Joint Stein's Unbiased Risk Estimation for Adaptive Sampling and Reconstruction

Tzu-Mao Li

August 6, 2017

Abstract

Adaptive sampling and reconstruction of Monte Carlo rendering often involves using a G-buffer to estimate the latent color of an image, by optimizing certain error estimator. Stein's Unbiased Risk Estimation (SURE) was shown to be suitable for being such error estimator. In this note, I show that SURE can and should be modified to take the correlation between the noise in G-buffer and color, and the correlation between color channels into account.

1 Motivation

Monte Carlo rendering generates photorealistic images but also takes days to compute them. Adaptive sampling and reconstruction methods are powerful variance reduction techniques that significantly reduce the computation time by adapting the sampling budget to difficult regions and filtering the images using auxiliary information obtained by the renderer (e.g. per-pixel normal or texture information). An important component of these methods is an error estimator, which can be used to identify difficult regions and to optimize the hyperparameters of the filters. In the past we showed that Stein's Unbiased Risk Estimator (SURE) [2] is suitable for being such error estimator [1]. The formulas in our previous paper assume the noise between G-buffer and color are uncorrelated. However, the assumption does not hold in general, since samples in G-buffer are often directly used to compute the color. We also ignored the correlation between color channels. In this note I derive a new SURE formula taking these correlations into account.

2 Joint Stein's Unbiased Risk Estimation

We are given a vector $x \in R^n$ contaminated by zero mean Gaussian noise n_x with covariance Σ_x:

$$\acute{x} = x + n_x. \tag{1}$$

We are also given a noisy vector $\acute{y} \in R^m$ *jointly normal distributed* with $\acute{x}$. We use $Cov(\acute{x}_i, \acute{y}_j)$ to denote the covariance between i-th component of $\acute{x}$ and j-th component of $\acute{y}$. In our application, $\acute{x}$ is the noisy Monte Carlo color estimation, and $\acute{y}$ is the *feature vector* (e.g. world position, texture albedo and surface normal). Thanks to central limit theorem, the noise will behave like normal distribution as we increase the number of samples.

In reconstruction for Monte Carlo rendering, a function f is applied to the color $\acute{x}$ and the feature $\acute{y}$. Our goal is to estimate the mean square error between the reconstructed color $f(\acute{x}, \acute{y})$ and the true latent color x. The error estimation can then be used for optimizing f's hyperparameters or adaptive sampling.

I will show that, for a weakly differentiable function f, the following function g:

$$g(\acute{x}, \acute{y}) = \|f(\acute{x}, \acute{y}) - \acute{x}\|^2 + 2\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial f_i(\acute{x}, \acute{y})}{\partial \acute{x}_j}Cov(\acute{x}_i, \acute{x}_j) + 2\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{\partial f_i(\acute{x}, \acute{y})}{\partial \acute{y}_j}Cov(\acute{x}_i, \acute{y}_j) - tr(\Sigma_x), \tag{2}$$

is an unbiased estimator of the mean square error:

$$E[g(\acute{x}, \acute{y})] = E[\|f(\acute{x}, \acute{y}) - x\|^2]. \tag{3}$$

g only depends on $\acute{x}$, $\acute{y}$, and the covariances, which we can obtain through finite number of Monte Carlo samples. Notice how g reduces to the original SURE [2] if $\acute{x}$ and $\acute{y}$ are uncorrelated and Σ_x's non-diagonal elements are zero.

In short, any adaptive sampling and reconstruction method for Monte Carlo rendering using the SURE formula and a noisy G-buffer can and should replace their error estimator with Eq. 2.

3 Derivation

In this section we derive the aforementioned function g. We expand the expectation in Eq. 3 using the linearity of expectation and the fact that $\acute{x}$ is Gaussian distributed:

$$
\begin{aligned}
E[\|f(\acute{x},\acute{y})-x\|^2] &= E[\|f(\acute{x},\acute{y})\|^2] + E[\|x\|^2] - 2E[f(\acute{x},\acute{y})^T x] \\
&= E[\|f(\acute{x},\acute{y})\|^2] + E[\|\acute{x}\|^2] - tr(\Sigma_x) - 2E[f(\acute{x},\acute{y})^T x] \\
&= E[\|f(\acute{x},\acute{y})\|^2] + E[\|\acute{x}\|^2] - tr(\Sigma_x) - 2\left(E[f(\acute{x},\acute{y})^T \acute{x}] - E[f(\acute{x},\acute{y})^T n_x]\right) \\
&= E[\|f(\acute{x},\acute{y})\|^2] + E[\|\acute{x}\|^2] - tr(\Sigma_x) - 2\left(E[f(\acute{x},\acute{y})^T \acute{x}] - \sum_{i=1}^{n} Cov(f_i(\acute{x},\acute{y}), n_{x,i})\right).
\end{aligned} \tag{4}
$$

By definition of covariance and Stein's lemma [2], we have:

$$
Cov(f_i(\acute{x},\acute{y}), n_{x,i}) = Cov(f_i(\acute{x},\acute{y}), \acute{x}_i) = \sum_{j=1}^{n} E[\frac{\partial f_i(\acute{x},\acute{y})}{\partial \acute{x}_j}] Cov(\acute{x}_i, \acute{x}_j) + \sum_{j=1}^{m} E[\frac{\partial f_i(\acute{x},\acute{y})}{\partial \acute{y}_j}] Cov(\acute{x}_i, \acute{y}_j). \tag{5}
$$

Inserting Eq. 5 back to Eq. 4 with some manipulations gives us:

$$
\begin{aligned}
E[\|f(\acute{x},\acute{y})-x\|^2] &= E\left[\|f(\acute{x},\acute{y})-\acute{x}\|^2 + 2\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial f_i(\acute{x},\acute{y})}{\partial \acute{x}_j} Cov(\acute{x}_i, \acute{x}_j) + 2\sum_{i=1}^{n}\sum_{j=1}^{m} \frac{\partial f_i(\acute{x},\acute{y})}{\partial \acute{y}_j} Cov(\acute{x}_i, \acute{y}_j) - tr(\Sigma_x)\right] \\
&= E[g(\acute{x},\acute{y})].
\end{aligned} \tag{6}
$$

References

[1] Tzu-Mao Li, Yu-Ting Wu, and Yung-Yu Chuang. SURE-based optimization for adaptive sampling and reconstruction. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2012)*, 31(6):186:1–186:9, November 2012.

[2] Charles M. Stein. Estimation of the mean of a multivariate normal distribution. *The Annals of Statistics*, 9(6):1135–1151, 1981.