Discriminatively Trained Dense Surface Normal Estimation

Ľubor Ladický, Bernhard Zeisl, and Marc Pollefeys
{lubor.ladicky,bernhard.zeisl,marc.pollefeys}@inf.ethz.ch

ETH Zürich, Switzerland

Abstract. In this work we propose the method for a rather unexplored problem of computer vision - discriminatively trained dense surface normal estimation from a single image. Our method combines contextual and segment-based cues and builds a regressor in a boosting framework by transforming the problem into the regression of coefficients of a local coding. We apply our method to two challenging data sets containing images of man-made environments, the indoor NYU2 data set and the outdoor KITTI data set. Our surface normal predictor achieves results better than initially expected, significantly outperforming state-of-the-art.

1 Introduction

Recently, single-view reconstruction methods, estimating scene geometry directly by learning from data, have gained quite some popularity. While resulting 3D reconstructions of such methods are of debatable quality, coarse information about the 3D layout of a scene has shown to help boost the performance of applications such as object detection [1], semantic reasoning [2] or general scene understanding [3].

The principal underlying idea behind these methods [4,5,6] is, that particular structures have a certain real world size, and thus their size in an image gives rise to the scene depth. We argue that this is a rather weak hypothesis, since structures are likely to exist at different size in reality and perspective projection distorts them. As a consequence it renders the problem of single image depth estimation ill-posed in general. However, perspective cues are not harmful, but actually helpful, because they carry information about the local surface orientation and allow to reason about the scene, for example about the viewpoint of the camera. We argue that it is beneficial to directly estimate first order derivatives of depth, i.e. surface normals, as it can provide more accurate results than estimation of absolute depth. In addition we do not need to worry about depth discontinuities, e.g. due to occlusions, which are difficult to detect and harm single image reconstruction [5,6].

While data-driven normal estimation seems to be a more promising approach, it has not been exploited much so far. We believe this is due to the lack of available ground truth data, which is hard to obtain, as recording requires accurate

capturing devices. With the recent advances in low cost commodity depth sensors such as Kinect, ToF cameras or laser scanners, acquisition was made easier and there are multiple data sets [7,8] available nowadays, which should foster research in this direction.

The importance of surface normal estimation has been already recognized long before such data was available. Due to the lack of data, proposed approaches [9,10,11] had to rely purely on the knowledge of underlying physics of light and shading. Thus, resulting methods work only under strong assumptions about the knowledge of locations of light sources and properties of the material, such as the assumption of Lambertian surfaces. However, these approaches do not work in more complex scenarios such as indoor or outdoor scenes, and thus are not applicable for general problems. The first approach, that directly tries to estimate surface normals from the data was proposed in [12]. The method aims to extract a set of both visually-discriminative and geometrically-informative primitives from training data. For test images the learned detector fires at sparse positions with similar appearance and hypothesizes about the underlying surface orientations by means of the learned primitives. Hoiem et al. [13] do not directly estimate normal directions, but formulate the task as a labeling problem with more abstract surface orientations, such as left- or right-facing, vertical, etc. in order to estimate the 3D contextual frame of an image. In [14] Gupta et al. extracted a qualitative physical representation of an outdoor scene by reasoning about the pairwise depth relations (and thus also not via absolute depth); though their model is approximated to consist of blocks only. Other authors have simplified the task to be more robust and incorporated strong orientation priors such as vanishing points and lines [15,16] or Manhattan world constrains [17,18,19].

In this work we aim to extract surface normals for each pixel in a single image without any measured knowledge about the underlying 3D scene geometry. We present a discriminative learning approach to estimate pixel-wise surface orientation solely from the image appearance. We do not incorporate any kind of geometric priors; rather we utilize recent work in image labelling as often used for semantic image segmentation, where context enhanced pixel-based and segment-based feature representations proved best performances. For the semantic segmentation problem it is reasonable to assume that all pixels within a detected segment share the same label, i.e. segments correspond to objects. However, for normal estimation this assumption of label-consistency holds only for planar regions, such as segments on a wall; for segments related to non-planar objects, e.g. a cylindrical shaped pot, it is violated.

We account for this property and propose a feature representation, that combines the cues of pixel-wise and segment-based methods. The strength of our approach stems from the fact that we join both representations and intrinsically learn, when to use which. It has the desired effect that results tend to follow segment (and by this object) boundaries, but do not necessarily have to. Then we formulate the surface normal estimation as a regression of coefficients of the local coding, to make the learning problem more discriminative. Finally, we adapt the standard boosting framework to deal with this specific learning problem.

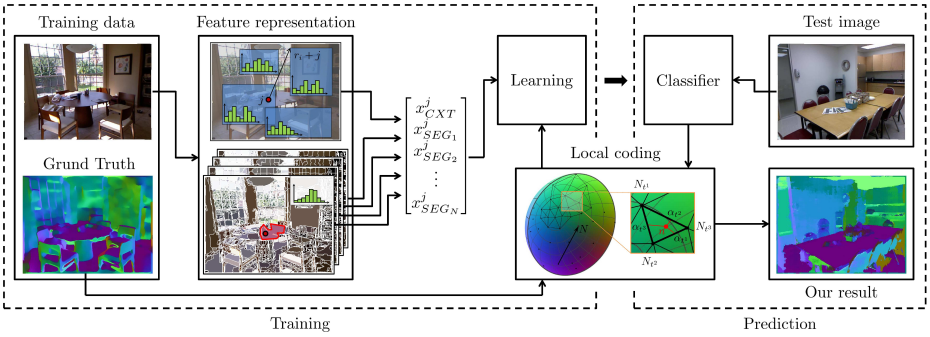

Fig. 1. Workflow of the algorithm. In the training stage images are segmented using multiple unsupervised segmentations, dense features are extracted and discriminative feature representations combining contextual and segment-based features are built. Ground truth normals are approximated using the local coding by a weighted sum of representative normals and the discriminative regressor for these coefficients is trained. In the test stage the likelihood of each representative normal is predicted by the classifier and the output normal is recovered as a weighted sum of representative normals. The colours of the half-sphere represent corresponding normal directions.

The whole pipeline is illustrated in Figure 1. We apply our method to two data sets from two different man-made environments - indoors [7] and outdoors [8]. Our classifier obtains surprisingly good results, successfully recognizing surface normals for a wide range of different scenes.

Our paper is structured as follows: In Section 2 we explain the details of our feature representation and draw connections to related work. Our learning procedure is illustrated in Section 3. In Section 4 we describe more implementation details, in Section 5 the acquisition of the ground truth data, and finally Section 6 reports our experimental results.

2 Feature representation for surface normal estimation

The standard label prediction pipeline for recognition tasks in computer vision consists of dense or sparse feature extraction, the composition of suitable compact discriminative feature representations, and the application of an appropriate machine learning algorithm, capable of discriminating between different labels corresponding to the different possible outcomes for the given task. For pixel-wise tasks, such as semantic segmentation, the feature representations are typically built either over pixels, or alternatively over segments (a.k.a. superpixels), obtained using any unsupervised segmentation technique [20,21,22,23]. Next we elaborate more on both approaches.

2.1 Context-based pixel-wise methods

For pixel-based approaches only a local feature vector of a pixel itself is insufficient to predict the label. Thus, a certain form of context [24,25,26,27], combining information from neighbouring pixels capturing a spatial configuration of features, has to be used. The context is captured either using a pixel-based or rectangle-based context representation. In a pixel-based context approach [25,26], the contextual information for a given pixel is obtained by concatenating individual feature vectors of neighbouring pixels, placed at a fixed set of displacements from the reference pixel. In a rectangle-based context approach [24,27], the feature representation for a pixel j is obtained by concatenating bag-of-words representations $bow(r_i + j)$ for a fixed set of rectangles $r_i \in R$, placed relative to the pixel j; i.e. $\mathbf{x}^j_{CXT} = [bow(r_1 + j), bow(r_2 + j), .., bow(r_{|R|} + j)]$. For both forms of context, multiple local features can be used jointly [27].

In practice, even for a small data set it is impossible to store these huge feature representations for all training samples in memory. Typically, the feature representation values are evaluated on the fly during the learning process. For a pixel-based context, the context is quickly evaluated from individual feature vector maps stored in memory. For a rectangle-based approach, the contextual information is obtained efficiently using integral images, calculated individually for each visual word. The response for one individual dimension, corresponding to the number of certain visual words in a rectangle, placed relative to the given pixel, is often referred to as a shape-filter response [24].

The predictions using context based approaches are typically very noisy, do not follow object boundaries, and thus require some form of regularization [28,27]. The rectangle-based context representation is typically more robust and leads to better performance quantitatively [24,25,27]. On the other hand the pixel-based approach is much faster and with a suitable learning method it can be evaluated in real-time [25,26] during testing. In this work we are more interested in high performance of our method, and thus we build on the superior rectangle-based rather than the faster pixel-based context.

2.2 Segment-based methods

Segment-based methods [29,30,31,32] are built upon the assumption, that predicted labels are consistent over segments obtained by unsupervised segmentation. This assumption plays two significant roles. First, the learning and evaluation over pixels can be reduced to a much smaller problem over segments, which allows for more complex and slower learning methods to be used, such as kernel SVMs [33]. Second, it allows us to build robust feature representations by combining features of all pixels in each segment. The most common segment-based representation is a L^1-normalized bag-of-words [29,30,31,27,32], modelling the distribution of visual words within a segment. Recently, several other alternatives beyond bag-of-words have been proposed [34,35,36], suitable for labelling of segments.

Fig. 2. The example segmentations obtained by 4 different unsupervised methods [20,21,22,23]. The segments largely differ in terms of smoothness, shape consistency or variances of size. The notion of their quality largely depends on the task they are applied to. For semantic segmentation similar sized segments have typically more discriminant feature representations, but methods producing segments of different scales are more suitable for enforcing label consistency in segments [28]. For normal estimation a single unsupervised segmentation method can not produce label-consistent segments in general, e.g. the lamp in an image is not planar at any scale. Optimally, the learning method should decide by itself, which method – if any – and which features are more suitable for each specific task.

All standard segmentation methods [20,21,22,23] have free colour and spatial range parameters, that can be tuned specially for each individual task or data set, and are either hand-tuned or chosen based on an appropriate quality measure [37] to satisfy the label consistency in segments. However, even choosing the best unsupervised segmentation method is harder than it seems. Human perception of the segment quality is very misleading, see Figure 2. Methods producing segments of large variation in size [20], capturing information over the right scale, may look visually more appealing, but the feature representations obtained using methods producing similar sized segments [22,23] may be more stable, and thus more discriminative. Choosing the right parameters is even harder; to obtain segments that will not contain multiple labels, the parameters of the unsupervised segmentation method must be chosen to produce a large number of very small segments. However, at that point the information in each segment is often not sufficient to correctly predict the label. Two kinds of approaches have been proposed to deal with this problem. In [32], the feature representation of segments also includes a feature representation of the union of neighbours to encode contextual information. Alternatively in [27] multiple segmentations are combined in the CRF framework by finding the smooth labelling that agrees with most of the predictions of individual pixel-wise and segment-wise classifiers and enforces label-consistency of segments as a soft constraint (see also [28]).

For normal estimation, the assumption of label-consistency is even more damming. It would imply, that all segments must be planar. It is a very good assumption for floor or walls, however, some objects are intrinsically not planar, such as cylindrical thrash bins or spherical lamp shades.

2.3 Joint context-based pixel-wise and segment-method

In our approach we propose a joint feature representation, that can deal with the weaknesses of individual context-based and segment-based methods. In partic-

ular, we overcome the inability of context-based approaches to produce smooth labellings tightly following boundaries and to capture the information on the correct object-based level, and the inability of segment-based methods to learn from a suitable-sized context. Unlike in [27], the contextual and segment cues are combined directly during in the learning stage.

This can be achieved by a very simple trick. Any learning method defined over segments, with a loss function weighted by the size of the segment, is equivalent to a learning method defined over pixels, with the feature vector $\mathbf{x}^j_{SEG} = \mathbf{x'}^{s(j)}_{SEG}$, where $s(j)$ is the segment the pixel j belongs to, and $\mathbf{x'}^k_{SEG}$ is any segment-based feature representation of the segment k. This transformation allows us to trivially combine feature representations over multiple segmentations as $\mathbf{x}^j_{MSEG} = (\mathbf{x}^j_{SEG_1}, \mathbf{x}^j_{SEG_2}, .., \mathbf{x}^j_{SEG_N})$, where $\mathbf{x}^j_{SEG_i}$ is the representation for an i-th segmentation. Learning over such pixel representations becomes equivalent to learning over intersections of multiple segmentations [38]. And finally, we concatenate this representation with contextual information $\mathbf{x}^j = (\mathbf{x}^j_{CXT}, \mathbf{x}^j_{MSEG})$. For normal estimation, this representation is powerful enough to learn the properties, such as *wall*-like features are more discriminative for segments from a particular segmentation or context-driven features can determine correct normals for spherical objects. Unlike in [27], the framework is able to potentially enforce label inconsistency in segments, which are identified to be non-planar.

3 Learning normals

Due to a large dimensionality of the problem, we preferred learning algorithms, that use only a small randomly sampled subset of dimensions in each iteration, such as random forests [39] or Ada-boost [40]. Direct application of a even a simple linear regression would not be feasible. In practice Ada-boost typically performs better in terms of performance [24,25], random forests in terms of speed. Similarly to the choice of contextual representation, we chose better over faster. Intuitively the most discriminative contextual features will correspond to the local configuration of corner-like dense features, each one discriminant for a narrow range of normals. Thus, we make the learning problem simpler by lifting it to the problem of regressing coefficients of local coding [41,42,43], typically used in the feature space to increase the discriminative power of linear SVM classifiers. Standard multi-class Ada-boost [44] is designed for classification over a discrete set of labels, not for continuous regression. We adapt the learning algorithm to deal with a set of continuous ground truth labels, both during training and evaluation.

3.1 Local coding in the label space

We start by performing standard k-means clustering on the set of ground truth normals n^j in the training set. In each iteration we back-project each cluster mean to the unit (half-)sphere. We refer to the cluster mean as the reference

normal N_k. The Delaunay triangulation is evaluated on the set of reference normals to obtain the set of triangles T, where each triangle $t_i \in T$ is an unordered triplet of cluster indexes $\{t_i^1, t_i^2, t_i^3\}$. For each ground truth normal n^j we find the closest triangle $t(j)$ by solving the non-negative least squares problem [41]:

$$t(j) = \arg\min_{t_i \in T} \min_{\alpha_{t_i}^j} |n^j - \sum_{p=1}^{3} \alpha_{t_i^p}^j N_{t_i^p}|^2, \tag{1}$$

such that $\sum_{p=1}^{3} \alpha_{t_i^p}^j = 1$ and $\alpha_{t_i^p}^j \geq 0, \forall p \in \{1,2,3\}$. Each ground truth normal is approximated by $n^j \approx \sum_k \alpha_k^j N_k$, where 3 potentially non-zero coefficients α_k^j come from the corresponding problem (1) for the triplet in $t(j)$ and for all other coefficients $\alpha_k^j = 0$. In general, any reconstruction based local coding can be used.

3.2 Multi-class boosting with continuous ground truth labels

A standard boosting algorithm builds a strong classifier $H(\mathbf{x}, l)$ for a feature vector $\mathbf{x}$ and a class label $l \in \mathcal{L}$ as a sum of weak classifiers $h(\mathbf{x}, l)$ as:

$$H(\mathbf{x}, l) = \sum_{m=1}^{M} h^{(m)}(\mathbf{x}, l), \tag{2}$$

where M is the number of iterations (boosts). The weak classifiers $h(\mathbf{x}, l)$ are typically found iteratively as: $H^{(m)}(\mathbf{x}, l) = H^{(m-1)}(\mathbf{x}, l) + h^{(m)}(\mathbf{x}, l)$.

Standard multi-class Ada-boost [44] with discrete labels minimizes the expected exponential loss:

$$J = \sum_{l \in \mathcal{L}} E\left[e^{-z^l H(\mathbf{x}, l)}\right], \tag{3}$$

where the $z_l \in \{-1, 1\}$ is the membership label for a class l. A natural extension to continuous ground truth labels is to minimize the weighted exponential loss defined as:

$$J = \sum_{l \in \mathcal{L}} E\left[\alpha_l e^{-H(\mathbf{x}, l)} + (1 - \alpha_l) e^{H(\mathbf{x}, l)}\right], \tag{4}$$

where α_l is the coefficient of a cluster mean l of the local coding, in our case corresponding to a reference normal. This cost function can be optimized using adaptive Newton steps by following the procedure in [40]. Each weak classifier $h^{(m)}(\mathbf{x}, l)$ is chosen to minimize the second order Taylor expansion approximation of the cost function (4). Replacing expectation by an empirical risk leads to a minimization of the error [40,44]:

$$\begin{aligned} J_{wse} = \sum_{l \in \mathcal{L}} \sum_{j} & (\alpha_l^j e^{-H^{(m-1)}(\mathbf{x}^j, l)} (1 - h^{(m)}(\mathbf{x}^j, l))^2 \\ & + (1 - \alpha_l^j) e^{H^{(m-1)}(\mathbf{x}^j, l)} (1 + h^{(m)}(\mathbf{x}^j, l))^2). \end{aligned} \tag{5}$$

Defining two sets of weights:

$$w_l^{j,(m-1)} = \alpha_l^j e^{-H^{(m-1)}(\mathbf{x}^j,l)}, \tag{6}$$

$$v_l^{j,(m-1)} = (1-\alpha_l^j) e^{H^{(m-1)}(\mathbf{x}^j,l)}, \tag{7}$$

the minimization problem transforms into:

$$J_{wse} = \sum_{l\in\mathcal{L}} \sum_j (w_l^{j,(m-1)}(1-h^{(m)}(\mathbf{x}^j,l))^2 + v_l^{j,(m-1)}(1+h^{(m)}(\mathbf{x}^j,l))^2). \tag{8}$$

The weights are initialized to $w_l^{j,(0)} = \alpha_l^j$ and $v_l^{j,(0)} = 1-\alpha_l^j$ and updated iteratively as:

$$w_l^{j,(m)} = w_l^{j,(m-1)} e^{-h^{(m)}(\mathbf{x}^j,l)}, \tag{9}$$

$$v_l^{j,(m)} = v_l^{j,(m-1)} e^{h^{(m)}(\mathbf{x}^j,l)}. \tag{10}$$

The most common weak classifier for multi-class boosting are generalized decision stumps, defined as [44]:

$$h^{(m)}(\mathbf{x},l) = \begin{cases} a^{(m)}\delta(x_{i^{(m)}} > \theta^{(m)}) + b^{(m)} & \text{if } l \in \mathcal{L}^{(m)} \\ k_l^{(m)} & \text{otherwise,} \end{cases} \tag{11}$$

where $x_{i^{(m)}}$ is one particular dimension of $\mathbf{x}$, $\mathcal{L}^{(m)} \subseteq \mathcal{L}$ is the subset of labels the decision stump is applied to; and $i^{(m)}$, $a^{(m)}$, $b^{(m)}$, $k_l^{(m)}$ and $\theta^{(m)}$ parameters of the weak classifier.

In each iteration the most discriminant weak classifier is found by randomly sampling dimensions $i^{(m)}$ and thresholds $\theta^{(m)}$ and calculating the set of remaining parameters $a^{(m)}$, $b^{(m)}$, $k_l^{(m)}$ and $\mathcal{L}^{(m)}$ by minimising the cost function (8). The parameters $a^{(m)}$, $b^{(m)}$ and $k_l^{(m)}$ are derived by setting the derivative of (8) to 0, leading to a close form solution:

$$b^{(m)} = \frac{\sum_{l\in\mathcal{L}^{(m)}} \sum_j (w_l^{j,(m-1)} - v_l^{j,(m-1)}) \delta(x_{i^{(m)}} \leq \theta^{(m)})}{\sum_{l\in\mathcal{L}^{(m)}} \sum_j (w_l^{j,(m-1)} + v_l^{j,(m-1)}) \delta(x_{i^{(m)}} \leq \theta^{(m)})}, \tag{12}$$

$$a^{(m)} = \frac{\sum_{l\in\mathcal{L}^{(m)}} \sum_j (w_l^{j,(m-1)} - v_l^{j,(m-1)}) \delta(x_{i^{(m)}} > \theta^{(m)})}{\sum_{l\in\mathcal{L}^{(m)}} \sum_j (w_l^{j,(m-1)} + v_l^{j,(m-1)}) \delta(x_{i^{(m)}} > \theta^{(m)})} - b^{(m)}, \tag{13}$$

$$k_l^{(m)} = \frac{\sum_j (w_l^{j,(m-1)} - v_l^{j,(m-1)})}{\sum_j (w_l^{j,(m-1)} + v_l^{j,(m-1)})}, \forall l \notin \mathcal{L}^{(m)}. \tag{14}$$

The subset of labels $\mathcal{L}^{(m)}$ is found greedily by iterative inclusion of additional labels, if they decrease the cost function (8).

3.3 Prediction of the surface normal

During test time the responses $H(\mathbf{x}, l)$ for each reference normal are evaluated and the most probable triangle is selected by maximizing:

$$t(\mathbf{x}) = \arg\max_{t_i \in T} \sum_{p=1}^{3} e^{H(\mathbf{x}, t_i^p)}. \tag{15}$$

The non-zero local coding coefficients for each index k of a triangle $t(\mathbf{x})$ are obtained as:

$$\alpha^j_{t(\mathbf{x})^k} = \frac{e^{H(\mathbf{x}, t(\mathbf{x})^k)}}{\sum_{p=1}^{3} e^{H(\mathbf{x}, t(\mathbf{x})^p)}}, \tag{16}$$

and the resulting normal $n^j(\mathbf{x})$ for a pixel j is recovered by computing the linear combination projected to the unit sphere:

$$n^j(\mathbf{x}) = \frac{\sum_{p=1}^{3} \alpha^j_{t(\mathbf{x})^p} N_{t(\mathbf{x})^p}}{|\sum_{p=1}^{3} \alpha^j_{t(\mathbf{x})^p} N_{t(\mathbf{x})^p}|}, \tag{17}$$

corresponding to the expected value under standard probabilistic interpretation of boosted classifier [40]. Weighted prediction leads to better performance both qualitatively and quantitatively (see Figure 6).

4 Implementation details

The complete work flow of our method is shown in the Figure 1. In our implementation four dense features were extracted for each pixel in each image - texton [45], SIFT [46], local quantized ternary patters [47] and self-similarity features [48]. Each feature was clustered into 512 visual words using k-means clustering, and for each pixel a soft assignment for 8 nearest cluster centres is calculated using distance-based exponential kernel weighting [49]. The rectangle-based contextual part of the feature representation consists of a concatenation of soft-weighted bag-of-words representations over 200 rectangles, resulting in $200 \times 4 \times 512$ dimensional feature vector $\mathbf{x}^j_{CXT}$. The Segment-based part $\mathbf{x}^j_{SEG}$ consists of soft-weighted bag-of-words representations over 16 unsupervised segmentations obtained by varying kernel parameters of 4 different methods, 4 segmentations each - Mean-shift [20], SLIC [23], normalized cut [21] and graph-cut based segmentation [22]. In the boosting process, the same number of dimensions from the contextual and segment part of the feature representation were sampled in each iteration, to balance different dimensionality of these representations. This was achieved by increasing the sampling probability of each dimension of the segment-part $\frac{200}{16}$ times. The strong classifier consisted of 5000 weak classifiers. The whole learning procedure has been applied independently for 5 different colour models - RGB, Lab, Luv, Opponent and GreyScale. Each individual classifier was expected to perform approximately the same, and thus

the final classifier response was simply averaged over these 5 classifiers without any additional training of weights for each individual colour space. In practice this averaging procedure has similar effects to the use of multiple decision trees in the random forest; it leads to smoother results and avoids over-fitting to noise.

5 Ground truth acquisition

Required ground truth measurements about the underlying 3D scene geometry can be captured with active devices, such as laser scanners, commodity depth sensors, stereo cameras or from dense 3D reconstructions of image collections. In all cases the depth measurements are likely to contain noise, which will get amplified in their first derivatives. Since our aim is to obtain piecewise constant normal directions – as reflected typically in man-made environments – we leverage (second order) Total Generalized Variation (TGV) [50] for denoising. The optimization is formulated as a primal-dual saddle point problem and solved via iterative optimization; for more detail we refer the interested reader to [51]. Normals are then computed on the 3D point cloud for each point in a local 3D spatial neighborhood. Compared to computations on the depth map itself, this guarantees that measurements of distant structures in 3D which project to neighboring pixels do not get intermixed. Finally, for point-wise normal estimation we utilize a least squares regression kernel in a RANSAC scheme in order to preserve surface edges. Given the quality of the depth data, obtained ground truth normals look visually significantly better than the direct first derivatives of the original raw depth data. However, the quality of normals often degrades in the presence of reflective surfaces, near image edges or in regions without sufficient amount of direct depth measurements.

6 Experiments

We trained our classifier on the indoor NYU2 [7] and on the outdoor KITTI [8] data set to demonstrate the ability of our method to predict normals in various man-made environments.

6.1 NYU2 data set

The NYU2 data set [7] consists of 795 training and 654 test images of resolution 640×480, containing pixel-wise depth obtained by a Kinect sensor. The data set covers a wide range of types of indoor scenes, such as offices, bedrooms, living rooms, kitchens or bathrooms. To train the classifier, the ground truth normals were clustered into 40 reference normals. The mean angle between neighbouring reference normals was 18 degrees. The training of the classifier took three weeks on five 8-core machines. Thus, the parameters of our method (such as number of normal clusters, segmentations or boosts) have been chosen based on our expert knowledge and not tweaked at all. The evaluation took 40 minutes per image on

a single core; however, it can be easily parallelized. The results are significantly better than initially expected. Our classifier consistently managed to successfully predict normals for various complicated indoor environments. The qualitative results are shown in Figure 3. Quantitative comparisons of our classifier (weighted by local coding coefficients and hard-assigned to the normal cluster with the highest response) to the state-of-the-art [12] are shown in Figure 6. The results are reported for full images (561×427 sub-window) and on the masks (as in [12]), that define the regions containing direct depth measurements. Approximately for one half of the pixels in the masks the predicted normals were within 20 degrees angle. The numbers do not reflect the quality of our results, because even in a flat surfaces the normals of the ground truth often vary by 10 or 20 degrees (see for example the ground truth of the bottom-right image in Figure 3). To get an idea of the interpretation of the error, the average angular error of the visually very appealing result on the test image in Figure 1 is 28 degrees.

The success of our method on this data set lead us to further experiments using the already trained classifier. We applied it to the Reconstruction-Meets-Recognition depth challenge (RMRC) data set [52], consisting of 558 images. Qualitative comparisons with the method [12] are shown in Figure 4. Ground truth depth images are not publicly available. Our method was able to successfully predict normals for images that looked visually similar to the NYU2 data. However, for images, that were not taken upright (as in NYU2), our classifier predicted normals as if they were. We evaluated our classifier also on images captured by ourselves, see Figure 5.

6.2 KITTI data set

The KITTI depth data set [8] consists of 194 training images and 195 test outdoor images, containing sparse disparity maps obtained by a Velodyne laser scanner. The distribution of normals within an image seemed much more predictable than for indoor scenes, due to a very typical image layout and lower variety of normals. Thus, to train a classifier we clustered normals only into 20 clusters. The training took five days on five 8-core machines. The evaluation took 20 minutes per image on a single core. Qualitative results are shown in Figure 7. The ground truth depths are not publicly available, thus we do not provide any quantitative results.

7 Conclusions and Future work

In this paper we proposed a method for dense normal estimation from RGB images by combining state-of-the-art context-based and segment-based cues in a continuous Ada-Boost framework. The results have the potential to be applied to several other reconstruction problems in computer vision, such as stereo, single-view or 3D volumetric reconstruction, as a geometric prior for their regularization. In the future we would like to do further research along these lines.

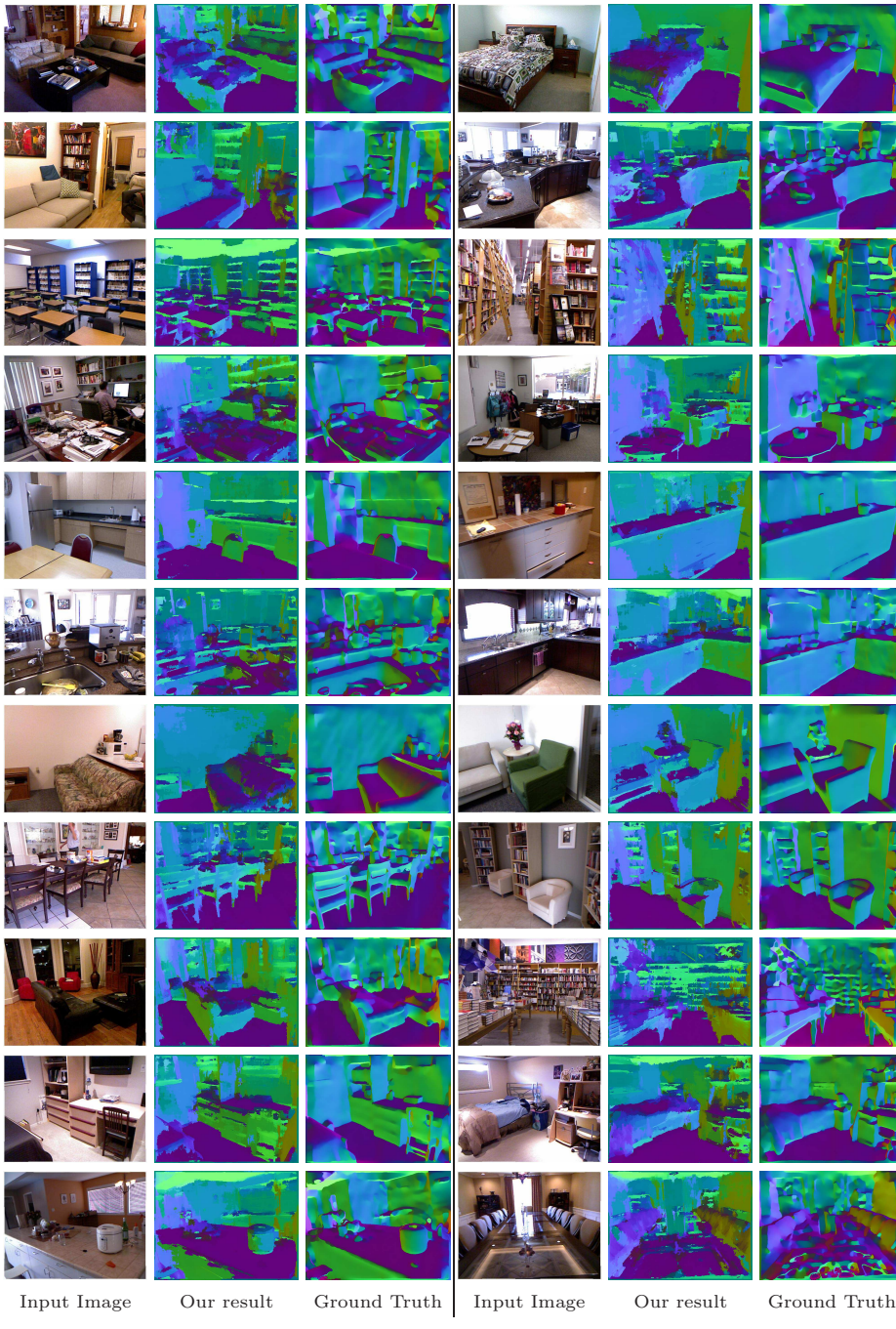

Fig. 3. Qualitative results on NYU2 data set. The colours, corresponding to different normals, are shown in Figure 1. Our algorithm consistently predicted high quality surface normals for a single image. Note, the imperfect quality of the ground truth labelling (see for example the image in bottom-right).

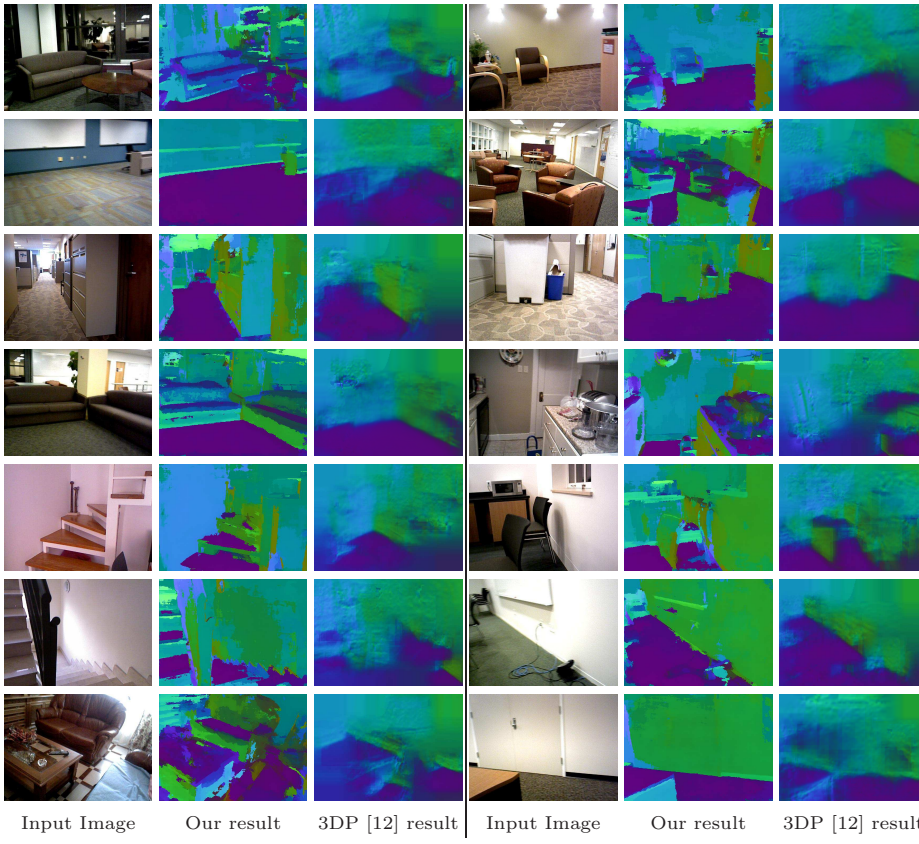

Fig. 4. Qualitative comparison of our method with 3DP [12] on the RMRC data set. Both methods were trained on the NYU2 training data. Several images of RMRC data set were taken from unusual viewpoints, not present in the NYU2 data set, causing troubles to both methods.

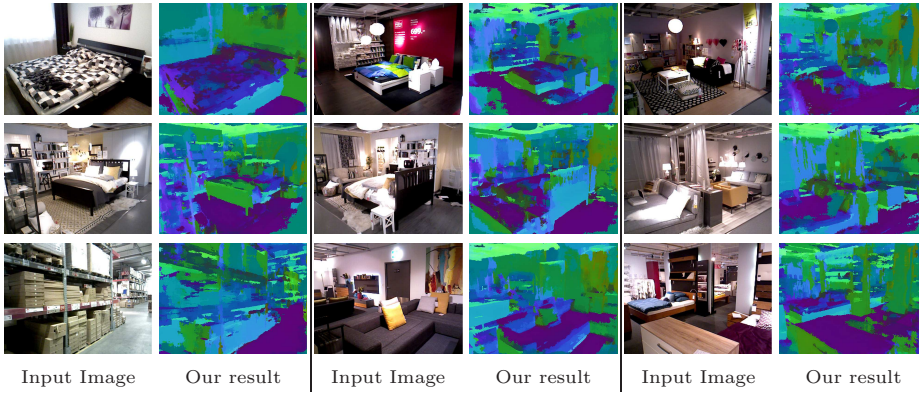

Fig. 5. Additional results of our method using the classifier trained on NYU2 data set.

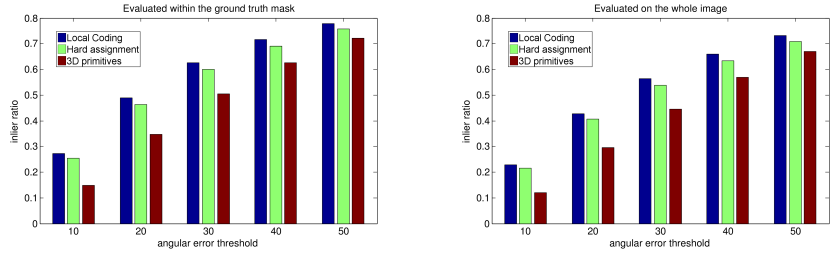

Fig. 6. Quantitative comparison of locally coded and hard assigned version of our method with 3DP [12] method on the NYU2 data set. The performance is evaluated in term of ratio of pixels within different angular error (10, 20, 30, 40 and 50) and calculated either over the masks, corresponding to the regions with direct depth measurements; or over the whole image. Our method estimated approximately half of normals in the masked region within 20 degree error. The numbers do not fully reflect the quality of our results due to the imperfection of the ground truth (see Figure 3).

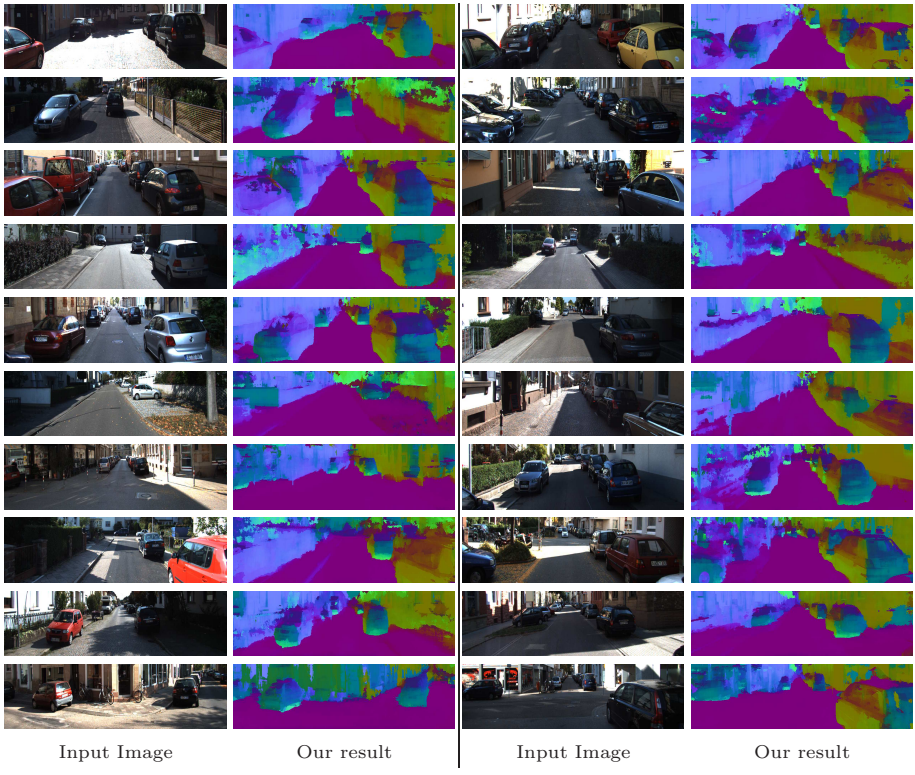

Fig. 7. Qualitative results of our method on the KITTI data set. The ground truth colours are the same as for the NYU2 data set. Our classifier essentially learnt the typical geometrical layout of the scene and the spatial configuration of surface normals of cars seen from various viewpoints.

References

1. Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. In: Conference on Computer Vision and Pattern Recognition. (2006)
2. Ladicky, L., Shi, J., Pollefeys, M.: Pulling things out of perspective. In: Conference on Computer Vision and Pattern Recognition. (2014)
3. Hoiem, D., Efros, A.A., Hebert, M.: Closing the loop on scene interpretation. In: Conference on Computer Vision and Pattern Recognition. (2008)
4. Saxena, A., Chung, S.H., Ng, A.Y.: 3-D Depth Reconstruction from a Single Still Image. International Journal of Computer Vision (2007)
5. Saxena, A., Sun, M., Ng, A.Y.: Make3D: learning 3D scene structure from a single still image. Transactions on Pattern Analysis and Machine Intelligence (2009)
6. Liu, B., Gould, S., Koller, D.: Single image depth estimation from predicted semantic labels. Conference on Computer Vision and Pattern Recognition (2010)
7. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from RGBD images. In: European Conference on Computer Vision. (2012)
8. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: Conference on Computer Vision and Pattern Recognition. (2012)
9. Horn, B.K.P., Brooks, M.J., eds.: Shape from Shading. MIT Press (1989)
10. Mallick, S.P., Zickler, T.E., Kriegman, D.J., Belhumeur, P.N.: Beyond lambert: reconstructing specular surfaces using color. In: Conference on Computer Vision and Pattern Recognition. (2005)
11. Ikehata, S., Aizawa, K.: Photometric stereo using constrained bivariate regression for general isotropic surfaces. In: Conference on Computer Vision and Pattern Recognition. (2014)
12. Fouhey, D., Gupta, A., Hebert, M.: Data-driven 3d primitives for single image understanding. In: International Conference on Computer Vision. (2013)
13. Hoiem, D., Efros, A.A., Hebert, M.: Recovering Surface Layout from an Image. International Journal of Computer Vision (2007)
14. Gupta, A., Efros, A., Hebert, M.: Blocks world revisited: Image understanding using qualitative geometry and mechanics. In: European Conference on Computer Vision. (2010)
15. Delage, E., Lee, H., Ng, A.: A Dynamic Bayesian Network Model for Autonomous 3D Reconstruction from a Single Indoor Image. In: Conference on Computer Vision and Pattern Recognition. (2006)
16. Barinova, O., Konushin, V., Yakubenko, A., Lee, K., Lim, H., Konushin, A.: Fast Automatic Single-View 3-d Reconstruction of Urban Scenes. In: European Conference on Computer Vision. (2008)
17. Lee, D.C., Hebert, M., Kanade, T.: Geometric reasoning for single image structure recovery. In: Conference on Computer Vision and Pattern Recognition. (2009)
18. Flint, A., Mei, C., Reid, I., Murray, D.: Growing semantically meaningful models for visual SLAM. In: Conference on Computer Vision and Pattern Recognition. (2010)
19. Flint, A., Mei, C., Murray, D., Reid, I.: A dynamic programming approach to reconstructing building interiors. In: European Conference on Computer Vision. (2010)
20. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. Transactions on Pattern Analysis and Machine Intelligence (2002)

21. Shi, J., Malik, J.: Normalized cuts and image segmentation. Transactions on Pattern Analysis and Machine Intelligence (2000)
22. Zhang, Y., Hartley, R.I., Mashford, J., Burn, S.: Superpixels via pseudo-boolean optimization. In: International Conference on Computer Vision. (2011)
23. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Susstrunk, S.: SLIC superpixels compared to state-of-the-art superpixel methods. Transactions on Pattern Analysis and Machine Intelligence (2012)
24. Shotton, J., Winn, J., Rother, C., Criminisi, A.: *TextonBoost*: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: European Conference on Computer Vision. (2006)
25. Shotton, J., Johnson, M., Cipolla, R.: Semantic texton forests for image categorization and segmentation. In: Conference on Computer Vision and Pattern Recognition. (2008)
26. Shotton, J., Fitzgibbon, A., Cook, M., Blake, A.: Real-time human pose recognition in parts from single depth images. In: Conference on Computer Vision and Pattern Recognition. (2011)
27. Ladicky, L., Russell, C., Kohli, P., Torr, P.H.S.: Associative hierarchical CRFs for object class image segmentation. In: International Conference on Computer Vision. (2009)
28. Kohli, P., Ladicky, L., Torr, P.H.S.: Robust higher order potentials for enforcing label consistency. In: Conference on Computer Vision and Pattern Recognition. (2008)
29. Yang, L., Meer, P., Foran, D.J.: Multiple class segmentation using a unified framework over mean-shift patches. In: Conference on Computer Vision and Pattern Recognition. (2007)
30. Batra, D., Sukthankar, R., Tsuhan, C.: Learning class-specific affinities for image labelling. In: Conference on Computer Vision and Pattern Recognition. (2008)
31. Galleguillos, C., Rabinovich, A., Belongie, S.: Object categorization using co-occurrence, location and appearance. In: Conference on Computer Vision and Pattern Recognition. (2008)
32. Boix, X., Cardinal, G., van de Weijer, J., Bagdanov, A.D., Serrat, J., Gonzalez, J.: Harmony potentials: Fusing global and local scale for semantic image segmentation. International Journal on Computer Vision (2011)
33. Guyon, I., Boser, B., Vapnik, V.: Automatic capacity tuning of very large vc-dimension classifiers. In: Advances in Neural Information Processing Systems. (1993)
34. Perronnin, F., Sánchez, J., Mensink, T.: Improving the fisher kernel for large-scale image classification. In: European Conference on Computer Vision. (2010)
35. Zhou, X., Yu, K., Zhang, T., Huang, T.S.: Image classification using super-vector coding of local image descriptors. In: European Conference on Computer Vision. (2010)
36. Carreira, J.a., Caseiro, R., Batista, J., Sminchisescu, C.: Semantic segmentation with second-order pooling. In: European Conference on Computer Vision. (2012)
37. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: International Conference on Computer Vision. (2007)
38. Pantofaru, C., Schmid, C., Hebert, M.: Object recognition by integrating multiple image segmentations. In: European Conference on Computer Vision. (2008)
39. Breiman, L.: Random forests. In: Machine Learning. (2001)
40. Friedman, J., Hastie, T., Tibshirani, R.: Additive Logistic Regression: a Statistical View of Boosting. The Annals of Statistics (2000)

41. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science (2000)
42. Yu, K., Zhang, T., Gong, Y.: Nonlinear learning using local coordinate coding. In: Advances in Neural Information Processing Systems. (2009)
43. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T.S., Gong, Y.: Locality-constrained linear coding for image classification. In: Conference on Computer Vision and Pattern Recognition. (2010)
44. Torralba, A., Murphy, K., Freeman, W.: Sharing features: efficient boosting procedures for multiclass object detection. In: Conference on Computer Vision and Pattern Recognition. (2004)
45. Malik, J., Belongie, S., Leung, T., Shi, J.: Contour and texture analysis for image segmentation. International Journal of Computer Vision (2001)
46. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (2004)
47. Hussain, S.u., Triggs, B.: Visual recognition using local quantized patterns. In: European Conference on Computer Vision. (2012)
48. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: Conference on Computer Vision and Pattern Recognition. (2007)
49. Gemert, J.C.V., Geusebroek, J., Veenman, C.J., Smeulders, A.W.M.: Kernel codebooks for scene categorization. In: European Conference on Computer Vision. (2008)
50. Bredies, K., Kunisch, K., Pock, T.: Total Generalized Variation. SIAM Journal on Imaging Sciences **3** (2010) 492–526
51. Chambolle, A., Pock, T.: A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging. Journal of Mathematical Imaging and Vision (2010)
52. Urtasun, R., Fergus, R., Hoiem, D., Torralba, A., Geiger, A., Lenz, P., Silberman, N., Xiao, J., Fidler, S.: Reconstruction Meets Recognition Challenge. http://ttic.uchicago.edu/ rurtasun/rmrc/ (2013)