Shading Annotations in the Wild

Balazs Kovacs Sean Bell Noah Snavely Kavita Bala
Cornell University

Abstract

Understanding shading effects in images is critical for a variety of vision and graphics problems, including intrinsic image decomposition, shadow removal, image relighting, and inverse rendering. As is the case with other vision tasks, machine learning is a promising approach to understanding shading—but there is little ground truth shading data available for real-world images. We introduce Shading Annotations in the Wild (SAW), a new large-scale, public dataset of shading annotations in indoor scenes, comprised of multiple forms of shading judgments obtained via crowdsourcing, along with shading annotations automatically generated from RGB-D imagery. We use this data to train a convolutional neural network to predict per-pixel shading information in an image. We demonstrate the value of our data and network in an application to intrinsic images, where we can reduce decomposition artifacts produced by existing algorithms. Our database is available at http://opensurfaces.cs.cornell.edu/saw.

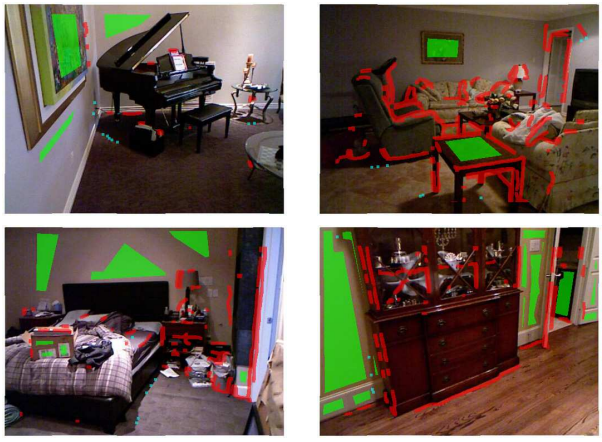

Figure 1: *Examples of annotations in the SAW dataset.* **Green** indicates regions of near-constant shading (but with possibly varying reflectance). **Red** indicates edges due to discontinuities in shape (surface normal or depth). **Cyan** indicates edges due to discontinuities in illumination (cast shadows). Using these annotations, we can learn to classify regions of an image into different shading categories.

1. Introduction

Understanding images requires reasoning about the shapes and materials in scenes, where the appearance of objects is modulated by illumination. A large body of research in scene understanding has focused on shape and materials, with lighting often overlooked or discounted as a nuisance factor. However, understanding shading and illumination in images is critical for a variety of problems in vision and graphics, including intrinsic image decomposition, shadow detection and removal, image relighting, and inverse rendering. How can we make progress on understanding illumination in natural images? As with other problem domains, we believe that data is key. Large-scale datasets such as ImageNet [10], COCO [26], Places [36], and MINC [6] have had significant impact in advancing research in object detection, scene classification and understanding, and material recognition. This success motivates the creation of a similar dataset for shading information.

In this paper, we present a new, large-scale crowdsourced dataset of *Shading Annotations in the Wild* (SAW). An important challenge in constructing a dataset of shading information is deciding what manner of shading information to collect, and how to collect it. We consider several possible approaches to collecting such data, and note that a key subproblem across many tasks is to determine whether an image edge is due to variation in reflectance, illumination, or some other cause (as with the Retinex algorithm for intrinsic images [24]). This observation leads us to collect two types of shading annotations in a large set of images: (1) image regions of approximately constant shading, and (2) examples of discontinuities in illumination (i.e., cast shadow boundaries), or shape (e.g., depth or surface normal discontinuities). These kinds of annotations are illustrated in Figure 1. We show how to collect these annotations at scale using a combination of crowdsourcing and automatic processing. Our dataset includes 15K shadow boundary points and 24K constant shading regions from nearly 7K photos.

Using our new dataset, we train a convolutional neural network (CNN) to identify various types of shading in new

images, and demonstrate competitive performance in this shading classification task compared to a number of baselines. Finally, we demonstrate the value of our data and learned network in an application to intrinsic image decomposition, where we can reduce mistakes commonly made by existing algorithms, namely, when texture due to reflectance is incorrectly attributed to shading.

In summary, our contributions are:

- a new large-scale dataset of shading annotations collected via crowdsourcing,
- a CNN trained to recognize shading effects using this data, and a comparison to baseline methods, and
- an example use of this model as a smooth shading prior to improve intrinsic image decomposition.

2. Related Work

Our goal is to build a dataset specifically addressing shading in images, and large enough to be well suited for machine learning. There exist a number of related datasets, but to our knowledge, none achieve both of these goals.

Intrinsic images. Intrinsic image decomposition is a classic, ill-posed problem involving separating an image into the product of a reflectance and a shading layer. Grosse *et al*. [16] introduced the MIT Intrinsic Images dataset, containing 16 objects with ground truth reflectance and shading. This dataset has led to important progress in intrinsic image decomposition, but the small size of the dataset, and its focus on single objects rather than entire scenes, means that it is not well suited to machine learning approaches on natural images. Beigpour *et al*. capture a dataset of similar size, but with multiple illuminants [3]. Bell *et al*. released the Intrinsic Images in the Wild (IIW) dataset [5], a large-scale dataset with over 5K real-world indoor photos, with relative reflectance judgments between millions of pairs of points. However, IIW only contains information about reflectance, and thus only captures indirect information about shading. As a result, intrinsic image algorithms evaluated on IIW data can sometimes shuffle error into the shading channel without penalty. Finally, synthetic datasets (from rendered CG scenes) also provide a way to obtain ground truth shading for intrinsic images and other problems [8, 4, 7]. However, we find that synthetic scenes still cannot fully represent the complexity of natural images.

Depth datasets. Several datasets contain RGB-D (depth) data, including NYUv2 [30], SUN RGB-D [31], and many others [12]. These datasets can be used to train algorithms to predict depth or surface normals from a single image [34, 11, 25, 2, 9]. These shape cues (particularly surface normals) are related to shading, but do not capture critical illumination effects such as cast shadows. Hence, we draw on RGB-D data to augment our dataset, but use crowdsourcing to annotate additional shading information.

Other illumination datasets. Other datasets capture particular types of illumination information, such as sun direction [21], environment maps [23], or shadows in outdoor scenes [38, 22]. These datasets tend to focus exclusively on outdoor illumination (e.g., from the sun), or only support a particular task (e.g., hard shadow detection and removal [22, 17]). Others have presented algorithms for estimating illumination from images, e.g., for object insertion tasks [20], relighting [18, 28], or more general inverse rendering problems [33]. However, these generally require user input or multiple images. One of our goals is to help advance such methods for illumination modeling and editing by providing data for use in machine learning methods.

3. Shading Annotations in the Wild

Our goal is to create an extensive dataset of shading phenomena in indoor scenes. Ideally we would collect per-pixel, dense absolute shading measurements for each image, as with the MIT dataset [16]. Unfortunately, the gray spray painting method they used is not feasible for whole indoor scenes. Synthetic scenes are a potential alternative to provide dense ground truth, but we found that it is difficult to build a large enough dataset of synthetic images that can fully represent the complex illumination in the real world.

Bell *et al*. targeted a broad set of real-world scenes by annotating Flickr images in their Intrinsic Images in the Wild dataset [5]. They argued that while humans cannot provide absolute reflectance or shading values, they can disentangle reflectance from shading by making pairwise reflectance judgments. Reflectance values tend to be sparse in indoor scenes, due to the overwhelming presence of human-made objects, which is often used as a prior in the intrinsic image literature [14, 5, 37]. Conversely, this sparsity observation does not hold for shading, which is often smooth and varies over a wide intensity range in natural scenes. Bell *et al*. pointed out that this makes it harder for humans to make relative shading judgments between arbitrary point pairs in images, so they did not collect pairwise shading annotations.

Our contribution is to identify and collect useful shading annotations that human beings can provide in a crowdsourced setting, at scale and with high accuracy.

3.1. Images

To create a comprehensive dataset of shading phenomena, we chose to build on the Intrinsic Images in the Wild (IIW) dataset [5] which has complementary data on relative reflectance annotations for 5,230 images.[1] We further

[1] SAW images are a superset of IIW images except for two images (IDs: 24541, 24851), which are atypical photos that we exclude. One is a painting, and the other is a closeup of a book cover.

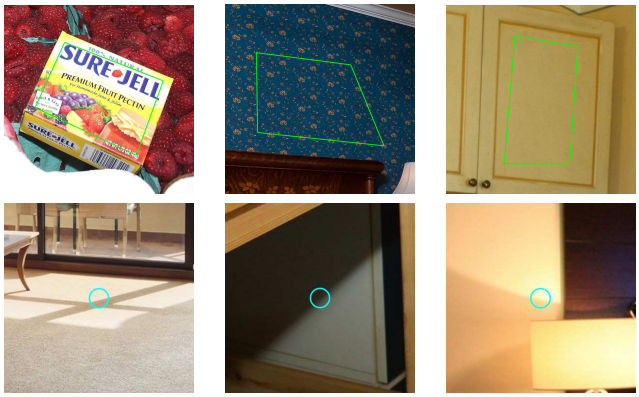

Figure 2: *Our shading annotations.* **First row:** Constant shading regions S (green polygons). **Second row:** Shadow boundary annotations NS-SB (cyan circles). The constant shading regions span the range from textured to textureless (the average color gradient magnitude for the regions shown from left to right is 3.972, 0.295, and 0.1).

Figure 3: Two types of shading annotations. **(a)** Constant shading regions (green polygons). **(b)** Shading point comparisons (red edges). Darker red indicates more confident judgments.

added 1,449 images with RGB-D data from the NYU Depth Dataset v2 [30] to have images from which we can get ground truth depth and surfaces normals. In total, the SAW dataset has 6,677 images.

3.2. Shading annotation taxonomy

Our goal is to collect shading annotations at scale. We taxonomize shading into two types: smooth (S) and non-smooth (NS), where the non-smooth shading is further split into two categories, shadow boundaries (NS-SB), and normal/depth discontinuities (NS-ND). Using a judicious combination of crowdsoucing when needed, and automatic image/scene processing when possible, our dataset includes these three types of shading annotations.

3.3. Our annotation pipeline

Pilot study. Inspired by IIW [5], our first attempt to collect shading annotations was to use the same kind of pairwise comparisons as in IIW, where workers were asked to make a series of pairwise reflectance judgments. For IIW, workers were shown a pair of points **1** and **2** in an image, and asked to specify whether: (1) **1** had a darker surface color compared to **2**, (2) **1** had a brighter surface color compared to **2**, or (E) **1** and **2** had approximately equal surface brightness (i.e., less-than, greater-than, or equal-to judgments). In our case, rather than collect pairwise reflectance comparisons, our aim was to collect pairwise *shading* annotations.

Reasoning about how lightfields differ between arbitrary points is not easy for humans [27]. Indeed, in this case humans often have to make judgments over very different regions of an image, and over different materials and shapes. Hence, we decided to allow workers to pick the points themselves instead of using the original point pairs from IIW. We created two tasks which ask workers to "pick two points with equal shading" and "pick two points with non-equal shading". Unfortunately, workers struggled with giving us good quality data for the former task. Learning from this pilot study we instead developed a new crowdsourcing pipeline to collect shading annotations that workers can confidently respond to.

Collecting S annotations. Knowing that human beings have difficulty reasoning about distant shading, we ask workers to instead annotate local regions which they select to have approximately constant shading. Since shading tends to be smooth in small regions, they can do this task reliably. Further, we get much more data from a region annotation than a pairwise comparison between two points. Thus, we were able to collect S annotations at scale with a small number of selected workers. These S annotations are collected over both IIW and the NYU dataset. See Figure 2 for examples, and Section 3.4 for details.

Collecting NS annotations. Non-smooth shading arises from a variety of causes, such as shadow boundaries or changes in the shape of a surface (e.g., through depth discontinuities or normal discontinuities). We employ a combination of automated scene/image processing and crowdsourcing to collect these annotations.

First, we note that shape discontinuities (i.e., depth or normal changes) can be obtained from existing datasets like the NYU RGB-D dataset. Therefore, instead of crowdsourcing these annotations (NS-ND), we automatically generate normal/depth discontinuities from the ground truth RGB-D data. More details are provided in Section 3.5.

Another type of non-smooth shading arises at shadow boundaries. For each image in IIW and the NYU dataset,

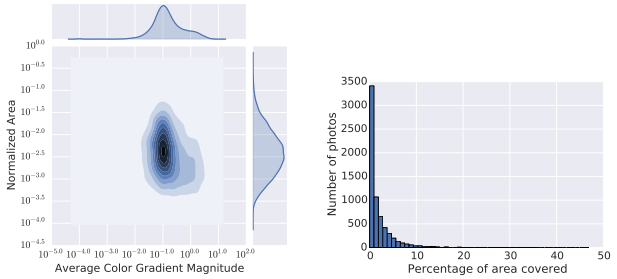

Figure 4: *Statistics of constant shading regions.* **Left:** Joint plot of the log average color gradient magnitude over each constant shading region and the log normalized area (1 means that the region covers the entire image). The gradient magnitude is correlated with how textured the region is. Textured regions are valuable because constant shading cannot be easily predicted based on simple pixel intensity measurements. **Right:** Histogram of the percentage of total image area covered by constant shading regions. As expected, most of the regions are relatively small, since it is unlikely that shading is constant over large regions. See the supplemental material for more analysis.

we allow workers to select point pairs with different shading. Since workers control the pair selection, they are able to choose cases where they can make a confident decision. We find that the pairs of selected points are often on opposite sides of sharp shadow boundaries. We use this knowledge to automatically generate candidate shadow boundary points from the pairwise data from workers, which we filter through another crowd-sourcing task to separate out true shadow boundaries (NS-SB) from shape discontinuities. See Figure 2 for examples, and Section 3.5 for details.

3.4. Collecting S annotations

For this task, each worker was asked to draw a polygon around an area which has approximately constant shading. The notion of constant shading is hard to understand for most workers, so to guide workers to submit higher quality regions we added extra criteria: the region has to be flat/smooth, opaque (i.e., non-transparent), non-glossy, and non-bumpy (i.e., have no surface normal variation). Based on our pilot study, we further restricted the region to be composed of a single type of material (e.g. wood, plastic), and not fabric, which tends to have small bumps in most situations. However, we explicitly request that workers annotate textured regions when possible, so that we do not simply collect regions with a single dominant color (e.g., painted walls). Such textured regions are very valuable, because constant shading cannot be easily predicted from simple pixel intensity measurements in these regions. We allowed eight MTurk workers who previously provided high-quality submissions to work on this task.

For quality control, we additionally sent each marked region through three filtering tasks to address common mistakes. These tasks asked workers to (1) "click on flat/smooth regions with one material type", (2) "click on glossy regions", and (3) "click on regions which have varying shading". Since these tasks are much simpler than the first task, we did not need to hand-select workers here. For each smooth shading region, we collected five responses for each of the three tasks and used CUBAM [32] to aggregate the votes into a single decision. We kept regions that passed all tests (i.e., regions that were flat/smooth with one material type, non-glossy, and did not exhibit varying shading).

In total, we collected 23,947 smooth regions (S), which on average covered 0.6% of the image area. The cost of this task was \$0.011 on average for the shading regions, plus \$0.056 for the three quality control tasks. Figure 2 (top) shows examples of annotated smooth shading regions and green polygons in Figure 3 show these regions in the context of an entire scene. Figure 4 provides insights into the quality of the constant shading region data.

3.5. Collecting NS annotations

Next, we turned our attention to non-smooth shading annotations (NS). Here we found from our pilot study that if workers are given a choice of where to position a pair of points, they can successfully decide which point has darker vs. brighter shading. However, these shading changes could be attributed to both shape changes (normal/depth discontinuities) or due to shadow boundaries. While we could crowdsource both these kinds of annotations, the shape discontinuties can be obtained directly from existing datasets. So we automatically generate NS-ND annotations, and only use crowdsourcing for the NS-SB annotations.

Auto-generated NS-ND annotations. At normal/depth discontinuities, shading tends to be non-smooth. We generate NS-ND annotations using depth maps of scenes from existing datasets such as NYU Depth Dataset v2 [30], and normal maps computed from these depth maps from [34]. Given a depth D and normal map N and thresholds τ_{depth} and τ_{normal}, we annotate a pixel p as having non-smooth shading if $(\|\nabla D\|_2)_p > \tau_{depth}$ or $(\|\nabla N\|_2)_p > \tau_{normal}$.

We ignore pixels where the Kinect camera used to capture the RGB-D imagery provides unreliable depths, using masks provided by [11]. We noticed that in some cases, these masks do not sufficiently remove all incorrect normal/depth regions, and so we use binary erosion with 3 iterations on each mask and also ignore pixels close to the image boundaries (within 5% of the image width).

Crowdsourcing NS-SB annotations. Finally, we crowdsource non-smooth shading annotations, with a pipeline focusing on shadow boundaries (Figure 5). The first task in the pipeline asks workers to select two points such that the

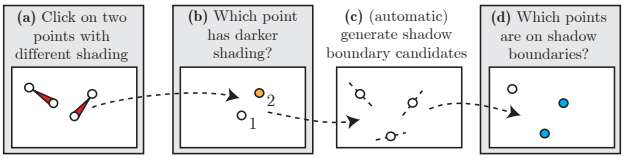

Figure 5: *Point annotation pipeline.* **(a)** Workers are asked to click on two points such that the first has darker shading than the second. **(b)** Then, 5 workers are asked to pick the point with darker shading for each point pair. **(c)** Next, we automatically generate a candidate shadow boundary point for each point pair based on image gradient. **(d)** Finally, workers are asked to select shadow boundary points.

first has darker shading than the second point. After filtering out comparisons which have non-opaque or glossy points, we collected five votes for each comparison asking which point has darker shading (Figure 5(b), similar to [5] for relative reflectance judgments). The original pair of two points counts as an additional vote for a total of 6 votes. See the supplementary for more details. We collected 97,294 shading comparisons with an average cost of $0.026. Red edges in Figure 3 show example relative shading judgments.

The last step is to generate and validate shadow boundary points (Figures 5(c) and (d)). Given the shading comparisons, we generate candidate shadow boundary points for each non-equal shading comparison by finding the point with the highest log intensity gradient magnitude on the line segment connecting the two points of the comparison (Figure 5(c)). We discard candidate points where the line segment is longer than 0.2 in normalized image coordinates, because these point pairs are too far apart and the candidate point usually lies on a shape discontinuity; or where the maximum gradient magnitude is smaller than 0.3, because such intensity differences are hard to notice. Then we asked five workers if the candidate point is on a shadow boundary (Figure 5(d)). We define the term "shadow boundary" here to exclude normal or depth discontinuities. This ensures that we can make a distinction between the automatically generated normal/depth discontinuity labels (NS-ND) from Section 3.5 and shadow boundary labels (NS-SB). We chose the final shadow boundaries with majority voting.

Using this pipeline, we obtain 15,407 shadow boundary points at an average cost of $0.039. Figure 2 (bottom) shows examples of shadow boundary annotations. We provide statistics of the collected shadow boundary points in the supplemental material.

Quality control for crowdsourcing S and NS-SB. It is important to control quality when collecting crowdsourcing data [1, 15]. Many workers misunderstand instructions or do not read them in detail. Therefore we implemented tutorials for most of our crowdsourcing tasks and did not let workers submit data until they passed the tutorial. We also ask multiple workers the same question and decide the final label by majority voting or CUBAM [32]. Finally, we employ sentinels (questions with known ground truth) to filter out workers with low accuracy.

4. Learning to Predict Shading Features

We demonstrate the utility of our shading annotation data by training a CNN to make per-pixel predictions of different types of shading features. We formulate this problem as classifying each pixel of an image into one of three classes based on the taxonomy defined in Section 3.2: smooth shading (S), normal/depth discontinuity (NS-ND), and shadow boundary (NS-SB).

4.1. Dataset processing

Before we train a classifier, we first convert our dataset into a pixel labeling for each image (note that only some pixels will be labeled, since our annotations only partially cover each image). First, we resize all images such that the maximum image dimension is 512. Next, we generate smooth shading (S) labels from our constant shading regions by taking the regions in the resized images and performing binary erosion with 3 iterations, to reduce the effect of any errors where constant-shading-region boundaries may touch shadow boundaries. This gives us 25,690,392 smooth shading pixel labels across the entire dataset.

We then generate the normal/depth discontinuity non-smooth shading (NS-ND) labels based on the resized normal/depth maps of the 1,449 NYUv2 images as described in Section 3.5 with $\tau_{normal} = 1.5$ and $\tau_{depth} = 2.0$. We manually chose the smallest thresholds where we deemed the annotations to be of high quality. Finally, we use our shadow boundary point annotations to generate the rest of the non-smooth shading (NS-SB) labels.

Note that we perform "label dilation" on the non-smooth shading labels when generating the training set: that is, we also label pixels that are very close to these non-smooth pixels within a 5×5 neighborhood. We do this to train a more conservative classifier which does not predict smooth shading very close to non-smooth shading effects. For the validation and test set, we do not perform this dilation. This way for the training, validation, and test set respectively, we get 4,758,500/1,512,257/2,418,490 NS-ND and 224,886/ 2,107/ 4,267 NS-SB labels.

4.2. Network architecture

We extend Bansal *et al.* [2]'s convolutional neural network (CNN) architecture for surface normal prediction to learn to predict shading effects in images using the Caffe deep learning framework [19]. We use the same architecture, but change the last fully-connected layer to predict the three classes described above.

4.3. Training

We assign each photo to the training/validation/test sets as follows: For photos in the original IIW set, we keep the training/test split used by [37] and add an additional training/validation split over their training set. For NYU images, we use the splits from [2]. This gives us 4,142 training, 836 validation and 1,699 test photos.

Since our training data is limited, we initialize the weights using the normal prediction net of [2], fix the weights of the convolutional part and only fine-tune the last three fully-connected layers. We experimented with fine-tuning all layers or only the last fully-connected layer, but observed worse validation performance. To avoid training bias, it is important to balance the training data across the three classes. We use a 2:1:1 balancing ratio (S : NS-ND : NS-SB) in our experiments, equivalent to a 1:1 balance between smooth and non-smooth categories.

Similar to [2], we resize each input image to 224×224 before passing it to the network, and upsample the output of the network back to the resolution of the original input image. Since not all pixels in an image are labeled and we want to enforce class balance, after passing all images in a batch through the convolutional layers, we randomly sample pixels for each class over the whole batch according to our balance ratio, and propagate features only for these sampled pixels to the rest of the network. Please see the supplemental material for detailed training parameters.

4.4. Inference

At inference time, we are interested in predicting the probability of the shading being smooth for each pixel in the image. In Figure 6 we show some example predictions shown as heatmaps of the probability of the smooth shading class (S). In the left image, the network correctly predicts smooth shading on the wall and polished stone surfaces. Of particular use are high probability predictions on textured surfaces that have smooth shading, because these are non-trivial to predict based on image intensity alone.

However, our method also makes some mistakes. In a few cases, the high probability areas "bleed over" shape discontinuities, as in the corner of the bathroom in the left image, or the sharp shape discontinuities of the trolley in the right image. In general, the network predicts smooth shading somewhat conservatively, and misses some smooth shading regions, but it does well in predicting non-smooth regions in most cases. Please see the supplemental material for further discussion and hundreds of heatmap predictions.

5. Evaluation

Since to our knowledge there are no existing algorithms for explicitly predicting the three types of shading classes we consider, we focus our evaluation on predictions of smooth shading vs. non-smooth shading, where we can use simple baselines for comparison.

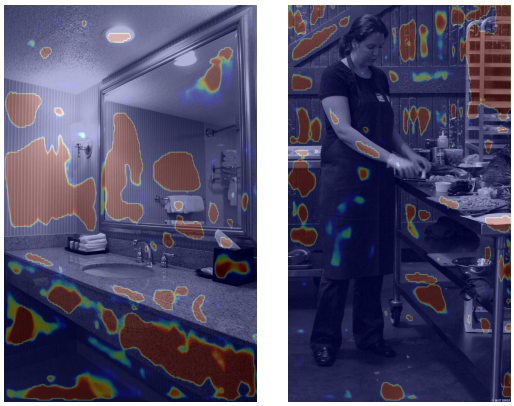

Figure 6: *Heatmaps for the predicted probability of the smooth shading class (*S*) overlaid on the original input images.* All images are selected from the test set. See Section 4.4 for discussion of these results.

5.1. Baselines

A natural set of baselines for predicting shading categories are intrinsic image algorithms, which take an input image I and decompose it into reflectance $\mathcal{R}$ and shading $\mathcal{S}$ layers. We use several state-of-the-art intrinsic image decomposition algorithms as baselines. In particular, given a decomposition $\mathcal{R}$ and $\mathcal{S}$, we classify a pixel p as smooth/non-smooth shading based on the gradient magnitude of the shading channel $\mathcal{S}$. If the gradient magnitude at p is less than a threshold τ, i.e., $\|\nabla \mathcal{S}(p)\|_2 < \tau$, then we say the predicted shading is smooth at p (otherwise, non-smooth). In practice, we found that applying a maximum filter of size 10×10 on the gradient image improved the results of these baselines, so we apply this filtering in our tests. We compare our CNN predictions to seven baseline algorithms: (1) "Constant Reflectance" (i.e., the shading channel is the luminance channel of the input image itself), (2) [Shen *et al*. 2011] [29], (3) Color Retinex [16], (4) [Garces *et al*. 2012] [13], (5) [Zhao *et al*. 2012] [35], (6) [Bell *et al*. 2014] [5], and (7) [Zhou *et al*. 2015] [37].

5.2. Precision-Recall

By running these baseline algorithms on our test image set, and sweeping the threshold τ, we can plot precision-recall (PR) curves for the smooth shading class predicted by the baselines (see the colored lines in Fig. 7).[2] Similarly, we can sweep a threshold σ for the smooth shading prob-

[2]Note that our test set has the same 2:1:1 label balance as the training set, and we only include non-smooth shading points exactly on boundaries in our test set, i.e., we do not perform label dilation. We do not evaluate on points that have no label in the dataset.

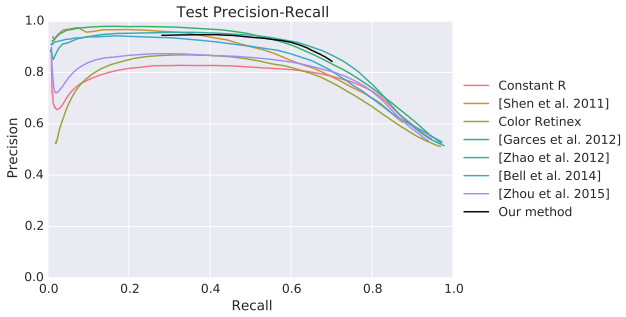

Figure 7: *Precision-recall for shading predictions*. We plot PR curves for the baselines and our algorithm. Our algorithm has competitive performance; see text for discussion.

Method	Precision @ 30%	50%	70%
Constant R	0.827	0.822	0.787
[Shen *et al*. 2011] [29]	0.958	0.899	0.784
Color Retinex [16]	0.867	0.850	0.755
[Garces *et al*. 2012] [13]	**0.977**	**0.949**	0.834
[Zhao *et al*. 2012] [35]	0.956	0.945	**0.868**
[Bell *et al*. 2014] [5]	0.936	0.902	0.802
[Zhou *et al*. 2015] [37]	0.873	0.858	0.802
Our method	0.946	0.938	0.845

Table 1: *Precision of shading predictions at varying levels of recall*. Precision @ recall levels of 30%, 50%, and 70% are shown for the seven baselines and our proposed method.

abilities predicted by our CNN, i.e. we say the shading is smooth at pixel p if the predicted smooth shading probability P_p is greater than σ. One way to interpret this evaluation is as a "smooth shading detector"—the algorithm must classify each pixel as smooth/not-smooth, and we evaluate precision and recall on this classification. The resulting PR curves are shown in Figure 7, and the performance at several recall values on these curves are shown in Table 1. The best methods are [Garces *et al*. 2012] and [Zhao *et al*. 2012], which use global optimization including clustering and long-range terms. In comparison, our method uses a single feed-forward pass and still achieves competitive performance.

Note that the PR curves are not all monotonic (see, for example, the left part of the PR curves), in part because the ground truth NS-ND labels contain a number of pixels that have very low contrast (i.e., have small image gradients and are thus very difficult to classify correctly). Some of these low-contrast pixels are due to effects like saturated pixels in the input imagery (e.g., a corner of a wall near a strong light source). On the other hand, these ground truth labels are based on the Kinect depth images whose quality is not limited by image contrast. These pixels are in the majority among the pixels classified as smooth shading at low recall regions of the curves where the threshold τ for the baselines is small. As τ, and consequently the recall increases, the proportion of low contrast pixels decreases and the precision increases for a short segment of the curve.

Finally, note that the curve for our method is truncated on both ends. This is due to the behavior of the CNN, where the prediction values it produces (after the final softmax layer) are often saturated at exactly 0 or 1. That is, there are a number of pixels where it reports maximal confidence in smooth or non-smooth shading, and this behavior manifests as truncation of the PR curve. As a result, our maximum recall is lower than that of other methods. This behavior suggests that the final softmax layer may be eliminating some useful dynamic range in the prediction scores.

It is interesting to note that [Bell *et al*. 2014] outperforms [Zhou *et al*. 2015] [37] on this shading prediction task, even though the latter is considered to have higher quality intrinsic image decompositions according to the IIW benchmark score [5]. We conjecture that since the IIW benchmark is based only on reflectance annotations, errors in the decomposed shading layer are not sufficiently penalized. Hence, our dataset offers another, complementary lens for evaluating the results of intrinsic image method. An area of future work is to use our annotations in conjunction with the IIW benchmark to devise a new, unified method for evaluating intrinsic image algorithms that considers both reflectance and shading annotations. We show example decompositions from both algorithms as supplemental material.

6. Application to Intrinsic Images

We now demonstrate a use of our smooth shading predictions as a prior for intrinsic image decomposition algorithms. To demonstrate the use of this prior, we modified the Retinex formulation of [35]. The original cost function they minimize to obtain a decomposition is the following:

$$f_l(\mathcal{S}) = \sum_{(p,q)\in\mathcal{N}} [(\mathcal{S}_p - \mathcal{S}_q)^2 + \omega_{(p,q)}(\mathcal{R}_p - \mathcal{R}_q)^2],$$

where $\mathcal{N}$ denotes the set of all neighboring pairs of pixels in the image, $\mathcal{S}_p$ and $\mathcal{R}_p$ are the shading and reflectance at pixel p, respectively. $\omega_{(p,q)}$ balances between shading and reflectance smoothness and is determined by the Retinex rule:

$$\omega_{(p,q)} = \begin{cases} 0 & \text{if } \left\| \hat{\mathcal{R}}_p - \hat{\mathcal{R}}_q \right\|_2 > t \\ 100 & \text{otherwise.} \end{cases}$$

where t is the Retinex threshold, $\hat{\mathcal{R}}_p$ and $\hat{\mathcal{R}}_q$ are the chromaticities of pixels p and q (see [35] for details). For simplicity, we do not use the non-local constraints of [35], only the Retinex constraint with $t = 0.02$. We incorporate our

(a) Input (b) Smooth $\mathcal{S}$ (S) (c) $\mathcal{R}$ w/o prior (d) $\mathcal{S}$ w/o prior (e) $\mathcal{R}$ with prior (f) $\mathcal{S}$ with prior

Figure 8: We show intrinsic image decompositions using our smooth shading prior. All images are selected from the test set. **(b)** shows a heatmap for our smooth shading (S) prediction. $\mathcal{R}$ and $\mathcal{S}$ denote reflectance and shading respectively. By using our smooth shading prior we can reduce artifacts in the decomposed shading layer, in particular removing surface texture effects which belong to the reflectance layer. For instance, our method removes more of the floor tile texture (top), or the texture of the blanket (bottom) from the shading layers.

smooth shading prior by modifying $\omega_{(p,q)}$:

$$\omega_{(p,q)} = \begin{cases} 0 & \text{if } \left\| \hat{\mathcal{R}}_p - \hat{\mathcal{R}}_q \right\|_2 > t \\ 100 \cdot [1 - (H_p + H_q)/2] & \text{otherwise.} \end{cases}$$

H_p and H_q are the smooth shading probabilities predicted by our model (Section 4) at pixels p and q. This formulation allows the decomposition algorithm to smoothly ignore the strong reflectance constancy constraint at regions where the predicted smooth shading heatmap has high probabilities.

In Figure 8, we show decompositions with and without our smooth shading prior. In some cases, we can see significant improvement in the decomposed shading layer. Specifically, our network is successful in detecting textured regions with smooth shading where most intrinsic image algorithms fail to remove the texture from the shading layer. The supplementary has more examples.

7. Conclusion and Future Work

We present Shading Annotations in the Wild, a new large-scale dataset of shading in real-world indoor scenes, created using a combination of crowdsourcing and automation. Using this dataset, we trained a CNN to achieve competitive performance against a number of baselines in per-pixel classification of shading effects in images. We also demonstrate a potential application of this network as a smooth shading prior for intrinsic image decomposition. We have made this dataset publicly available at http://opensurfaces.cs.cornell.edu/saw. Illumination is a key property of image formation; we hope that our dataset can enable other researchers to explore this property in a richer way that harnesses modern machine learning tools.

Our work suggests a few possibilities for future work. Evaluation of intrinsic image algorithms on our data suggests that our annotations may provide another way to rank these algorithms based on shading performance, complementary to the widely used WHDR metric [5] that only directly measures performance on reflectance. Using SAW with the reflectance annotations of IIW, we believe that new intrinsic image metrics can be established to advance the state of the art.

Our CNN for classifying pixels into different shading categories could be extended in a number of ways. For instance, we could jointly predict shading categories, shape, and materials (i.e., learn in a multi-task setting inside an approach like PixelNet [2]). Pushing this idea further, one could create a network that directly predicts an intrinsic image decomposition along with scene geometry, or further still, one that predicts a full 3D description of geometry and illumination trained using our data.

Acknowledgment This work was supported by the National Science Foundation (grants IIS-1617861, IIS-1011919, IIS-1161645, IIS-1149393), and by a Google Faculty Research Award.

References

[1] M. Allahbakhsh, B. Benatallah, A. Ignjatovic, H. R. Motahari-Nezhad, E. Bertino, and S. Dustdar. Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, 17(2):76–81, Mar. 2013. 5

[2] A. Bansal, B. Russell, and A. Gupta. Marr Revisited: 2D-3D model alignment via surface normal prediction. In *Proc. Computer Vision and Pattern Recognition*, 2016. 2, 5, 6, 8

[3] S. Beigpour, A. Kolb, and S. Kunz. A comprehensive multi-illuminant dataset for benchmarking of the intrinsic image algorithms. In *Proc. International Conference on Computer Vision*, December 2015. 2

[4] S. Beigpour, M. Serra, J. van de Weijer, R. Benavente, M. Vanrell, O. Penacchio, and D. Samaras. Intrinsic image evaluation on synthetic complex scenes. In *Int. Conf. on Image Processing*, 2013. 2

[5] S. Bell, K. Bala, and N. Snavely. Intrinsic images in the wild. *ACM Trans. on Graphics (SIGGRAPH)*, 33(4), 2014. 2, 3, 5, 6, 7, 8

[6] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database. *Proc. Computer Vision and Pattern Recognition*, 2015. 1

[7] N. Bonneel, B. Kovacs, S. Paris, and K. Bala. Intrinsic decompositions for image editing. *Computer Graphics Forum (Eurographics State of the Art Reports 2017)*, 36(2), 2017. 2

[8] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *Proc. European Conference on Computer Vision*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012. 2

[9] W. Chen, Z. Fu, D. Yang, and J. Deng. Single-image depth perception in the wild. In *Neural Information Processing Systems*, 2016. 2

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009. 1

[11] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proc. International Conference on Computer Vision*, pages 2650–2658, 2015. 2, 4

[12] M. Firman. RGBD Datasets: Past, Present and Future. In *CVPR Workshop on Large Scale 3D Data: Acquisition, Modelling and Analysis*, 2016. 2

[13] E. Garces, A. Munoz, J. Lopez-Moreno, and D. Gutierrez. Intrinsic images by clustering. *Computer Graphics Forum (Eurographics Symposium on Rendering)*, 31(4), 2012. 6, 7

[14] P. Gehler, C. Rother, M. Kiefel, L. Zhang, and B. Scholkopf. Recovering intrinsic images with a global sparsity prior on reflectance. In *Neural Information Processing Systems*, 2011. 2

[15] Y. Gingold, A. Shamir, and D. Cohen-Or. Micro perceptual human computation for visual tasks. *ACM Trans. on Graphics*, 31(5):119:1–119:12, Sept. 2012. 5

[16] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *Proc. International Conference on Computer Vision*, 2009. 2, 6, 7

[17] R. Guo, Q. Dai, and D. Hoiem. Single-image shadow detection and removal using paired regions. In *Proc. Computer Vision and Pattern Recognition*, 2011. 2

[18] T. Haber, C. Fuchs, P. Bekaer, H.-P. Seidel, M. Goesele, and H. P. Lensch. Relighting objects from image collections. In *Proc. Computer Vision and Pattern Recognition*, 2009. 2

[19] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 5

[20] K. Karsch, V. Hedau, D. Forsyth, and D. Hoiem. Rendering synthetic objects into legacy photographs. In *ACM Trans. on Graphics*, volume 30, page 157. ACM, 2011. 2

[21] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Estimating natural illumination from a single outdoor image. In *Proc. International Conference on Computer Vision*, pages 183–190. IEEE, 2009. 2

[22] J.-F. Lalonde, A. A. Efros, and S. G. Narasimhan. Detecting ground shadows in outdoor consumer photographs. In *Proc. European Conference on Computer Vision*, 2010. 2

[23] J.-F. Lalonde and I. Matthews. Lighting estimation in outdoor image collections. In *International Conference on 3D Vision*, 2014. 2

[24] E. H. Land and J. J. McCann. Lightness and retinex theory. *J. Opt. Soc. Am.*, 61(1), 1971. 1

[25] B. Li, C. Shen, Y. Dai, A. van den Hengel, and M. He. Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In *Proc. Computer Vision and Pattern Recognition*, 2015. 2

[26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Proc. European Conference on Computer Vision*, pages 740–755. Springer, 2014. 1

[27] Y. Ostrovsky, P. Cavanagh, and P. Sinha. Perceiving illumination inconsistencies in scenes. *Perception*, 34(11), 2005. 3

[28] P. Ren, Y. Dong, S. Lin, X. Tong, and B. Guo. Image based relighting using neural networks. 2015. 2

[29] J. Shen, X. Yang, Y. Jia, and X. Li. Intrinsic images using optimization. In *Proc. Computer Vision and Pattern Recognition*, 2011. 6, 7

[30] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *Proc. European Conference on Computer Vision*, 2012. 2, 3, 4

[31] S. Song, S. P. Lichtenberg, and J. Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *Proc. Computer Vision and Pattern Recognition*, 2015. 2

[32] P. Welinder, S. Branson, S. Belongie, and P. Perona. The multidimensional wisdom of crowds. In *Neural Information Processing Systems*, 2010. 4, 5

[33] Y. Yu, P. Debevec, J. Malik, and T. Hawkins. Inverse global illumination: recovering reflectance models of real scenes from photographs. *ACM Trans. on Graphics (SIGGRAPH)*, 1999. 2

[34] B. Zeisl, M. Pollefeys, et al. Discriminatively trained dense surface normal estimation. In *Proc. European Conference on Computer Vision*, pages 468–484. Springer, 2014. 2, 4

[35] Q. Zhao, P. Tan, Q. Dai, L. Shen, E. Wu, and S. Lin. A closed-form solution to retinex with nonlocal texture constraints. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34(7), 2012. 6, 7

[36] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Neural Information Processing Systems*, 2014. 1

[37] T. Zhou, P. Krahenbuhl, and A. A. Efros. Learning data-driven reflectance priors for intrinsic image decomposition. In *Proc. International Conference on Computer Vision*, pages 3469–3477, 2015. 2, 6, 7

[38] J. Zhu, K. G. Samuel, S. Z. Masood, and M. F. Tappen. Learning to recognize shadows in monochromatic natural images. In *Proc. Computer Vision and Pattern Recognition*, pages 223–230. IEEE, 2010. 2