International Journal of
Intelligent Systems and
Applications in Engineering

ISSN:2147-6799 www.atscience.org/IJISAE

Advanced Technology and Science

Original Research Paper

Structure-Texture Decomposition of RGB-D Images

Aykut Erdem*[1]

Accepted 24th October 2016

Abstract: In this paper, we study the problem of separating texture from structure in RGB-D images. Our structure preserving image smoothing operator is based on the *region covariance smoothing* (RCS) method in [16] that we present a number of modifications to this framework to make it depth-aware and increase its effectiveness. In particular, we propose to incorporate three geometric depth features, namely height above ground, angle with gravity and horizontal disparity to the pool of image features used in that study. We also suggest to use a new kernel function based on KL-divergence between the distributions of extracted features. We demonstrate our approach on challenges images from NYU-Depth v2 Dataset [24], achieving more accurate decompositions than the state-of-the-art approaches which do not utilize any depth information.

Keywords: *RGB-D images, structure-preserving smoothing, image decomposition, region covariances*

1. Introduction

Natural images typically contain several textured regions with different fine and coarse scale details. These regions that have large oscillatory structure, are quite hard to separate for classical image filters such as the Gaussian or the median filters. Figure 1 shows some texture examples, demonstrating the richness and the variability of the texture components. In recent years, the community has witnessed the surge of development of new filters specifically designed to decompose an image into its structure and texture components, examples of which include *weighted least square* (WLS) [10], *guided filter* (GF) [12], *L_0-smoothing* [32], *relative total variation* (RTV) [34], *region covariance smoothing* (RCS) [16], *bilateral texture filtering* (BTF) [5] and *rolling guidance filter* (RGF) [39]. These, the so-called structure-preserving filters, aim at filtering out textures or low-contrast details while retaining prominent image structures and sharp edges. These filters can serve as a useful preprocessing tool to improve the performances of many computer vision and computational photography applications like tone mapping, detail enhancement, colorization, intrinsic image decomposition and scene understanding [1,5,10, 14,16,27,33-35].

Meanwhile, with the recent availability of affordable and reliable 3D acquisition devices like Microsoft Kinect, Apple PrimeSense, Google Project Tango and Intel RealSense, there has been a renewed interest in incorporating depth information into the pipelines for many different computer vision tasks, *e.g.* semantic image segmentation [6,11], object detection [11,26], object tracking [4,25] and visual saliency detection [7,21]. The cues derived from a depth map could help the algorithms to disambiguate the visual interpretation of the scene, and hence significantly improve their accuracy. For image smoothing, the depth map can supply additional information about the scene layout, which can be exploited to better distinguish the object boundaries. For example, Figure 2 shows an image of a bathroom scene where the boundary of the white towel placed on the sink is visually indistinguishable from the background whereas this object boundary is clearly visible in the corresponding depth map.

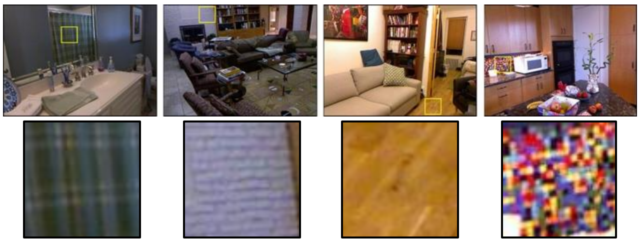


Figure 1. Textured region examples which appear in some indoor images (from NYU-Depth v2 Dataset [23]).

This study explores, for the first time, how much depth information helps decomposing an image into its structure and texture components. Here, building on the framework in [16], we propose a novel depth-aware structure-preserving image smoothing model which utilizes additional depth features complementary to the image features. Moreover, we suggest to use a more effective kernel function that is based on KL-divergence between the distributions of extracted features. We note that our aim here is still separating the texture from the structure of an RGB image, but while attempting to do so, we additionally make use of depth cues. We assume that the depth image does not have an ultra-high resolution so that the details of the textured regions are not visible hence it is sufficient enough to work on smoothing the RGB image alone. Although our ideas can be applied to decomposition of depth images, it is beyond the scope of this paper.

The rest of the paper is organized as follows: In Section 2, we review the previous studies on structure-preserving image smoothing, putting more emphasis on more recent work. In Section 3, we describe our depth-aware image decomposition approach in detail. In Section 4, we present and discuss our experimental results on images from NYU-Depth v2 Dataset [24]. Finally, in Section 5, we conclude the paper with a brief summary and some remarks on directions for future research.

[1] *Hacettepe University, Department of Computer Engineering, Ankara, Turkey.*
* *Corresponding Author: Email: aykut@cs.hacettepe.edu.tr*

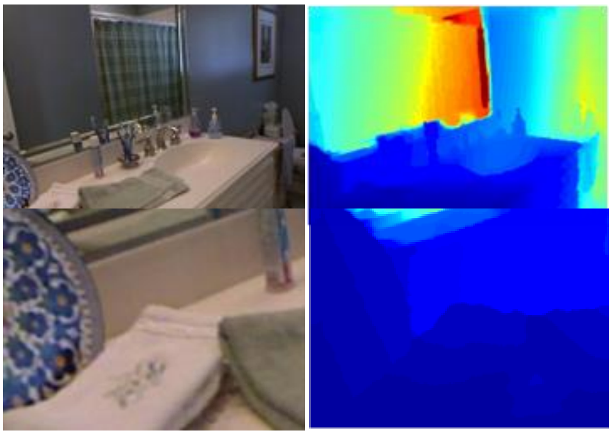


Figure 2. An example RGB and depth image pair from NYU-Depth v2 Dataset [24]. The boundary of the white towel on the bathroom sink is not visible in the RGB image, but is apparent in the depth map (Best viewed in color).

2. Related Work

In this section, we provide a brief review of traditional and more recent structure-preserving smoothing filters. All these filters try to decompose a given image into its structure and detail components by applying smoothing while simultaneously preserving image edges or details. In terms of their underlying computational frameworks, the existing filtering approaches can be mainly grouped into four family of studies, as *mode and median filters* [17,18,30,31,38], *optimization-based approaches* [10,22,23,27, 32,34], *weighted averaging methods* [5,9,12,16,20,28,36,39], and *learning-based approaches* [33,35,37]. Below we summarize these approaches and compare and contrast their properties.

2.1. Mode and Median Filters

As compared to popular *mean filtering* or *Gaussian filters*, *mode* and *median filters* can provide much more satisfactory results in removing high-contrast details. They simply replace the value of a pixel with the mode [17,30] or the median [17,31] computed within a local neighborhood rather than taking the average, or alternatively some median filters additionally employ weighted schemes [18,38]. All these filters are excellent at removing salt-and-pepper noise but they are computationally expensive and they are not very successful in eliminating the oscillatory parts of images that generally correspond to the textured regions.

2.2. Optimization-Based Approaches

The filters in this category of works pose the smoothing operation as an optimization problem, and they differ from each other in their optimization frameworks. Examples include *anisotropic diffusion* model [22], *total variation* (TV) model [23], *weighted least squares* (WLS) [10], *envelope extraction* [27], L_0 *gradient minimization model* [32] and *relative total variation* (RTV) model [34].

The *anisotropic diffusion model* of Perona and Malik [22] utilizes a PDE-based formulation where spatially-varying diffusivity values are calculated for each pixel based on local image gradients, which are then used to guide the smoothing process to preserve edges and thus important image structures. However, this approach is very sensitive to noise and it can not distinguish texture from edges.

Another well-known optimization model is the *total variation* (TV) model by Rudin *et al.* [23] which explicitly penalizes large gradient magnitudes via additional L_1-norm based regularization term. There exist many different extensions to improve the basic formulation which propose to use other regularization and data fidelity terms [2,19]. This formulation can provide fairly good smoothing results but its drawback is that the corresponding smoothing process could influence the contrast. Recently, Xu *et al.* [34] proposed a global optimization scheme with a *relative total variation* (RTV) regularization scheme, which specifically addresses separation of structure from texture. However, RTV measure could fail if the scale and the shape of edges are similar to the nearby texture.

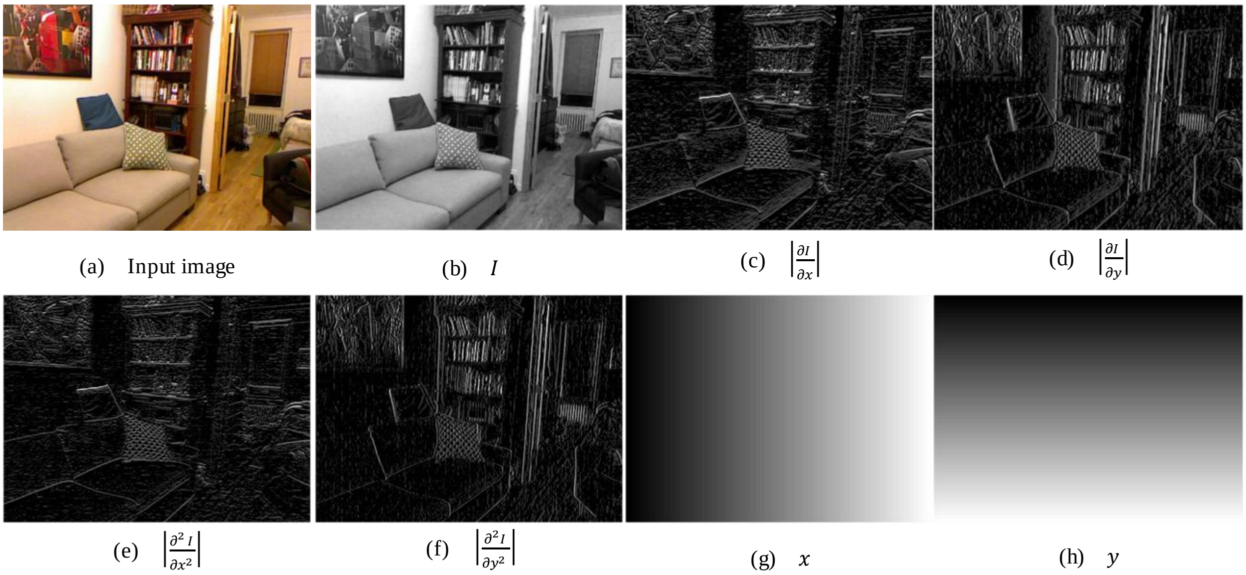
Other examples to optimization based formulations are the WLS method by Farbman *et al.* [10], which uses *weighted least squares*, and the model of Xu *et al.* [32], which is based on the optimization on the L_0 norm of image gradients. These two approaches are very successful to suppress low-contrast details but their performance deteriorates for high-contrast details and textures as their formulations are directly based on image gradients. In addition, Subr *et al.* [27] performs an extrema and then performs texture smoothing by taking the average of the extremal envelopes.

2.3. Weighted Averaging Methods

Another popular kind of structure-preserving smoothing approaches is the weighted averaging methods, one representative of which is the widely-known *bilateral filter* (BF) [28]. Since its computational complexity is very high, many accelerated versions exist [9,20,36]. The basic formulation suffers from the so-called halo effect, which is caused by oversmoothing of high-contrast image edges.

An important extension to the BF is the *guided filter* (GF) [12] which considers a guidance image, the same input image or a different one, which serves as the base for a locally linear transform of the input image. As compared to BF, the guided image filter is much faster and better preserves the gradients around the edges. However, it is still susceptible to the halo artifacts. Recently, Zhang *et al.* propose an iterative scheme called the *rolling guidance filter* (RGF) [39], which is quite fast and gives superior structure-texture decomposition results. Likewise, an idea similar to GF is explored in [5], in which the authors introduce a modified RTV measure which is used to compute a guidance image to guide the separation of texture from the prominent image structure.

An alternative patch-based local filtering approach, *region covariance smoothing* (RCS), is presented in [16], in which the authors propose to use first and second order feature statistics to define a novel similarity measure based on region covariance descriptor [29]. Despite being very simple and easy to implement, the method is quite effective in smoothing out textured regions, while preserving important image structures. Since our depth-aware smoothing approach is based on this model, more details about this smoothing filter will be given in Section 3.



(a) Input image

(b) I

(c) $\left|\frac{\partial I}{\partial x}\right|$

(d) $\left|\frac{\partial I}{\partial y}\right|$

(e) $\left|\frac{\partial^2 I}{\partial x^2}\right|$

(f) $\left|\frac{\partial^2 I}{\partial y^2}\right|$

(g) x

(h) y

Figure 3. The image features used in the RCS method [16], namely intensity, orientation and pixel coordinates.

2.4. Learning-Based Approaches

The final group of approaches utilize learning-based strategies to train edge/structure-preserving filters directly from data. The first example of this kind of works is the *SVM-based filter* in [37] which learns a function to map feature vector representation of a pixel consisting of exponentiation of the pixel value and their Gaussian filtered responses to the desired output. Similarly, Xu *et al.* [33] propose *deep edge filters*, which uses deep convolutional neural networks to learn filters in the gradient domain from a large set of natural image patches and their smoothed versions as the training data. Lastly, Yang [35] introduce the notion of *semantic filters*, which employs confidence map of a learning-based edge detection model trained on human labelled data to guide the smoothing process.

3. The Approach

The problem of decomposing an image I into its structure (S) and texture (T) components is usually defined with the following composition equation:

$$I = S + T \tag{1}$$

where S should contain on the main structural parts where the textural part T should reflect on the detail components, devoid of any noticeable structural information.

In the following, we first introduce the image and depth features used in our smoothing framework in Section 3.1 and Section 3.2, respectively. In Section 3.3, we then describe how we modify the *region covariance smoothing* (RCS) method of Karacan *et al.* [16] to make it additionally consider depth information about the scene to improve its decomposition quality.

3.1. Image Features

In our implementation, we use the same set of image features utilized in [16], namely intensity, orientation and pixel coordinates. Hence, an image pixel is represented with the following 7-dimensional feature vector:

$$F_{image}(x,y) = \left[I(x,y) \quad \left|\frac{\partial I}{\partial x}\right| \quad \left|\frac{\partial I}{\partial y}\right| \quad \left|\frac{\partial^2 I}{\partial x^2}\right| \quad \left|\frac{\partial^2 I}{\partial y^2}\right| \quad x \quad y \right] \tag{2}$$

with I denoting the pixel intensity, $\left|\frac{\partial I}{\partial x}\right|, \left|\frac{\partial I}{\partial y}\right|, \left|\frac{\partial^2 I}{\partial x^2}\right|, \left|\frac{\partial^2 I}{\partial y^2}\right|$ expressing the first and second-order derivatives of the image intensities, estimated with the filters [-1 0 1] and [-1 2 -1] in horizontal and vertical directions, and (x, y) corresponding to the pixel location. For a sample image, these features are presented in Figure 3.

3.2. Depth Features

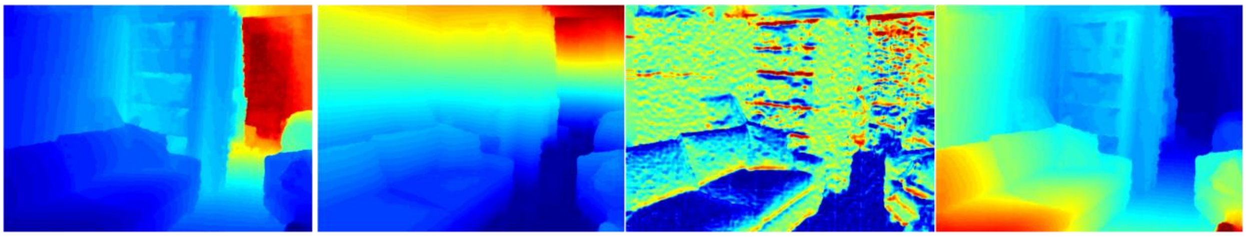
The novelty of our work lies in the utilization of extra depth information for structure-texture decomposition. For the computer vision problems which consider RGB-D images as the input, the most straightforward way to consider depth information is to treat the depth image as a standard grayscale image, *e.g.* in tracking [25].

Our decomposition approach, however, uses a much richer and geometrically meaningful set of depth features, namely *horizontal disparity* (disparity), *angle with gravity* (angle), and *height above ground* (height), which are proven to be effective for object detection and semantic segmentation in a prior work [11]. This, in the end, gives us a 3-dimentional representation to denote the pixelwise depth of a pixel, as follows:

$$F_{depth}(x,y) = [\text{disparity}(x,y) \quad \text{angle}(x,y) \quad \text{height}(x,y)] \tag{3}$$

In Figure 4, we visualize these depth features, which are extracted from the depth map of the image shown in Figure 3(a). As will be discussed in Section 4, incorporating this encoding into the standard image filtering pipeline of [16] greatly improves the quality of the decompositions.

3.3. Depth-Aware Structure-Texture Decomposition



Depth image *disparity* *angle* *height*

tions used to estimate

Figure 4. The depth features used in our depth-aware smoothing approach, namely horizontal disparity, angle with gravity and height above ground.

Our depth-aware image decomposition approach extends the structure-preserving smoothing approach of Karacan *et al.* [16] with additional depth features and a novel KL-divergence based kernel function. Below, we summarize the the method in [16] and describe how we incorporate the complementary depth features into this framework.

The method of Karacan *et al.* [16] is a novel variant of the *non-local means filter* [3], which employs *region covariance descriptor* [29] and two different adaptive kernel functions to define patch similarity. The model represents each pixel by the first and second order statistics of visual features, which are extracted from a patch around the pixel. More formally, let $F(x,y)$ represent the $W \times H \times D$ dimensional feature image extracted from a given image I. Then, a region R inside F can be expressed by a $d \times d$ covariance matrix $\mathbf{C}_R$, computed as:

$$\mathbf{C}_R = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{z}_i - \mu)(\mathbf{z}_i - \mu)^T \qquad (4)$$

where $\mathbf{z}_{i=1\,\ldots\,n}$ denotes the d-dimensional feature vectors inside R and μ is the mean of these feature vectors.

In our study, we propose to extend 7-dimensional simple image features (F_{image}) given in Section 3.1 with the depth features (F_{depth}) proposed in Section 3.2. Hence, we obtain a 10-dimensional feature vector to represent each pixel of the given image:

$$F(x,y) = [F_{image}(x,y) \quad F_{depth}(x,y)] \qquad (5)$$

The region covariance descriptor only encodes the second-order statistical relationships among the features and moreover, measuring the similarity between two covariance matrices is computationally expensive since they lie on a Riemannian manifold. To transform covariance matrices into Euclidean space, Hong *et al.* [13] exploit the property that every symmetric positive semi-definite matrix (like covariance matrices) has a unique Cholesky decomposition and propose to represent a $d \times d$ covariance matrix $\mathbf{C}$ with a set of points $\mathcal{S} = \{\mathbf{s_i}\}$, which is computed as follows:

$$\mathbf{s_i} = \begin{cases} \sqrt{2d}\mathbf{L}_i & \text{if } 1 \le i \le d \\ -\sqrt{2d}\mathbf{L}_i & \text{if } d+1 \le i \le 2d \end{cases} \qquad (6)$$

where $\mathbf{L}_i$ denotes the ith column of the lower triangular matrix $\mathbf{L}$ obtained with the Cholesky decomposition $\mathbf{C} = \mathbf{L}\mathbf{L}^T$.

The structure component of a single pixel $\mathbf{p}$ can be computed with the following simple equation:

$$S(\mathbf{p}) = \frac{1}{Z_\mathbf{p}}\sum_{\mathbf{q}\in N(\mathbf{p},r)} w_{\mathbf{pq}} I(\mathbf{q}) \qquad (7)$$

where $N(\mathbf{p},r)$ stands for the pixel neighborhood of size $(2r+1)\times(2r+1)$ centered at $\mathbf{p}$, $w_{\mathbf{pq}}$ denotes the similarity between two pixels $\mathbf{p}$ and $\mathbf{q}$, which is measured according to the region covariance based kernel functions based on the $k \times k$ patches centered around these pixels, and $Z_\mathbf{p} = \sum_\mathbf{q} w_{\mathbf{pq}}$ is just a normalization factor.

The effectiveness of RCS method depends on the kernel functions used to estimate patch similarity. In [16], the authors propose two models based on two different kernel functions, which are explained below.

The first model (Model 1) is based on the Euclidean encoding of covariance matrices in [13] where the patch similarity $w_{\mathbf{pq}}$ is defined as:

$$w_{\mathbf{pq}} \propto \exp\left(-\frac{\left\|\varphi(\mathbf{C_p})-\varphi(\mathbf{C_q})\right\|^2}{2\sigma^2}\right) \qquad (8)$$

with $\varphi(\mathbf{C}) = (\mu, \mathbf{s}_1, \ldots, \mathbf{s}_\mathrm{d}, \mathbf{s}_{\mathrm{d}+1}, \ldots, \mathbf{s}_{2\mathrm{d}})^T$ being a feature mapping function which concatenates the first-order statistics of the features (μ) to the vectorial form of the covariance $\mathbf{C}$, and σ being a spatial parameter, controlling the level of smoothing.

The second model (Model 2), on the other hand, uses a different kernel function, which is based on an approximation of the Bhattacharyya distance between two multivariate normal distributions, defined as:

$$w_{\mathbf{pq}} \propto \exp\left(-\frac{(\mu_\mathbf{p}-\mu_\mathbf{q})\mathbf{C}^{-1}(\mu_\mathbf{p}-\mu_\mathbf{q})^T}{2\sigma^2}\right) \qquad (9)$$

where $\mathbf{C} = \mathbf{C_p} + \mathbf{C_q}$ with $\mathbf{C_p}$ and $\mathbf{C_q}$ being the covariance matrices of the features extracted from the patches centered at pixels $\mathbf{p}$ and $\mathbf{q}$, and $\boldsymbol{\mu}_\mathbf{p}$ anf $\boldsymbol{\mu}_\mathbf{q}$ are the means of these features, respectively.

In addition to these two kernel functions, in this study, we also adopt the KL-divergence based distance measure in [15], which is proposed for sampling based image matting, as a third adaptive kernel function. We refer to this new model as Model 3. Mathematically, the KL-divergence between two multivariate normal distributions is computed as:

$$D_{KL}(\mathbf{p},\mathbf{q}) = \frac{1}{2}\left(\mathrm{tr}(\mathbf{C_q}^{-1}\mathbf{C_p}) + \ln\left(\frac{\det(\mathbf{C_q})}{\det(\mathbf{C_p})}\right) + (\mu_\mathbf{q}-\mu_\mathbf{p})\mathbf{C_q}^{-1}(\mu_\mathbf{q}-\mu_\mathbf{p})^T - d\right) \qquad (10)$$

Then, we define the similarity between two image patches centered at pixels $\mathbf{p}$ and $\mathbf{q}$, as follows:

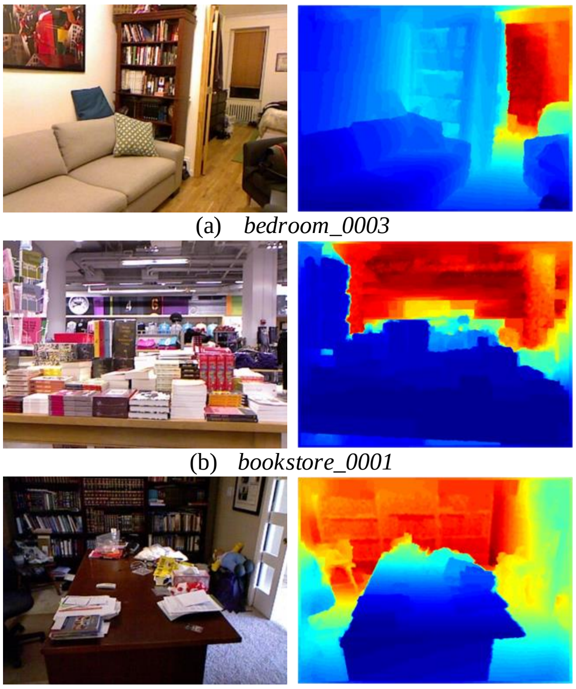
$$w_{\mathbf{pq}} \propto \exp\left(-\frac{D_{KL}(\mathbf{p},\mathbf{q})+D_{KL}(\mathbf{q},\mathbf{p})}{2\sigma^2}\right) \qquad (11)$$

4. Experimental Results

In this section, we present our experimental evaluation of the proposed depth-aware smoothing method. To qualitatively evaluate our approach, we compare its results on a set of test images to the results of three state-of-the-art structure preserving smoothing methods, namely the original *region covariance smoothing* (RCS) method of Karacan et al. [16], *relative total variation* (RTV) model [34], and *rolling guidance filter* (RGF) [39], whose implementations are publicly available on the web. To highlight the effectiveness of our approach, we also demonstrate its use in detecting structural edges as an application.

4.1. Test images

We evaluate our proposed structure-texture decomposition approach on the three challenging images shown in Figure 5, which are all from NYU-Depth v2 Dataset [24] where the depth maps are readily available. These images are carefully chosen for our task such that they all contain highly textured regions. The *bedroom_0003* image shows a bedroom with a diamond print pillow and wooden floor. The *bookstore_0001* image shows an image of a bookstore where many colorful books are stacked on a wooden desk. The *home_office_0011* image is a cluttered indoor scene containing a desk in front of a bookcase that is places on a fluffy rug.


(a) *bedroom_0003*

(b) *bookstore_0001*

(c) *home_office_0011*

Figure 5. The test images from NYU-Depth v2 Dataset [24].

4.2. Comparison

In our experiments, we test our approach against three recently proposed structure-preserving smoothing approaches, *i.e. region covariance smoothing* (RCS) [16], *relative total variation* (RTV) [34], and *rolling guidance filter* (RGF) [39]. In addition, we also incorporate our proposed KL-divergence based kernel function to RCS, which we refer to as RCS (Model 3). The nature of the problem requires us to qualitatively compare the results such that a better smoothing model should preserve structure while only smoothing out the fine details and texture, and the texture component should not contain any information about the structure.

In Figure 6-8, we respectively show the structure-texture decomposition results of the test images introduced in Figure 5, together with the close-up views of some image regions that contain different textures. Here, we note that we fine tuned the parameters of each tested method. As for the RTV [34] and RGF [39] models, it seems that they both oversmooth some prominent structures as some structures are clearly visible in the corresponding texture components. Compared to these two methods, the RCS models (Model 1 and Model 2), and the variant with our proposed kernel function (Model 3) provide much better structure-texture decomposition results and moreover they preserve smoothly varying shading information, as discussed in [16]. However, they slightly blur the edges during the smoothing process. As can be seen from the provided close-up views, the proposed depth-aware models (Model 1 and Model 3) always improve the quality of the corresponding base models, such as better capturing the shading on the sofa (Figure 6), the structure of the pipes on the ceiling (Figure 7), and the edges of the papers on the desk (Figure 8). Model 2, however, produces ineffective smoothing results. We suspect that the corresponding kernel function could be giving more weight to the depth features. Overall, our results suggest that considering depth in the filtering process is beneficial to obtain better structure-texture decompositions.

4.3. Detecting Structural Edges

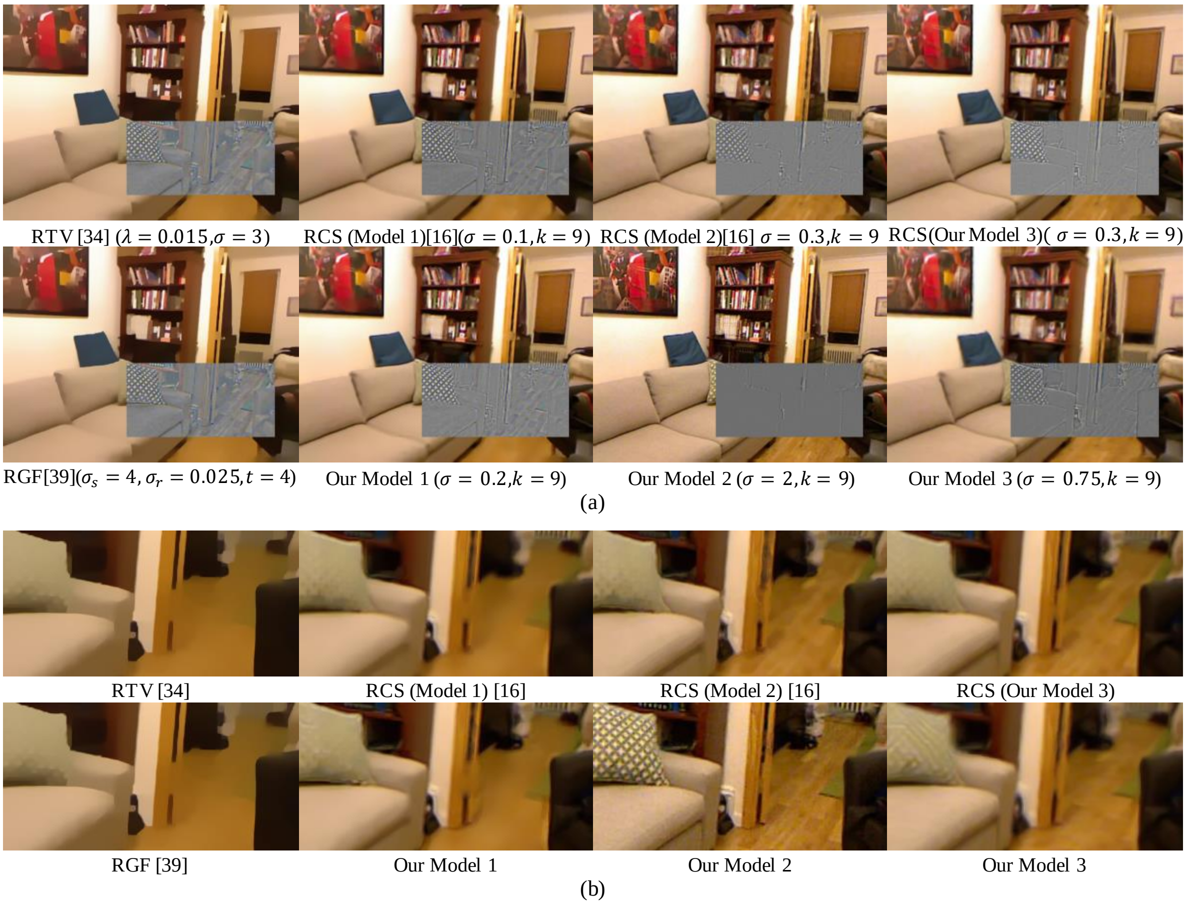
For the edge detection algorithms, it is really hard to distinguish edges from oscillations in the images. A direct application of structure-preserving smoothing methods is to improve the performance of this classical image processing task, where edge detection is performed not on the originally given input image but its smoothed version. In Figure 9, we illustrate this use, in which we apply the recently proposed edge detection algorithm developed by Dollar and Zitnick [8] on one of our test images and its structure components obtained with RCS (Model 1) [16] and its depth-aware version proposed in this study. As can be seen from the figure, the additional depth information improves the detection of the structural edges, especially the image edges associated with the sofa in the scene.

5. Conclusion

We have developed a novel depth-aware image smoothing approach to decompose an RGB-D image into its structural and textural parts. The proposed methods extends the *region covariance smoothing* method of Karacan *et al.* [16] with additional geometrical features, which are extracted from the depth map of the given image. Moreover, we suggest to use a KL-divergence based adaptive kernel function to better approximate similarity between two image patches. This study has demonstrated, for the first time, that that additionally considering the depth information about the scene provides much better decomposition results, compared to the results of state-of-the-art structure-preserving smoothing methods that do not take into account any information about the depth. We have also shown a straightforward application of our approach on better detecting the structural edges. Possible areas of future research would be to investigate other uses of this method, such as improving the intrinsic image decomposition algorithm proposed in [14].

Acknowledgements

This work was supported by a grant from The Scientific and Technological Research Council of Turkey (TUBITAK) – Career Development Award 112E146.



RTV [34] ($\lambda = 0.015, \sigma = 3$) RCS (Model 1)[16]($\sigma = 0.1, k = 9$) RCS (Model 2)[16] $\sigma = 0.3, k = 9$ RCS(Our Model 3)($\sigma = 0.3, k = 9$)

RGF[39]($\sigma_s = 4, \sigma_r = 0.025, t = 4$) Our Model 1 ($\sigma = 0.2, k = 9$) Our Model 2 ($\sigma = 2, k = 9$) Our Model 3 ($\sigma = 0.75, k = 9$)

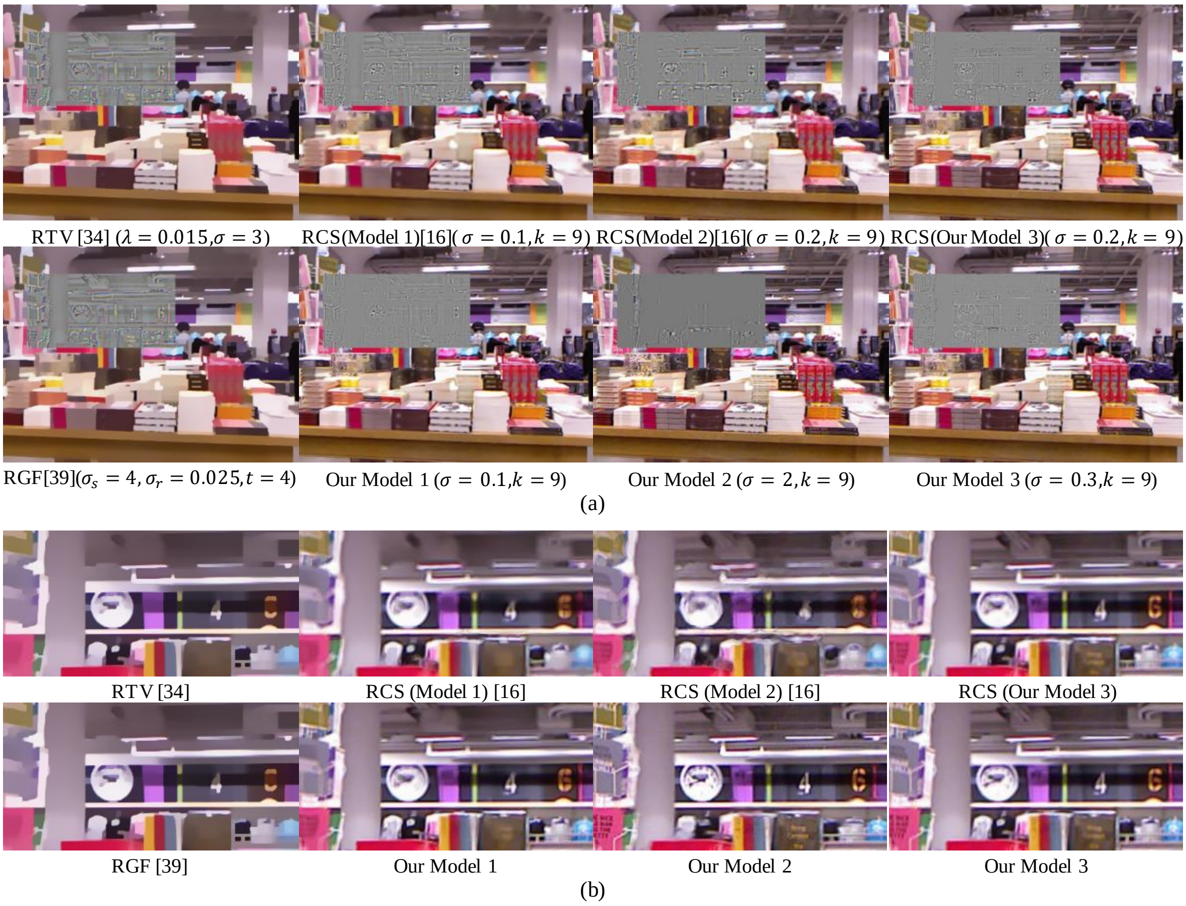
(a)

RTV [34] RCS (Model 1) [16] RCS (Model 2) [16] RCS (Our Model 3)

RGF [39] Our Model 1 Our Model 2 Our Model 3

(b)

Figure 6. (a) Structure-texture decomposition results on the *bedroom_0003* image. (b) Some close-up views of the extracted structures.



RTV [34] ($\lambda = 0.015, \sigma = 3$) RCS(Model 1)[16]($\sigma = 0.1, k = 9$) RCS(Model 2)[16]($\sigma = 0.2, k = 9$) RCS(Our Model 3)($\sigma = 0.2, k = 9$)

RGF[39]($\sigma_s = 4, \sigma_r = 0.025, t = 4$) Our Model 1 ($\sigma = 0.1, k = 9$) Our Model 2 ($\sigma = 2, k = 9$) Our Model 3 ($\sigma = 0.3, k = 9$)

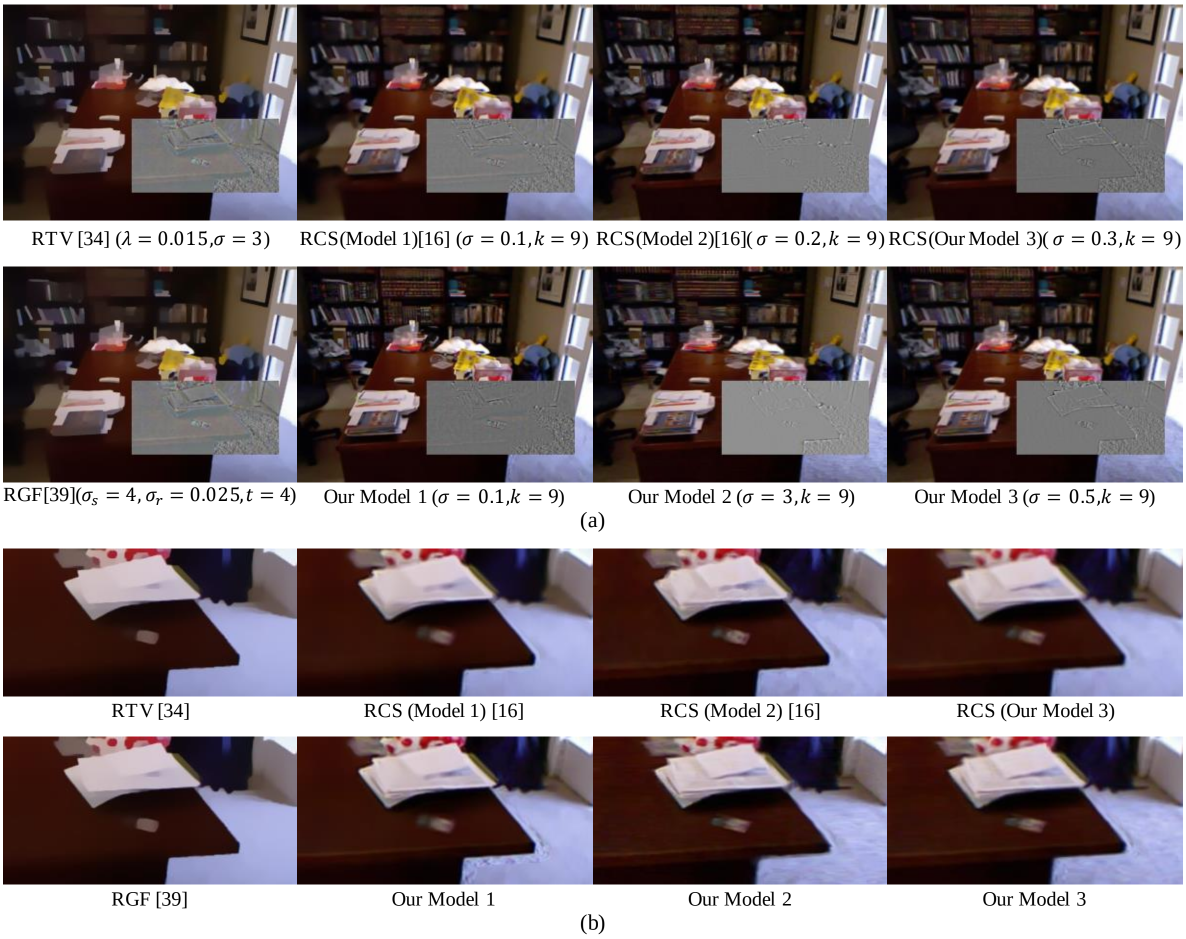
(a)

RTV [34] RCS (Model 1) [16] RCS (Model 2) [16] RCS (Our Model 3)

RGF [39] Our Model 1 Our Model 2 Our Model 3

(b)

Figure 7. (a) Structure-texture decomposition results on the *bookstore_0001* image. (b) Some close-up views of the extracted structures.



RTV [34] ($\lambda = 0.015, \sigma = 3$) RCS(Model 1)[16] ($\sigma = 0.1, k = 9$) RCS(Model 2)[16]($\sigma = 0.2, k = 9$) RCS(Our Model 3)($\sigma = 0.3, k = 9$)

RGF[39]($\sigma_s = 4, \sigma_r = 0.025, t = 4$) Our Model 1 ($\sigma = 0.1, k = 9$) Our Model 2 ($\sigma = 3, k = 9$) Our Model 3 ($\sigma = 0.5, k = 9$)

(a)

RTV [34] RCS (Model 1) [16] RCS (Model 2) [16] RCS (Our Model 3)

RGF [39] Our Model 1 Our Model 2 Our Model 3

(b)

Figure 8. (a) Structure-texture decomposition results on the *home_office_0011* image. (b) Some close-up views of the extracted structures.

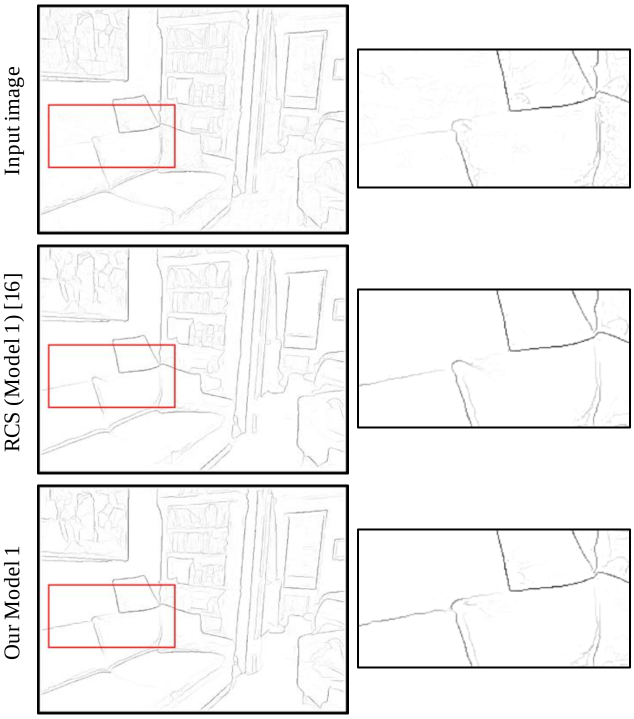


Figure 9. Edge detection results. Our depth-aware model better captures the structure, thus improving the edge detection process.

References

[1] B. Arbelot, R. Vergne, T. Hurtut, and J. Thollot, "Automatic texture guided color transfer and colorization", in *Proc. Expresive'16*, 2016

[2] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher, "Structure-texture image decomposition–modeling, algorithms, and parameter selection", *International Journal of Computer Vision*, vol. 67, issue 1, Apr. 2006, pp. 111–136.

[3] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising", in *Proc. CVPR'05*, vol. 2, 2005, pp. 60–65.

[4] M. Camplani, S. Hannuna, M. Mirmehdi, D. Damen, A. Paiement, L. Tao, and T. Burghardt, "Real-time RGB-D tracking with depth scaling kernelised correlation filters and occlusion handling", in *Proc. BMVC'15*, 2015.

[5] H. Cho, Hyunjoon Lee, H. Kang, and S. Lee, "Bilateral texture filtering", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH 2014*, vol. 33, issue 4, July 2014.

[6] Z. Deng, S. Todorovic, and L.J. Latecki, "Semantic Segmentation of RGBD Images with Mutex Constraints", in *Proc. ICCV'15*, 2015.

[7] K. Desingh, K.M. Krishna, D. Rajan, and C. Jawahar, "Depth really matters: Improving visual salient region detection with depth", in *Proc. BMVC'13*, 2013.

[8] P. Dollar, and C. L. Zitnick, "Structured Forests for Fast Edge Detection", in *Proc. ICCV'13*, 2013.

[9] F. Durand, and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images". *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH 2001*, vol. *21*, issue 3, Jul. 2002, 257–266.

[10] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH 2008*, vol. 27, issue 3, Aug. 2008.

[11] S. Gupta, R. Girshick, P. Arbeláez, J. Malik, "Learning rich

features from RGB-D Images for object detection and segmentation", in *Proc. ECCV'14,* 2014.

[12] K. He, J. Sun, and X. Tang, "Guided image filtering", in *Proc. ECCV'10,* 2010.

[13] X. Hong, H. Chang, S. Shan, X. Chen, and W. Gao, "Sigma set: A small second order statistical region descriptor", in *Proc. CVPR'09,* 2009, pp. 1802–1809.

[14] J. Jeon, S. Cho, X. Tong, and S. Lee, "Intrinsic image decomposition using structure-texture separation and surface normals", in *Proc. ECCV'14*, 2014.

[15] L. Karacan, A. Erdem, and E. Erdem, "Image Matting with KL-Divergence Based Sparse Sampling", in *Proc. ICCV'15*, 2015, pp. 424-432.

[16] L. Karacan, E. Erdem, and A. Erdem, "Structure-preserving image smoothing via region covariances", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH Asia 2013*, vol. 32, issue 6, Nov. 2013.

[17] M. Kass, and J. Solomon, "Smoothed local histogram filters", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH 2010*, vol. 29, issue 4, July 2010.

[18] Z. Ma, K. He, Y. Wei, J, Sun, and E. Wu, "Constant time weighted median filtering for stereo matching and beyond", in *Proc. ICCV'13,* 2013.

[19] Y. Meyer, *Oscillating patterns in image processing and nonlinear evolution equations: the fifteenth Dean Jacqueline B. Lewis memorial lectures*, vol. 22. American Mathematical Society, 2001.

[20] S. Paris and F. Durand, "A Fast Approximation of the Bilateral Filter using a Signal Processing Approach", in *Proc. ECCV'06,* 2006.)

[21] H. Peng, B. Li, W. Xiong, W. Hu, and R. Ji, "RGBD salient object detection: A benchmark and algorithms", in *Proc. ECCV'14*, 2014, pp. 92–109.

[22] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 12, Jul. 1990, pp. 629–639.

[23] L. Rudin, S. Osher and E. Fatemi, "Nonlinear total variation based noise removal algorithms", *Physica D: Nonlinear Phenomena*. vol. 60, Nov. 1992, pp. 259–268.

[24] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor Segmentation and Support Inference from RGBD Images", in *Proc. ECCV'12*, 2012.

[25] S. Song and J. Xiao, "Tracking revisited using RGBD camera: Unified benchmark and baselines", in *Proc. ICCV'13*, 2013.

[26] S. Song and J. Xiao, "Deep sliding shapes for amodal 3D object detection in RGB-D images", In *Proc. CVPR'16*, 2016.

[27] K. Subr, C. Soler, and F. Durand, "Edge-preserving multiscale image decomposition based on local extrema", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH Asia 2009*, vol. 28, issue 5, Dec. 2009.

[28] C. Tomasi, and R. Manduchi, "Bilateral filtering for gray and color images", in *Proc. ICCV'98*, 1998, pp. 839–846.

[29] O. Tuzel, F. Porikli, and P. Meer, "Region covariance: A fast descriptor for detection and classification", in *Proc. ECCV'06,* 2006, pp. 589– 600.

[30] J. van de Weijer, and R. Van den Boomgaard, "Local mode filtering", in *Proc. CVPR'01,* 2001, pp. II–428-433.

[31] B. Weiss, "Fast median and bilateral filtering", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH 2006,* vol. 25, issue 3, 2006, pp. 519–526.

[32] L. Xu, C. Lu, Y. Xu, and J. Jia, "Image smoothing via L_0 gradient minimization", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH Asia 2011*, vol. 30, issue 6, Dec. 2011.

[33] L. Xu, J. SJ. Ren, Q. Yan, R. Liao, and J. Jia, "Deep edge-aware filters", in *Proc. ICML'15,* 2015.

[34] L. Xu, Q. Yan, Y. Xia, J. Jia, "Structure extraction from texture via relative total variation", *ACM Transactions on Graphics (TOG) – Proc. of ACM SIGGRAPH Asia 2012*, vol. 31, issue 6, Nov. 2012.

[35] Q. Yang, "Semantic filtering", in *Proc. CVPR'16*, 2016, pp. 4517-4526.

[36] Q. Yang, K.-H. Tan, and N. Ahuja, "Real-time O(1) bilateral filtering", in *Proc. CVPR'09,* 2009, pp. 557–564.

[37] Q. Yang, S. Wang, and N. Ahuja, "SVM for edge-preserving filtering", in *Proc. CVPR'10*, 2010, pp. 1775–1782.

[38] Q. Zhang, L. Xu, and J. Jia, "100+ times faster weighted median filter", in *Proc. CVPR'14,* 2014.

[39] Q. Zhang, L. Xu, and J. Jia, "Rolling guidance filter", in *Proc. ECCV'14*, 2014.