

# Adaptive Rendering based on Adaptive Order Selection

Hongliang Yuan<sup>1,2</sup>, Changwen Zheng<sup>1</sup>, Quan Zheng<sup>1,2</sup> and Yu Liu<sup>1,2</sup>

<sup>1</sup>Institute of Software, Chinese Academy of Sciences, Beijing, China

<sup>2</sup>University of Chinese Academy of Sciences, Beijing, China

**Keywords:** Adaptive Rendering, Adaptive Order Selection, Monte Carlo Ray Tracing, Mean Squared Error.

**Abstract:** We propose a new adaptive sampling and reconstruction method based on a novel, adaptive order polynomial fitting which can preserve various high-frequency features in generated images and meanwhile mitigate the noise efficiently. Some auxiliary features have strong linear correlation with luminance intensity in the smooth regions of the image, but the relationship does not hold in the high-frequency regions. In order to handle these cases robustly, we approximate luminance intensity in the auxiliary feature space by constructing local polynomial functions with order varying adaptively. Firstly, we sample the image space uniformly. Then we decide the order of fitting with the least estimated mean squared error (MSE) for each pixel. Finally, we distribute additional ray samples to areas with higher estimated MSE if sampling budget remains. We demonstrate that our method makes significant improvement in terms of both numerical error and visual quality compared with the state-of-the-art.

## 1 INTRODUCTION

Monte Carlo (MC) ray tracing (Kajiya, 1986) is the commonly acknowledged efficient algorithm to synthesize photo-realistic images from 3D models. However, the rendered images generated by MC ray tracing at low sampling rate are noisy, and generating noise-free images needs dozens of hours. In order to synthesize satisfactory images in reasonable time, removing MC noise in image-space has been actively studied recently. The model of filtering MC noise can be formulated as:

$$y = m(\mathbf{x}) + \varepsilon \quad (1)$$

where  $y$  is a scalar response variable,  $\mathbf{x} = (x_1, \dots, x_d)^T$  is a  $d$ -dimensional vector of explanatory variables and  $\varepsilon$  represents MC noise.

From the perspective of rendering,  $m(\mathbf{x})$  and  $y$  denote an unknown ground truth image and a noisy image, respectively. The vector  $\mathbf{x}$  contains spatial and range components of the image, as well as additional geometric information including surface normal, texture color and direct illumination visibility and so on. The additional geometric information, which is also called the auxiliary features, is computed at every pixel where we average out geometries collected from multiple sample rays.

Image-space denoising locally approximates the color value of a pixel  $c$  as a weighted average of its

neighbors:

$$\hat{m}(\mathbf{x}_c; \mathbf{H}) = \frac{\sum_{i \in \Omega_c} y_i K_{\mathbf{H}}(i, c)}{\sum_{i \in \Omega_c} K_{\mathbf{H}}(i, c)} \quad (2)$$

where  $\hat{m}(\mathbf{x}_c; \mathbf{H})$  is the filtered color value of pixel  $c$ ,  $y_i$  is the input color value of pixel  $i$ ,  $\Omega_c$  is a local window centered on  $c$  and  $K_{\mathbf{H}}(\cdot)$  denotes a nonnegative weight function with a bandwidth matrix  $\mathbf{H}$ . The equation is called Nadaraya-Watson (Nadaraya, 1964; Watson, 1964) estimator which is the weighted least square solution of the following optimization problem:

$$\min_{\alpha} \sum_{i \in \Omega_c} (y_i - \alpha)^2 K_{\mathbf{H}}(i, c) \quad (3)$$

i.e.,  $\hat{m}(\mathbf{x}_c; \mathbf{H}) = \hat{\alpha}$ . The solution can be also expressed as follows:

$$\begin{aligned} \hat{\alpha} &= \mathbf{e}^T (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{Y} \\ &= \mathbf{L}^T(\mathbf{x}_c) \mathbf{Y} \end{aligned} \quad (4)$$

where  $\mathbf{e} = (1)^T$ ,  $\mathbf{Y} = (y_1, \dots, y_n)^T$  and  $n$  represents the number of pixels within a filtering window centered on  $c$ ,  $\mathbf{W} = \text{diag}(K_{\mathbf{H}}(1, c), \dots, K_{\mathbf{H}}(n, c))$  and  $\mathbf{X} = (1, \dots, 1)^T$ .  $\mathbf{L}^T(\mathbf{x}_c)$  is called equivalent kernel. The bilateral filter (Tomasi and Manduchi, 1998) is an example of Nadaraya-Watson estimator. The weight  $K_{\mathbf{H}}(i, c)$  is defined as:

$$K_{\mathbf{H}}(i, c) = \exp\left(\frac{-\|i - c\|^2}{2\sigma_s^2}\right) \exp\left(\frac{-\|y_i - y_c\|^2}{2\sigma_r^2}\right) \quad (5)$$

so that the bandwidth matrix  $H = \text{diag}(\sigma_s, \sigma_s, \sigma_r, \sigma_r, \sigma_r)$ . The non-local means (NL-Means) filter (Buades et al., 2008) is also a generalization of Nadaraya-Watson estimator where the weight is computed based on small patches. From the Equation 4, we know the Nadaraya-Watson estimator  $\hat{\alpha}$  is a locally weighted average of the response variables in the neighborhood of the given pixel  $c$ .

The Nadaraya-Watson estimator converges more slowly at the boundary and its conditional variance is larger in practice for points on the boundary than for points on the interior. Moon et al. (Moon et al., 2014) introduces a local linear estimator to remove MC noise:

$$\min_{\alpha, \beta} \sum_{i \in \Omega_c} (y_i - \alpha - \beta^T(\mathbf{x}_i - \mathbf{x}_c))^2 K_{\mathbf{H}}(i, c) \quad (6)$$

Its solution is also expressed as the Equation 4, but  $\mathbf{e} = (1, 0, \dots, 0)^T$  is a  $(d+1)$ -dimensional vector and the  $i$ th row of  $\mathbf{X}$  is set as  $(1, (\mathbf{x}_i - \mathbf{x}_c)^T)$ .

Intuitively it is clear that in the smooth region of the image the Nadaraya-Watson estimator or local linear estimator is preferable, whereas in the high-frequency region due to MC effects such as motion blur and depth-of-field, polynomials of higher order such as local cubic estimator is recommendable. Hence, the order of local polynomial fitting should be varied to reflect the local curvature of the unknown ground truth image.

Based on previous analysis, we present a novel, adaptive order polynomial fitting based adaptive sampling and reconstruction method to effectively handle various MC rendering effects (e.g., motion blur, depth-of-field, soft shadow, etc). Given a fixed spatial bandwidth, our method estimates an unknown function based on a data-driven variable order selection procedure. Our core idea is to develop a robust estimate of the MSE of each pixel and to use this estimate as a criterion for adaptive order selection (AOS), i.e. varying the order of the Taylor series approximation. Our idea is based on obvious observations that a reference image in the area of motion blur no longer has a linear correlation with given auxiliary features (e.g., textures). Specifically, we make the following technical contributions: For a given fixed spatial bandwidth and each order (i.e., local linear estimator and local cubic estimator), our method estimates the MSE of each pixel which is decomposed into estimation of bias and variance terms. The local optimal order is selected with the least MSE.

## 2 PREVIOUS WORK

Adaptive sampling and reconstruction was pioneered by Kajiya (Kajiya, 1986). The key of adaptive sampling is to develop a robust error metric which can guide where we need to allocate more ray samples. Common error metrics includes Stein's Unbiased Risk Estimator (SURE) (Stein, 1981), contrast metric (Mitchell, 1987), median absolute deviation (MAD) (Donoho and Johnstone, 1994), MSE which can be decomposed into variance plus the square of bias and so on. We will review the most recent adaptive renderings based on previous analysis and classify them into Nadaraya-Watson estimator, local linear estimator and higher order estimator.

**Nadaraya-Watson Estimator.** Li et al. (Li et al., 2012) leverage cross-bilateral filter to filter MC noise. They compute the weight between two pixels similar to the Equation 5 as well as considering auxiliary features. Sen et al. (Sen and Darabi, 2012) also use cross-bilateral filter, but utilizing mutual information to measure the relation between feature differences and filter weights. Rousselle et al. (Rousselle et al., 2013) adopt SURE to estimate MSE of three cross-bilateral filters with the same spatial support, but different their other parameters. Moon et al. (Moon et al., 2013) use a virtual flash image as a stopping function to compute the weight of a cross-bilateral filter. Rousselle et al. (Rousselle et al., 2012) exploit NL-Means filter for adaptive rendering, while Delbraccio et al. (Delbraccio et al., 2014) propose a multi-scale NL-Means filter for adaptive reconstruction. Kalantari et al. (Kalantari et al., 2015) present a machine learning approach to reduce MC noise. These methods are all the cases of the Nadaraya-Watson estimator.

**Local Linear Estimator.** Moon et al. (Moon et al., 2014) construct local linear estimator to estimate the color value of each pixel. Moon et al. (Moon et al., 2015) propose to approximate the color value of each pixel in a prediction window with multiple, but sparse local linear estimators. Bitterli et al. (Bitterli et al., 2016) study existing approaches and present nonlinearly weighted first-order regression for denoising MC noise, which is also the case of local linear estimator, but computing the weight based on small patches.

**High Order Estimator.** Most recently, Moon et al. (Moon et al., 2016) locally approximate an image with polynomial functions and the optimal order of each polynomial function is estimated so that multi-stage reconstruction error can be minimized. How-

ever, they construct polynomial functions only considering pixel position (2D). In contrast, we construct higher order regression functions considering all auxiliary features and present a bias estimation leveraging Taylor expansion.

There are some adaptive renderings which do not belong to these categories. For example, multi-dimensional adaptive rendering proposed by Hachisuka et al. (Hachisuka et al., 2008), and adaptive wavelet rendering presented by Overbeck et al. (Overbeck et al., 2009).

### 3 LOCAL POLYNOMIAL FITTING

Local linear estimator can locally estimate image functions efficiently when non-linearity in the 2D image space becomes nearly linear in the auxiliary feature space. This is definitely true in smooth region of a noisy image, but in rendering, some MC effects such as glossy highlights and defocused areas can be nonlinear as shown in Fig. 1. In such cases, we had better select higher order of regression function to reduce bias due to the curvature of the signal being reconstructed.

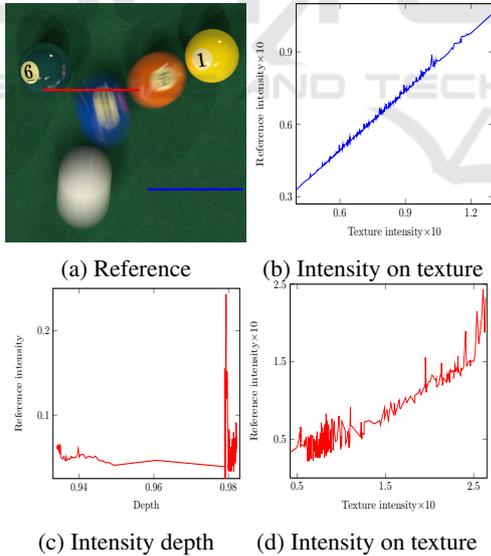


Figure 1: The reference image (a) is POOL scene generated with 32K ray samples per pixel. Intensity on texture plot (b) is drawn from data in blue line of (a) which shows strong linear correlation, where blue line is in smooth region. Intensity on depth plot (c) and on texture plot (d) are drawn from data in red line of (a) respectively which shows non-linearity, where red line covers smooth and motion blur regions.

Based on our observations, we propose an adap-

tive order polynomial fitting based method to reconstruct smooth and high-frequency areas adaptively and efficiently with fixed spatial bandwidth. Bias will increase and variance will decrease as bandwidth increases. On the contrary, bias will decrease and variance will increase as bandwidth decreases. This is the famous bias-variance trade-off issue. Bandwidth robustification can be explained as follows. If the given spatial bandwidth is too large, our method will choose a higher order fitting to reduce the bias. On the contrary, if the given spatial bandwidth is too small, our method will select a lower order fitting to reduce variance. As a result, this procedure adapts to smooth and high-frequency regions of a MC input image.

Weighted local regression (WLR) proposed by Moon et al. (Moon et al., 2014) uses local linear estimator to approximate the nonlinear effects in a high dimensional auxiliary feature space. It is susceptible to oversmooth the regions with high curvature, even though they suggest a sophisticated two-step optimization procedure. Fig. 2 demonstrates that our method can improve the image quality compared with WLR.

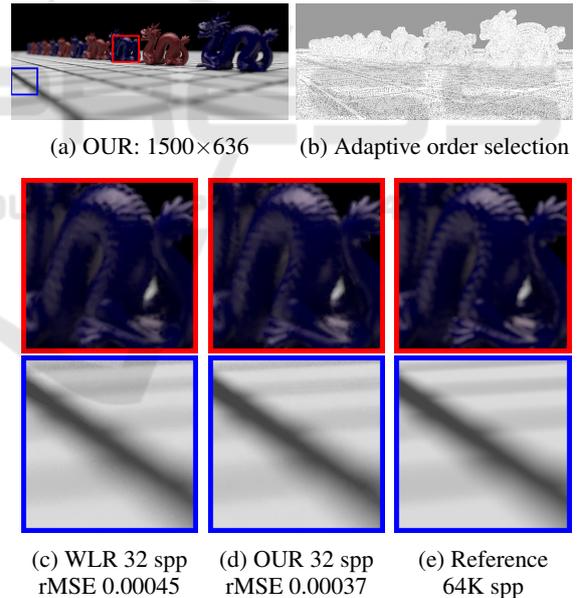


Figure 2: Adaptive order selection image (b) is generated when reconstructing our image (a), it is clear that in smooth region (e.g., black region in (a)) local linear estimator indicated by grey is optimal choice. Whereas, in high-frequency regions (e.g., red and blue rectangle in (a)), local cubic estimator indicated by white can preserve details efficiently, see details comparison with WLR. Even though relative MSE (rMSE defined in Sec. 7) of WLR and our method is at same order of magnitude, our method has an advantage over WLR at the high-frequency regions from the perspective of visual quality.

In order to perform AOS, we need fit a high or-

der multivariate regression function. For the sake of reducing computational overhead, we bring in partial local polynomial regression. For example, we will use the following design matrix for a local polynomial of order  $v$ :

$$\mathbf{X} = \begin{pmatrix} 1 & (\mathbf{x}_1 - \mathbf{x}_c)^T & \dots & (\mathbf{x}_1 - \mathbf{x}_c)^{vT} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (\mathbf{x}_n - \mathbf{x}_c)^T & \dots & (\mathbf{x}_n - \mathbf{x}_c)^{vT} \end{pmatrix} \quad (7)$$

where  $(\mathbf{x}_i - \mathbf{x}_c)^{vT} = ((x_{i1} - x_{c1})^v, \dots, (x_{id} - x_{cd})^v)^T$ ,  $i = 1, \dots, n$ . The design matrix is only used for deciding which order of fitting can minimize the MSE at each pixel. Once it is selected, we then estimate color value and MSE which can guide the additional sampling.

Then, the optimization problem of higher order regression can be written as follows:

$$\min_{\beta} \sum_{i=1}^n (y_i - \beta_0 - \sum_{v=1}^p \beta_v^T (\mathbf{x}_i - \mathbf{x}_c)^v)^2 \mathbf{K}_H(\mathbf{x}_i - \mathbf{x}_c) \quad (8)$$

where  $p$  is the order of the polynomial function,  $\beta = (\beta_0, \beta_1^T, \dots, \beta_p^T)^T$ ,  $\beta_v = (\beta_{v1}, \dots, \beta_{vd})^T$  and  $(\mathbf{x}_i - \mathbf{x}_c)^v = ((x_{i1} - x_{c1})^v, \dots, (x_{id} - x_{cd})^v)$ .  $\mathbf{K}_H(i, c) = \exp\left(\frac{-\|i-c\|^2}{2h^2}\right)$ , since we only optimize the order of local polynomial function, we select fixed spatial filtering window  $h$ . Solving (8) by the least squares method provides the solution:

$$\hat{\beta}(\mathbf{x}_c) = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{Y} \quad (9)$$

whose conditional mean and variance are:

$$\begin{aligned} E(\hat{\beta}(\mathbf{x}_c) | \mathbf{x}_1, \dots, \mathbf{x}_n) &= (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{m} \\ &= (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} (\mathbf{X} \beta + \mathbf{r}) \\ &= \beta + (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \mathbf{r} \end{aligned} \quad (10)$$

$$\begin{aligned} var(\hat{\beta}(\mathbf{x}_c) | \mathbf{x}_1, \dots, \mathbf{x}_n) &= (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{W}^2 \mathbf{X}) \\ &\quad \times (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \sigma^2(\mathbf{x}_c) \end{aligned} \quad (11)$$

where  $\mathbf{m} = (m(\mathbf{x}_1), \dots, m(\mathbf{x}_n))^T$  denotes the unknown ground truth image and  $\mathbf{r} = \mathbf{m} - \mathbf{X}\beta$  represents bias image between the unknown ground truth image and estimated image. Since  $\mathbf{r}$  contains unknown term  $\mathbf{m}$ , we need to estimate it (Sec. 4).

## 4 ADAPTIVE ORDER SELECTION

Fan et al. (Jianqing Fan, 1995a) proposed to use the estimated MSE as a criterion to opt for the best order of regression function, but their method can only apply to regression function with one response variable. Motivated by the previous work, we present MSE estimation for multivariate regression. We introduce bias estimation first.

**Bias Estimation.** We estimate bias vector  $\mathbf{r} = \mathbf{m} - \mathbf{X}\beta = (r_1, \dots, r_n)^T$  via Taylor expansion

$$r_i \approx \sum_{v=p+1}^{p+a} \beta_v^T (\mathbf{x}_i - \mathbf{x}_c)^v \equiv \xi_i \quad (12)$$

where  $a$  is an order which is greater than the approximation order  $p$ . Fan et al. (Jianqing Fan, 1995b) discuss the choice of the approximation order, who suggest to use  $a = 2$  for practical implementation. In this paper, we follow this recommendation. The unknown bias in  $\xi = (\xi_1, \dots, \xi_n)^T$  can be estimated by fitting a local polynomial with the order  $p + a$  and the spatial bandwidth  $h$  defined previously. Let  $\hat{\beta}_{p+1}^*, \dots, \hat{\beta}_{p+a}^*$  be the locally weighted least squares estimation, then  $\hat{\xi}_i = \sum_{v=p+1}^{p+a} \hat{\beta}_v^{*T} (\mathbf{x}_i - \mathbf{x}_c)^v$ . Bias at pixel  $c$  is estimated by

$$\widehat{\mathbf{bias}}_p(\mathbf{x}_c) = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \hat{\xi} \quad (13)$$

**Variance Estimation.** After fitting a local polynomial with order  $p + a$ , the estimation result  $\hat{m}(\mathbf{x}_c; \mathbf{H})$  can be expressed by  $\sum_{i=1}^n l_i(\mathbf{x}_c) y_i$ , where  $l_i(\mathbf{x}_c)$  is the item of row 0 and column  $i$  of matrix  $(\mathbf{X}^{*T} \mathbf{W}^* \mathbf{X}^*)^{-1} \mathbf{X}^{*T} \mathbf{W}$ , then

$$\hat{\sigma}^2(\mathbf{x}_c) = \sum_{i=1}^n (l_i(\mathbf{x}_c))^2 \sigma^2(y_i) \quad (14)$$

where  $\mathbf{X}^*$  and  $\mathbf{W}^*$  are the design matrix and weight matrix respectively for the local  $(p + a)$ th order polynomial fitting.  $\sigma^2(y_i)$  is the mean sample variance at pixel  $i$ . Substituting (14) into (11) produces an approximated variance matrix:

$$\begin{aligned} \widehat{\mathbf{var}}_p(\mathbf{x}_c) &= (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} (\mathbf{X}^T \mathbf{W}^2 \mathbf{X}) \\ &\quad \times (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \sigma^{*2}(\mathbf{x}_c) \end{aligned} \quad (15)$$

**MSE Estimation.** Equation (13) is a  $(d \times p + 1) \times 1$  bias vector for  $\beta$ , the first item is the estimated bias of  $\beta_0$ , denoted by  $\widehat{\mathbf{bias}}_p[0]$ . Equation (15) is a  $(d \times p + 1) \times (d \times p + 1)$  covariance matrix for  $\beta$ ,  $\widehat{\mathbf{var}}_p(\mathbf{x}_c)[0][0]$  is the estimated variance of  $\beta_0$ . Hence, the MSE of  $\hat{m}(\mathbf{x}_c)$  is estimated by

$$\widehat{MSE}_p(\mathbf{x}_c; \mathbf{H}) = \widehat{\mathbf{var}}_p(\mathbf{x}_c)[0][0] + (\widehat{\mathbf{bias}}_p(\mathbf{x}_c)[0])^2 \quad (16)$$

**Algorithm of AOS.** With the estimated MSE in equation (16), algorithm 1 describes the procedure of AOS-based MSE estimation, sampling map generating and color value estimation for each pixel.

Algorithm 1: AOS-based adaptive rendering.

**Input:**

Local polynomial fitting of order  $p_{max} = 3$  and order increment  $a = 2$ , Auxiliary feature vector  $\mathbf{x}_i$  and intensity  $y_i$  generated by renderer

**Output:**

Sampling density for next iteration or final filtered image

- 1: **for** each pixel **do**
- 2: Dimension reduction by truncated singular value decomposition (Moon et al., 2014);
- 3: Fit a local polynomial function of order  $p_{max} + a$ ;
- 4: **for** each order  $l \in \{1, 3\}$  **do**
- 5: Get coefficients for a local polynomial fitting of order  $l + a$  from step 3;
- 6: Estimate  $\widehat{MSE}_l(\mathbf{x}_c; \mathbf{H})$  by equation (16);
- 7: **end for**
- 8: Choose order  $l$  for this pixel with the least estimated  $\widehat{MSE}_l(\mathbf{x}_c; \mathbf{H})$ ;
- 9: For each channel, compute  $\hat{m}(\mathbf{x}_c; \mathbf{H})$  using local polynomial fitting with order  $l$ ;
- 10: **end for**
- 11: **if** Sampling budget remains **then**
- 12: **return** a sampling density based on  $\Delta rMSE(\mathbf{x}_c)$  (Sec. 5);
- 13: **else**
- 14: **return**  $\hat{m}(\mathbf{x}_c; \mathbf{H})$ ;
- 15: **end if**

## 5 ADAPTIVE SAMPLING

The state-of-the-art adaptive rendering methods (Li et al., 2012), (Rousselle et al., 2013) and (Moon et al., 2014; Moon et al., 2015)) adopt a common iterative approach to distribute additional ray samples to the regions with higher estimated MSE. In this paper, we implement that iterative model. To guide our adaptive sampling, we uniformly distribute a small number of ray samples (e.g., four ray samples per pixel) at the first iteration. Then we estimate  $\widehat{MSE}(\mathbf{x}; \mathbf{H})$  by our AOS-based reconstruction method with fixed spatial bandwidth and predict an error reduction  $\Delta MSE(\mathbf{x}) = \widehat{MSE}(\mathbf{x}; \mathbf{H}) \times n(\mathbf{x})^{\frac{-4}{d+4}}$  for each pixel, where the error reduction factor  $n(\mathbf{x})^{\frac{-4}{d+4}}$  is derived in (Moon et al., 2014) and  $n(\mathbf{x})$  is the number of ray samples which has been already allocated.

Rousselle et al. (Rousselle et al., 2011) suggest relative MSE which is based on human visual perception that is more sensitive to darker areas. Since then, most of adaptive rendering methods (e.g., (Li et al., 2012), (Rousselle et al., 2012; Rousselle et al., 2013)

and (Moon et al., 2014; Moon et al., 2015)) adopt that suggestion. The relative MSE is computed as follows:  $\Delta rMSE(\mathbf{x}) = \frac{\Delta MSE(\mathbf{x})}{\hat{m}^2(\mathbf{x}; \mathbf{H}) + \epsilon}$ ,  $\epsilon$  is a small number, e.g., 0.0001, which is used to avoid a divide-by-zero. If sampling budget still remains, we allocate ray samples per pixel which are proportional to  $\Delta rMSE(\mathbf{x})$ .

## 6 IMPLEMENTATION DETAIL

We implement our AOS-based adaptive rendering method using CUDA on top of PBRT2 (Pharr and Humphreys, 2010). The auxiliary features contain 10 dimensions: 2D pixel position, 1D depth, 3D texture, 3D normal and 1D direct illumination visibility. We normalize auxiliary features to the range [0,1]. We find that the direct illumination visibility is very noisy at low sampling rate, so we prefilter direct illumination visibility leveraging a non-local means filter in an  $10 \times 10$  window with patch size  $6 \times 6$  as was done in Rousselle et al. (Rousselle et al., 2013) and Kalantari et al. (Kalantari et al., 2015). Even though other features have less noise than visibility, we also apply non-local means filtering for them.

In order to continue to reduce the noise and mitigate the curse of dimensionality in multivariate regression, we apply the truncated singular value decomposition (Moon et al., 2014) to auxiliary features within given filtering window. In all our tests, we use a  $11 \times 11$  filtering window in the iterative steps and  $19 \times 19$  filtering window in the final step. The reason why we use smaller filtering windows in the iterative stage is that we only estimate relative MSE. This decision has only a slight effect on adaptive sampling. We opt for a small number of iterations (i.e., 3) for adaptive sampling.

We tested two orders of local polynomial fitting, one order is 1 which leads to local linear estimator and the other is 3 which leads to local cubic estimator. All the choice of the order of fitting is odd-order, because the asymptotic variance becomes large when moving from an odd-order estimation to its consecutive even-order estimation.

## 7 RESULTS AND DISCUSSIONS

We test our method using a mobile workstation with Intel Core i7-4700MQ CPU @2.40GHz, and NVIDIA Quadro K1100M with CUDA 7.5 SDK for accelerating our proposed reconstruction method based on adaptive order selection (AOS). We compare our method with low discrepancy (Dobkin et al.,

1996) (LD) and the state-of-the-art adaptive rendering methods including WLR (Moon et al., 2014), APR (Moon et al., 2016) and LBF (Kalantari et al., 2015). To evaluate the WLR and LBF method, we use the source code provided by authors and set parameters recommended by their corresponding papers. Since the authors of APR method didn't make their source code public, we implemented their algorithm on the CPU using the Eigen 3 library for efficient matrix operations. Even though our implementation might not reach the same image quality as in their original paper, it demonstrates that locally changing polynomial function order can generate high-quality image.

For purpose of comparing image quality generated by different methods in terms of a quantitative measure, we introduce the relative mean squared error (rMSE) (Rousselle et al., 2011) which is defined as:  $\frac{1}{n} \sum_{i=1}^n \frac{(\hat{m}(\mathbf{x}_i) - m(\mathbf{x}_i))^2}{m(\mathbf{x}_i)^2 + \epsilon}$ , where  $\epsilon = 0.001$  prevents a divide by zero,  $m(\mathbf{x})$  refers to the ground truth image. We also use the structural similarity (SSIM) (Wang et al., 2004) index to measure the similarity between two images.

We evaluate our method on rendering following scenes: POOL ( $1024 \times 1024$ ), SANMIGUEL ( $1024 \times 1024$ ), SIBENIK ( $1024 \times 1024$ ) and TEAPOT ( $1024 \times 1024$ ). These scenes include a variety of MC effects, e.g., motion blur, depth-of-field, soft shadow, glossy reflection and so on. For each comparison, we adopt equal-sample comparison with LBF and WLR method, and equal-time comparison with LD method. Our method and WLR implement adaptive sampling, while LBF method only implement uniform sampling. Considering this case, all methods adopt uniform sampling for SANMIGUEL scene.

**Motion Blur Scene Comparison.** The POOL scene, in which three billiard balls have different motion directions, is used to compare motion blur effect for different methods. Fig. 3 shows comparison results. LD generates noisy result even with larger ray samples than other methods. LBF leaves artifacts in the motion blur regions (blue box). The reason is that LBF does not implement adaptive sampling and rendering motion blur effect needs more ray samples. APR method can preserve high-frequency edge while WLR gives oversmoothed it (red box). Our method not only generates a high-quality reconstruction result on the motion blurred regions but also preserves soft shadows clearly. In addition, our method obtains a lower rMSE and higher SSIM than other methods.

**Complex Scene Comparison.** The SANMIGUEL scene has complex geometric structure with more than 10 million triangles due to the use of object instancing and is widely used for adaptive rendering techniques. Fig. 4 illustrates comparison results. LD method produces a very noisy image. Our method preserves the edge and textures on wall (closeup in red box) and the foliage details as well as the small direct soft shadows (closeup in blue box). APR method can preserve high-frequency edge and also the foliage details. WLR fails to preserve edge and LBF tends to blur textures on wall and the foliage details. Overall, our method generates more visual pleasing images than other techniques.

**Depth-of-field Scene Comparison.** Fig. 5 illustrates depth-of-field effect comparison using the SIBENIK scene, which has an environment light which can be seen by refraction through the windows. LBF and WLR produces an over-blurred fence (closeup in blue box). Our method, on the other hand, preserves the fence as on the reference image.

**Glossy Reflection Scene Comparison.** The TEAPOT scene (Fig. 6) with glossy reflection and high-frequency bump mapping on the floor is recognized as a challenging scene. LD produces noisy image even with more than two times ray samples than ours. All methods fail to reconstruct the bump map (closeup in blue box) well. In addition, our method preserves glossy highlights more accurately than WLR which is inclined to oversmooth glossy regions (closeup in red box).

**Convergence Comparison.** We also plot convergence rate for the POOL and SANMIGUEL scenes comparing with LD, LBF and WLR methods, see Fig. 7. Our method and WLR adopt adaptive sampling for the POOL scene. Considering that LBF method does not implement adaptive sampling, all the methods use uniform sampling for the SANMIGUEL scene.

## 8 CONCLUSIONS

We have proposed a novel, adaptive order polynomial fitting based adaptive rendering approach for efficiently denoising images with diverse MC effects. Given fixed spatial bandwidth, we handle bias-variance trade-off by varying order of polynomial fitting. For the given two choices of the order of fitting, we estimate the MSE for each pixel and select

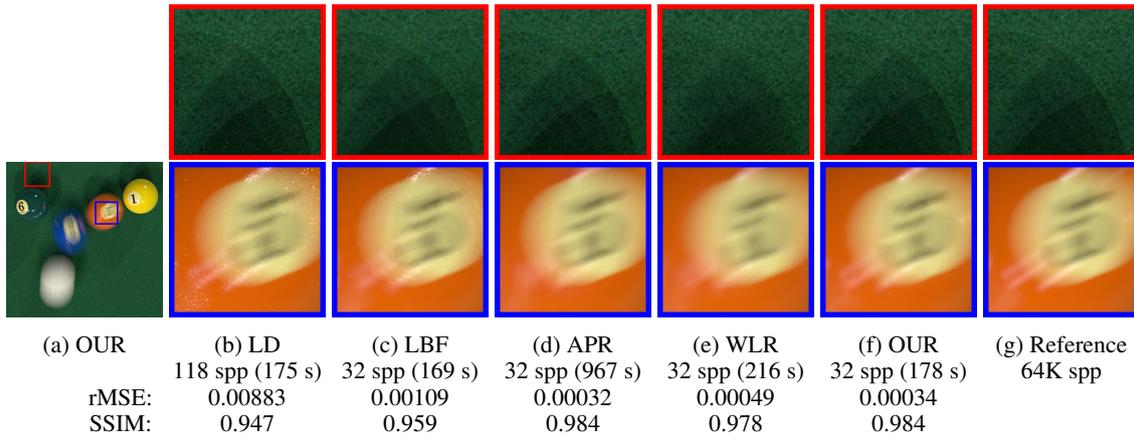


Figure 3: Motion blur scene comparison.

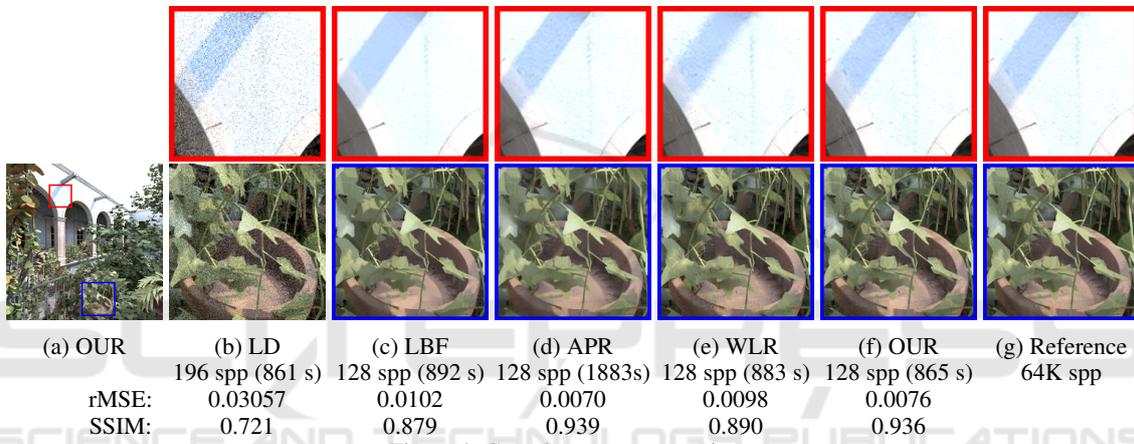


Figure 4: Complex scene comparison.

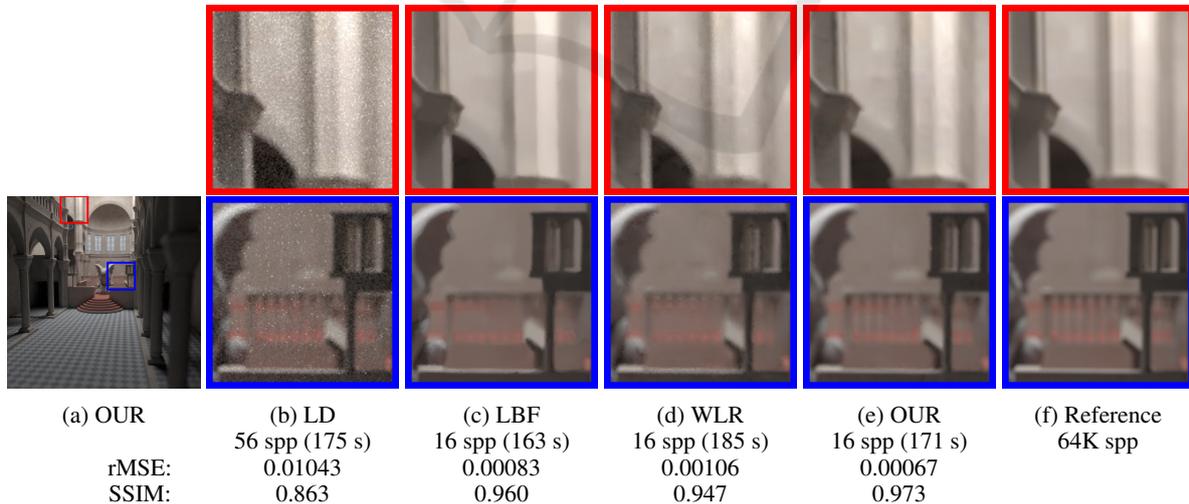


Figure 5: Depth of field scene comparison.

the order of fitting with the least estimated MSE adaptively. We also distribute additional ray samples to the regions with higher estimated MSE iteratively. We compare our method with the state-of-the-art meth-

ods on a wide variety of MC distributed effects. The comparison shows that our method enhances the image quality in terms of both numerical error and visual quality over the previous methods. In the future, we

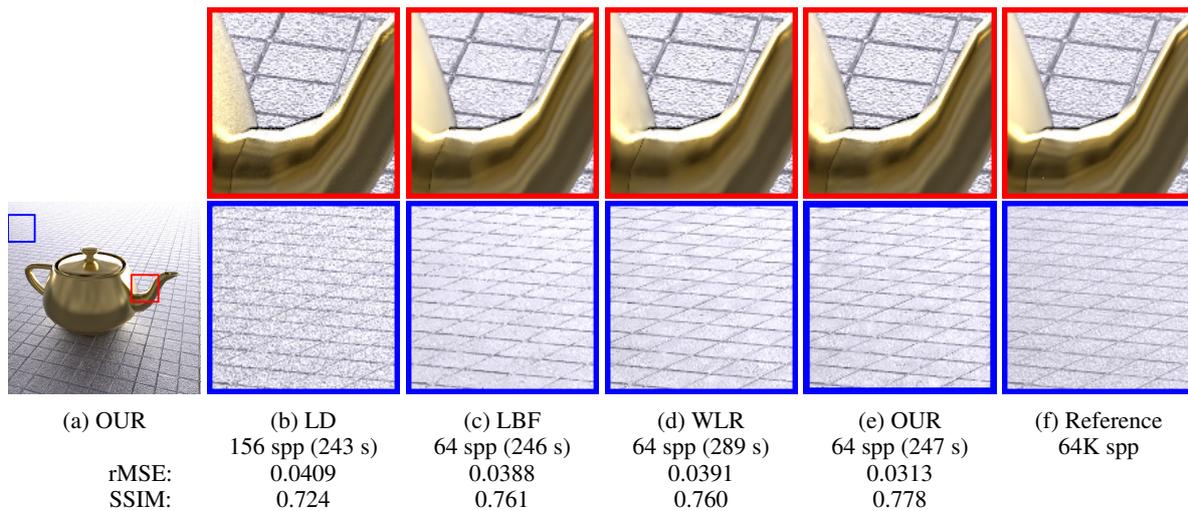


Figure 6: Glossy reflections scene comparison.

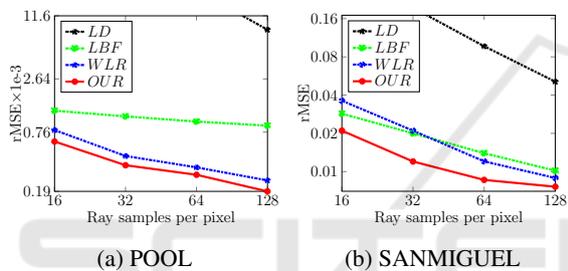


Figure 7: Convergence plots in LD, LBF, WLR and our method. Our method outperforms other three methods across all the tested ray samples per pixel.

will explore extension of AOS-based adaptive rendering to handle animated images.

## REFERENCES

Bitterli, B., Rousselle, F., Moon, B., Guitián, J. A. I., Adler, D., Mitchell, K., Jarosz, W., and Novák, J. (2016). Nonlinearly weighted first-order regression for denoising Monte Carlo renderings. *Comput. Graph. Forum*, 35(4):107–117.

Buades, A., Coll, B., and Morel, J.-M. (2008). Nonlocal image and movie denoising. *Int. J. Comput. Vision*, 76(2):123–139.

Delbracio, M., Musé, P., Buades, A., Chauvier, J., Phelps, N., and Morel, J.-M. (2014). Boosting Monte Carlo rendering by ray histogram fusion. *ACM Trans. Graph.*, 33(1):8:1–8:15.

Dobkin, D. P., Eppstein, D., and Mitchell, D. P. (1996). Computing the discrepancy with applications to supersampling patterns. *ACM Trans. Graph.*, 15(4):354–376.

Donoho, D. L. and Johnstone, I. M. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455.

Hachisuka, T., Jarosz, W., Weistroffer, R. P., Dale, K., Humphreys, G., Zwicker, M., and Jensen, H. W. (2008). Multidimensional adaptive sampling and reconstruction for ray tracing. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, pages 33:1–33:10, New York. ACM.

Jianqing Fan, I. G. (1995a). Adaptive order polynomial fitting: Bandwidth robustification and bias reduction. *Journal of Computational and Graphical Statistics*, 4(3):213–227.

Jianqing Fan, I. G. (1995b). Data-driven bandwidth selection in local polynomial fitting: Variable bandwidth and spatial adaptation. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(2):371–394.

Kajiya, J. T. (1986). The rendering equation. In *Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '86, pages 143–150, New York. ACM.

Kalantari, N. K., Bako, S., and Sen, P. (2015). A machine learning approach for filtering Monte Carlo noise. *ACM Trans. Graph.*, 34(4):122:1–122:12.

Li, T.-M., Wu, Y.-T., and Chuang, Y.-Y. (2012). SURE-based optimization for adaptive sampling and reconstruction. *ACM Trans. Graph.*, 31(6):194:1–194:9.

Mitchell, D. P. (1987). Generating antialiased images at low sampling densities. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87, pages 65–72, New York. ACM.

Moon, B., Carr, N., and Yoon, S.-E. (2014). Adaptive rendering based on weighted local regression. *ACM Trans. Graph.*, 33(5):170:1–170:14.

Moon, B., Iglesias-Guitian, J. A., Yoon, S.-E., and Mitchell, K. (2015). Adaptive rendering with linear predictions. *ACM Trans. Graph.*, 34(4):121:1–121:11.

Moon, B., Jun, J. Y., Lee, J., Kim, K., Hachisuka, T., and Yoon, S. (2013). Robust image denoising using a virtual flash image for monte carlo ray tracing. *Comput. Graph. Forum*, 32(1):139–151.

- Moon, B., McDonagh, S., Mitchell, K., and Gross, M. (2016). Adaptive polynomial rendering. *ACM Trans. Graph.*, 35(4):40:1–40:10.
- Nadaraya, E. A. (1964). On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142.
- Overbeck, R. S., Donner, C., and Ramamoorthi, R. (2009). Adaptive wavelet rendering. In *ACM SIGGRAPH Asia 2009 Papers*, SIGGRAPH Asia '09, pages 140:1–140:12, New York. ACM.
- Pharr, M. and Humphreys, G. (2010). *Physically Based Rendering: From Theory to Implementation*. Morgan Kaufmann Publishers Inc., San Francisco.
- Rousselle, F., Knaus, C., and Zwicker, M. (2011). Adaptive sampling and reconstruction using greedy error minimization. *ACM Trans. Graph.*, 30(6):159:1–159:12.
- Rousselle, F., Knaus, C., and Zwicker, M. (2012). Adaptive rendering with non-local means filtering. *ACM Trans. Graph.*, 31(6):195:1–195:11.
- Rousselle, F., Manzi, M., and Zwicker, M. (2013). Robust denoising using feature and color information. *Comput. Graph. Forum*, 32(7):121–130.
- Sen, P. and Darabi, S. (2012). On filtering the noise from the random parameters in Monte Carlo rendering. *ACM Trans. Graph.*, 31(3):18:1–18:15.
- Stein, C. M. (1981). Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6):1135–1151.
- Tomasi, C. and Manduchi, R. (1998). Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision, ICCV '98*, pages 839–, Washington, DC, USA. IEEE Computer Society.
- Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *Trans. Img. Proc.*, 13(4):600–612.
- Watson, G. S. (1964). Smooth regression analysis. *Sankhy: The Indian Journal of Statistics, Series A (1961-2002)*, 26(4):359–372.