linearly with the distance, suggesting that the neighbouring depths d_i and d_{i+1} should have a fixed ratio $\frac{d_{i+1}}{d_i}$, chosen depending on the desired accuracy. This allows us to transform the problem into a classification over a pyramid of images $\alpha_i * I = \frac{d_i}{d_c} * I$ for each training or test image I.

For pixels at the depth $\alpha_i d_c$, the scaling of an image by α_i corresponds to the transformation to the canonical depth. Thus in the training stage, a point of the image pyramid $x^i_j \in (\alpha_i * I)$ is used as a positive or negative sample based on how close is the ground truth depth of the corresponding pixel in the original non-scaled image $d_{x^i_j} = d_{(x_j/\alpha_i)}$ to the depth $\alpha_i d_c$. If their ratio is close to 1, e.g.

$$\max(\frac{d_{x^i_j}}{\alpha_i d_c}, \frac{\alpha_i d_c}{d_{x^i_j}}) < \delta_{POS}, \quad (5)$$

the pixel x_j is used as a positive for a corresponding semantic class and negative for all other classes. If they are sufficiently different, e.g.

$$\max(\frac{d_{x^i_j}}{\alpha_i d_c}, \frac{\alpha_i d_c}{d_{x^i_j}}) > \delta_{NEG}, \tag{6}$$

the scaling by α_i does not transform the sample to the canonical depth d_c and thus is used as a negative for all semantic classes.

In the training stage each object of the same real-world size and shape should have the same influence on the learnt classifier, no matter how far they are. Thus, the samples are sampled from $\alpha_i * I$ with the same subsampling for all α_i, and are used as positives or negatives if they satisfy corresponding constraints (5) or (6) respectively.

Transforming the problem into the canonical depth aligns the data based on their real-world physical size, which could be significantly different for different semantic classes. Thus, the most suitable classifiers are context-based with an automatically learnt context size, such as [24, 23, 13]. Following the methodology of the multi-feature [13] extension of TextonBoost [24], the dense features, namely texton [18], SIFT [17], local quantized ternary patters [12] and self-similarity features [21], are extracted for each pixel in each image in a pyramid $\alpha * I$. Each feature is clustered into 512 visual words using k-means clustering and for each pixel a soft weighting for 8 nearest neighbours is calculated using distance-based weighting [7] with an exponential kernel. The feature representation for a window $W^{w,h}(\alpha_i * I, x^i_j)$ consists of a concatenation of the soft-weighted bag-of-words representations over its fixed random set of 200 sub-windows as in [24]. The multi-class boosted classifier [26] is learnt as a sum of decision stumps, comparing one dimension of the feature representation to a threshold $\theta \in T$. Unlike in [24], the set of thresholds T is found independently for each particular dimension by uniformly splitting its range in the training set. Long feature vectors (512×200 for each feature) for each pixel can not be kept in memory, but have to be calculated *on fly* using integral images [24] for each image in the pyramid and each visual word of a feature. We implement several heuristics to decrease the resulting memory requirements. The soft weights from $(0, 1)$ are approximated using 1-byte. The integral images are built only for a sub-window of an image, that covers all the features of given visual word, using an integer type ($1 - 8$ bytes) based on the required range for each individual visual word for each image. The multiple features are, unlike in [13], fused using late fusion, e.g. the classifiers for each feature are trained independently and eventually averaged. Thanks to these heuristics, the requirements for memory dropped approximately $40\times$ for the NYU2 dataset [25] to below $32GB$. An approximately $10\times$ drop was due to the integral image updates and $4\times$ due to the late fusion.

5. Experiments

We tested our algorithm on KITTI [6] and NYU2 [25] datasets. The KITTI dataset was chosen to demonstrate the ability of our classifier to learn depth for semantic classes with a relatively small number of training samples. The NYU2 dataset was used to show that depth can be predicted for an unconstraint problem with no assumptions about the layout of the scene. Furthermore, we show that for both datasets learning of the problem jointly leads to an improvement of the performance.

5.1. KITTI data set

The KITTI data set [6] consists of a large number of outdoor street scene images of the resolution 1382×512, out of which 194 images contain sparse disparity maps obtained by Velodyne laser scanner. We labelled the semantic segmentation ground truth for the 60 images with ground truth depth and split them into 30 training and 30 test images. The label set consists of 12 semantic class labels (see Table 3). Three semantic classes (bicycle, person and sign) with high variations were ignored in the evaluation due to the insufficient training data (only 2 instances in the training set). We aimed to recognize depth in the range of $2-50$ meters with a maximum relative error $\delta = \max(\frac{d_{gt}}{d_{res}}, \frac{d_{res}}{d_{gt}}) < 1.25$, where d_{res} is the estimated depth and d_{gt} the ground truth depth. Thus, we set $\delta_{POS} = 1.25$. Visual recognition of depth with higher precision is very hard for human also. The canonical depth was set to 20 meters. Training samples were taken as negatives if their error exceeded $\delta_{NEG} = 2.5$. Training samples with error between δ_{POS} and δ_{NEG} were ignored. Quantitative comparison to the state-of-the-art class-only unary classifier [13] is given in the table 3. Quantitative comparison of the Make3D [20], trained with the same data, with our depth-only and joint depth semantic classifier is given in the table 2. Our joint classifier significantly outperformed competing algorithms in both domains. Qualitative results are given in the figure 3. Qualitative comparison of depth-only and joint classifier is given in figure 4. The distribution of relative errors of estimated depth is given in figure 6.

5.2. NYU2 data set

The NYU2 data set [25] consist of 1449 indoor images of the resolution 640×480, containing ground truth for semantic segmentation with 894 semantic classes and pixel-wise depth obtained by a Kinect sensor. For the experiments we used the subset of 40 semantic classes as in [19, 25, 8]. The images were split into 725 training and 724 test images. We aimed to recognize depth in the range $1.2 - 8.5$ meters with a maximum relative error $\delta = \max(\frac{d_{gt}}{d_{res}}, \frac{d_{res}}{d_{gt}}) <$

1.25. The canonical depth was set to 6.9 meters. Training samples were taken as positives if their error was below $\delta_{POS} = 1.25$, as negatives if their error exceeded $\delta_{NEG} = 2.5$. Quantitative comparison to the class-only unary classifier [13] and state-of-the-art algorithms using both RGB and depth during test time is given in the table 1. Our classifier performance was comparable to these methods even in these unfair conditions. Quantitative results in the depth domain can be found in the table 2. There was no other classifier we could have compared to; all other methods are constrained only to very specific scenes with strong layout constraints, such as road scenes. The performance was higher than for the KITTI dataset mainly due to the significantly more narrow range of depths.

RGB methods		RGB-D methods		
Class-only [13]	Joint classifier	[19]	[25]	[8]
34.85	37.11	38.23	37.64	45.29

Table 1. *Quantitative comparison in the frequency-weighted intersection vs union measure on the* 40*-class NYU2 dataset. Our pixel-wise method outperformed the baseline non-scale-adaptive method and obtained comparable results to the methods that use full RGB-D data during test time.*

	KITTI			NYU2	
	Make3D	Depth-only	Joint	Depth-only	Joint
$\delta < 1.25$	26.21%	43.83%	47.00%	44.35%	54.22%
$\delta < 1.25^2$	48.24%	68.01%	72.09%	70.82%	82.90%
$\delta < 1.25^3$	64.20%	82.19%	85.35%	85.90%	94.09%

Table 2. *Quantitative comparison of Make3D [20], our depth-only classifier and joint classifier on the KITTI and NYU2 dataset in the ratio of pixels correctly labelled, depending on the maximum allowed relative error* $\delta = \max(\frac{d_{gt}}{d_{res}}, \frac{d_{res}}{d_{gt}})$*, where* d_{res} *is the estimated depth and* d_{gt} *the ground truth depth.*

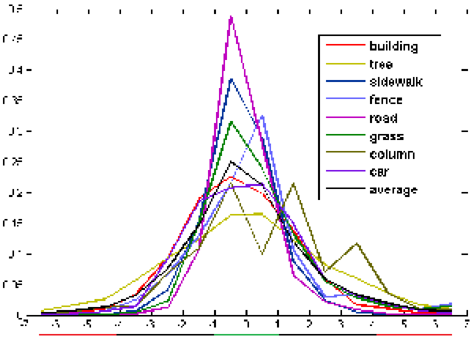


Figure 2. The distribution of the relative errors of an estimated depth given a semantic label for the KITTI dataset in the log space with the base 1.25 for each semantic class. Average distribution is calculated excluding the sky label. Green and red line below the x-axis indicates, in which interval the training samples were used as positives and negatives respectively.

6. Discussion and conclusions

In this paper we proposed a new pixel-wise classifier, that can jointly predict a semantic class and a depth label from a single image. In general, the obtained results look very promising. In the depth domain the reduction of the problem into classification of one depth only turned up to be very powerful due to its intrinsic dealing with the dataset biases. In the semantic segmentation domain we showed the importance of proper alignment, leading to quantitatively better results. The main weaknesses of the method are the inability to deal with low resolution images, very large requirements in terms of hardware and apparent inability to locate the objects more precisely for semantic classes with high variance. Obtained results suggest future work in three directions: further alignment of orientation using estimated normals, estimation of depth as a latent variable during training for the datasets with small amount of images (or none) with ground truth depth, and development of different forms of regularizations suitable for the problem. The naïve use of standard Potts model pairwise potentials did not lead to an improvement, because this form of regularization typically removed all the small distant objects, and thus we omitted these results in the paper. However, our joint depth/semantic classifier has the potential to enable richer pairwise potentials representing expected spatial relations between objects of different classes.

Acknowledgements We gratefully acknowledge the support of the 4DVideo ERC Starting Grant Nr. 210806.

References

[1] O. Barinova, V. Konushin, A. Yakubenko, K. Lee, H. Lim, and A. Konushin. Fast automatic single-view 3-d reconstruction of urban scenes. In *European Conference on Computer Vision*, 2008. 1, 3

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Conference on Computer Vision and Pattern Recognition*, 2005. 2

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition*, 2011. 2

[4] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples an incremental bayesian approach tested on 101 object categories. In *Workshop on GMBS*, 2004. 2

[5] P. F. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Conference on Computer Vision and Pattern Recognition*, 2008. 2

[6] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition*, 2012.

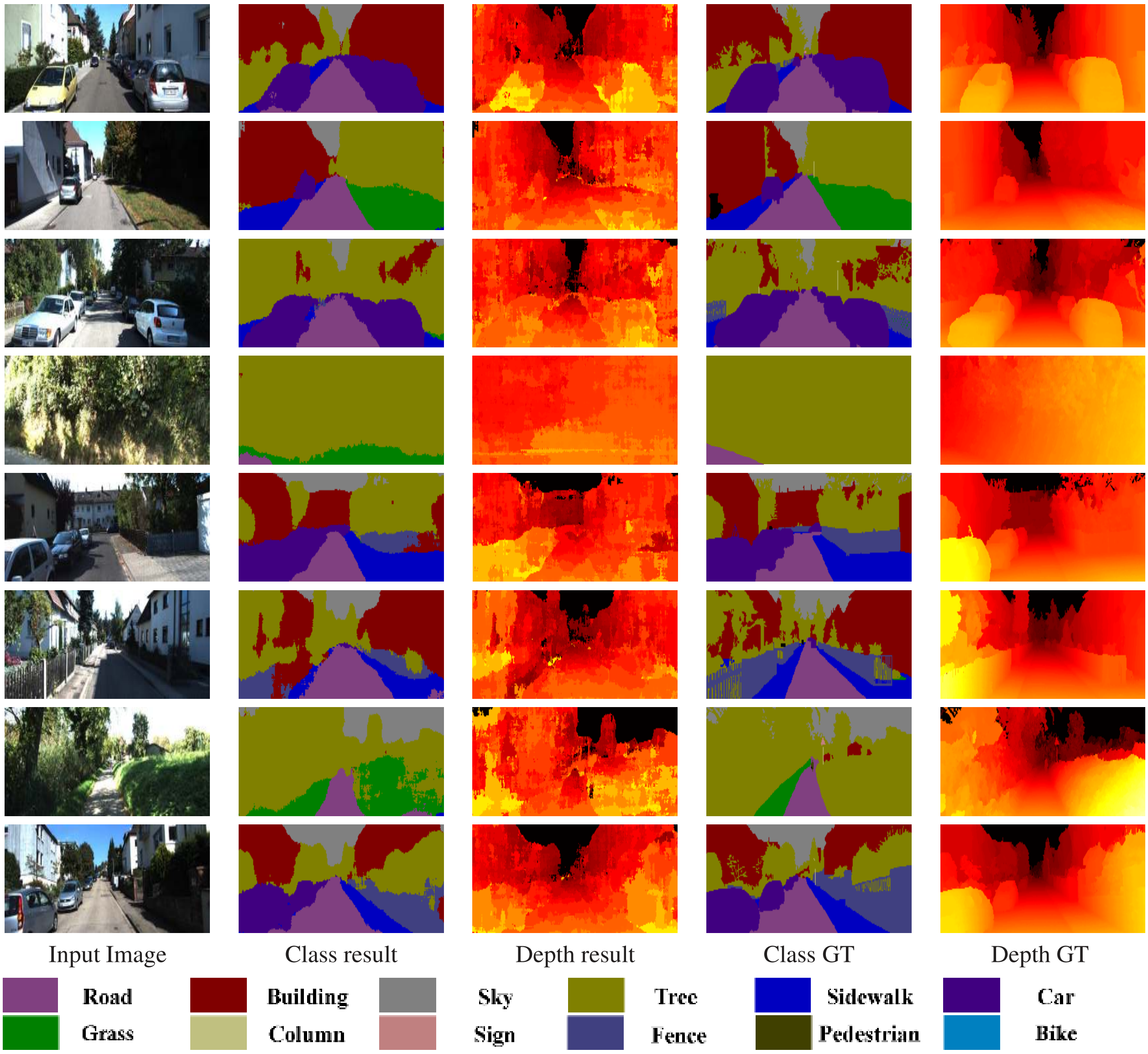


Figure 3. The pixel-wise result of our classifier on the test set. Note, that no additional assumption about the structure of the scene. Each pixel is classified individually based on the window surrounding the pixel without using 2D location. Most mislabellings were for ambiguous pairs of classes, such grass vs tree (image 7), sidewalk vs road (6), building vs fence (3). Note that the noisiness of the classifier depends on the depth, and distant objects are recognized with higher precision, unlike for a class-only classifier.

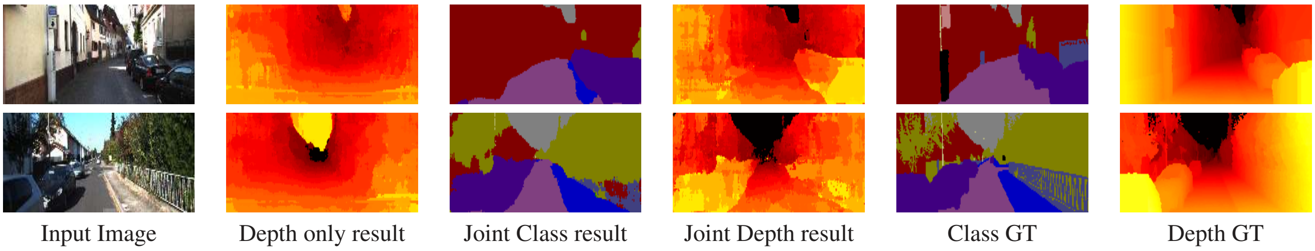
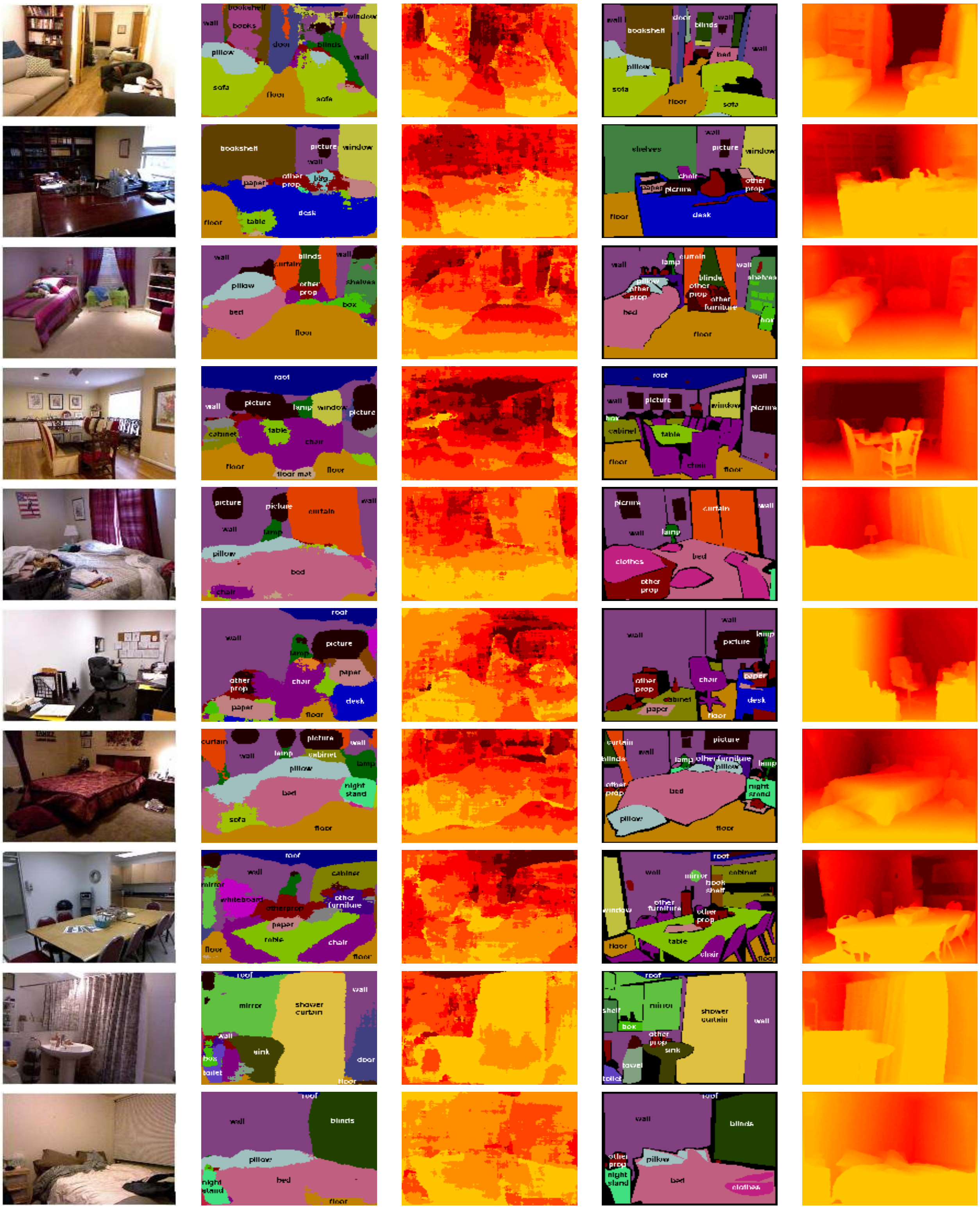


Figure 4. Comparison of single view disparity estimation without and with semantic classes. Legend for semantic classes in the figure above. Depth only classifier tends to correctly estimate depth only for dominant classes (building, road, tree) and ignore smaller classes, such as car or column/pole. The depth-only classifier also can not deal with the sky class.



Input Image Class result Depth result Class GT Depth GT

Figure 5. Qualitative comparisons of the unary classifier on the NYU2 dataset. Our classifier typically recognizes the gist of the scene in both semantic segmentation and depth domains. However, the classifier is context-based and often the objects are not localized exactly and boundaries do not match the ground truth. Typical misclassifications are between visually similar classes, such as desk vs table (image 2), floor vs floor mat (4), window vs mirror (8); due to the dataset inconsistencies, such as bookshelf labelled as shelf (2) or as books (1); or for classes not strictly defined, such other prop (5) or other furniture (7).

	Global	Average	Building	Tree	Sky	Sidewalk	Fence	Road	Grass	Column	Car
Class-only classifier [13]	80.2	66.2	87.0	82.8	89.7	68.4	31.6	**84.8**	61.2	7.3	83.2
Joint classifier	**82.4**	**72.2**	**87.2**	**84.6**	**91.6**	**76.5**	**39.4**	83.2	**69.9**	**28.5**	**88.9**

Table 3. *Quantitative results in the recall measure for the Kitti data set. Our scale adjusting classifier outperformed the class-only baseline classifier [13]. Most improvement was obtained on the rare classes and on the pixels belonging to objects far from the camera.*

2, 4

[7] J. C. V. Gemert, J. Geusebroek, C. J. Veenman, and A. W. M. Smeulders. Kernel codebooks for scene categorization. In *European Conference on Computer Vision*, 2008. 4

[8] S. Gupta, P. Arbelaez, and J. Malik. Perceptual organization and recognition of indoor scenes from RGB-D images. *Conference on Computer Vision and Pattern Recognition*, 2013. 4, 5

[9] C. Hane, C. Zach, A. Cohen, R. Angst, and M. Pollefeys. Joint 3D scene reconstruction and class segmentation. In *Conference on Computer Vision and Pattern Recognition*, 2013. 2

[10] D. Hoiem, A. Efros, and M. Hebert. Geometric context from a single image. In *International Conference on Computer Vision*, 2005. 1

[11] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. In *Conference on Computer Vision and Pattern Recognition*, 2006. 2

[12] S. u. Hussain and B. Triggs. Visual recognition using local quantized patterns. In *European Conference on Computer Vision*, 2012. 4

[13] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Associative hierarchical CRFs for object class image segmentation. In *International Conference on Computer Vision*, 2009. 2, 4, 5, 8

[14] L. Ladicky, P. Sturgess, C. Russell, S. Sengupta, Y. Bastanlar, W. Clocksin, and P. H. S. Torr. Joint Optimization for Object Class Segmentation and Dense Stereo Reconstruction. *International Journal of Computer Vision*, 2012. 2

[15] 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*, 2006. 2

[16] B. Liu, S. Gould, and D. Koller. Single image depth estimation from predicted semantic labels. In *Conference on Computer Vision and Pattern Recognition*, 2010. 1, 2, 3

[17] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004. 4

[18] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 2001. 4

[19] X. Ren, L. Bo, and D. Fox. RGB-(D) scene labeling: Features and algorithms. In *Conference on Computer Vision and Pattern Recognition*, 2012. 4, 5

[20] A. Saxena, S. H. Chung, and A. Y. Ng. Learning depth from single monocular images. In *Advances in Neural Information Processing Systems*, 2005. 1, 2, 3, 4, 5

[21] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *Conference on Computer Vision and Pattern Recognition*, 2007. 4

[22] J. Shotton, A. Fitzgibbon, M. Cook, and A. Blake. Real-time human pose recognition in parts from single depth images. In *Conference on Computer Vision and Pattern Recognition*, 2011. 2

[23] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *Conference on Computer Vision and Pattern Recognition*, 2008. 4

[24] J. Shotton, J. Winn, C. Rother, and A. Criminisi. *TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *European Conference on Computer Vision*, 2006. 4

[25] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision*, 2012. 2, 4, 5

[26] A. Torralba, K. Murphy, and W. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Conference on Computer Vision and Pattern Recognition*, 2004. 4

[27] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. In *International Conference on Computer Vision*, 2009. 2

[28] J. Wang, J. Yang, K. Yu, F. Lv, T. S. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Conference on Computer Vision and Pattern Recognition*, 2010. 2

[29] K. Yu, T. Zhang, and Y. Gong. Nonlinear learning using local coordinate coding. In *Advances in Neural Information Processing Systems*, 2009. 2