Eurographics/ IEEE-VGTC Symposium on Visualization 2010
G. Melançon, T. Munzner, and D. Weiskopf
(Guest Editors)

Volume 29 (2010), Number 3

Estimation and Modeling of Actual Numerical Errors in Volume Rendering

Joel Kronander[1], Jonas Unger[1], Torsten Möller[2], Anders Ynnerman[1]

[1] VITA, Linköping University, Sweden, [2] GrUVi Lab, Simon Fraser University, Vancouver, Canada

Abstract

In this paper we study the comprehensive effects on volume rendered images due to numerical errors caused by the use of finite precision for data representation and processing. To estimate actual *error behavior we conduct a thorough study using a volume renderer implemented with arbitrary floating-point precision. Based on the experimental data we then model the impact of floating-point pipeline precision, sampling frequency and fixed-point input data quantization on the fidelity of rendered images. We introduce three models, an average model, which does not adapt to different data nor varying transfer functions, as well as two adaptive models that take the intricacies of a new data set and transfer function into account by adapting themselves given a few different images rendered. We also test and validate our models based on new data that was not used during our model building.*

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques—Graphics data structures and data types

1. Introduction

The reliance on volume visualization is increasingly common in a wide range of critical application areas such as medical diagnosis. The confidence of the user in the quality of the generated images therefore becomes of highest importance and is manifested in a renewed interest in the visualization community as uncertainty visualization [JS03]. This puts emphasis on software and system properties such as reliability, accuracy, stability and predictability. These requirements are often traded for the need of real-time interaction and efficiency of the rendering pipeline. While there has been significant effort to achieve the latter, less has gone into a comprehensive treatment of the resulting overall accuracy.

Our goal with this work is to increase understanding of the compound effects of uncertainty in the volume rendering pipeline. Specifically, we analyze the adverse effects of performing the computational steps in the rendering pipeline using finite precision. In contrast to previous work we study the effects of varying floating-point precision. This is motivated by the fact that floating-point numbers are the standard numerical representation in scientific computing. Furthermore, along with the development of modern *High Dynamic Range* imaging [YNCP06], the constraint of 8-bit fixed-point output is lifted. As volumetric data sets in many application areas today are increasingly large, understanding the effects of using different bit quantizations (e.g. during compression) is important for boosting data handling and increasing rendering performance. The precision requirements are also dependent on the number of computations, indirectly given by the sampling frequency. We therefore study the co-dependency effects of using varying input data quantization and sampling frequencies on various data sets and transfer function combinations. We analyze the resulting errors and their propagation through the pipeline. This analysis forms the basis for developing algorithms of high confidence, and a high fidelity standard for volume rendering. It can also be used to minimize the required rendering time given specific image fidelity requirements and parameter constraints. The presented analysis is applicable to both volume rendering users relying on floating point formats in todays hardware, and also the development of new rendering software and hardware with novel floating-point format implementations.

While a theoretical error analysis, typically based on maximum error estimations [Hig02], of the compound effect of the pipeline precision serves as a starting point, the error propagation through the rendering pipeline is either intractable or unrealistic when using maximum error bounds. Hence, in order to investigate the *actual* effects of limited

precision in practice, we perform a comprehensive study - first, via an experimental study we observe and tabulate the actual error behavior (Sections 3 and 4), which we then generalize into a model (an average, an adaptive and a hybrid model) (Section 5). These models have then been validated. Our experimental setup is realized using a renderer with arbitrary numerical precision. To make our study feasible, we have limited the investigation to an emission-absorption model for volume rendering, and do not include the effects of more advanced illumination models and derivative filters. We consider the standard linear interpolation filter for data reconstruction. Furthermore, we focus on the prediction of numerical error propagation and thus do not employ perceptual error metrics.

2. Background

Verification and validation of results are increasingly important topics for computer science in general [BO04]. This includes numerical errors, studied extensively in the past, both with regard to maximal error analysis and statistical error propagation [Hig02, WK08]. For rendering, previous work has mostly focused on assessing image fidelity based on separate stages of the pipeline, e.g. errors due to the sampling density [BMWM06], the impact of post-classified vs pre-classified pipelines [WMG98], bounds of integration errors [NA92], error metrics and properties of filters both for data reconstruction [MMMY97b, ML94] and gradient estimation [MMMY97a]. Due to the complexity of the interactions and co-dependencies between pipeline parameters, estimating such effects by a theoretical approach is often intractable, and instead a practical methodology is often preferred. An early attempt to such an analysis was made by Williams et al. [WU99] who derived guidelines and metrics for comparing volume rendered images. In the work by Kwansik et al. [KWP01], a set of test data sets is presented along with a discussion of parameters affecting image fidelity. More recently Giesen et al. [GMS*07] proposed a user study design to measure the impact of parameters.

The importance of numerical precision in volume rendering has been pointed out by several researchers. Meissner et al. [MHB*00] noted the importance of sufficient precision when comparing different volume rendering techniques. The implications of a limited numerical precision for the compositing step is also discussed by Engel et al. [EHK*06]. In the context of pre-classified volume rendering Wittenbrink et al. [WMG98] discussed the need for more than 8-bit fixed-point formats. A more detailed derivation of errors due to fixed-point precision was presented by Bitter et al. [BNMK04]. They derived the minimal fixed-point precision needed at each step in the pipeline for a required output precision. They also investigated the maximum error bounds for each step and showed that they are dependent on sampling frequency and volume size. However, due to the consideration of maximum errors, the analysis leads to unrealistic demands.

To our knowledge, no previous work has conducted a rigorous investigation of the compound effects of *actual* errors caused by varied floating-point pipeline precision, data quantization, and ray sampling frequency on real data sets and transfer functions. The need for such an analysis forms the starting point for the work presented here.

2.1. Volume rendering integral

In this paper, we consider an *emission-absorption* model of volume rendering [Max95] for which the volume rendering integral can be formulated as

$$I(D) = \int_0^D \tau(s(x))c(s(x)) * \exp\left(-\int_0^x \tau(s(\acute{x}))d\acute{x}\right)dx \quad (1)$$

where $s(x)$ denotes the scalar field, $\tau(s)$ the *absorption* and $c(s)$ the color intensity. We solve the integral as a Riemann sum by discretizing the ray into equidistant segments of length d [KE04]. Neglecting self-attenuation within ray segments and approximating *opacity* as $\alpha_i \approx \tau(s(x(id)))d$ we can derive the *front-to-back* compositing scheme

$$C_i^{'} = C_{i-1}^{'} + (1-\alpha_{i-1}^{'})C_i \quad (2)$$

$$\alpha_i^{'} = \alpha_{i-1}^{'} + (1-\alpha_{i-1}^{'})\alpha_i \quad (3)$$

where C_i is the approximated (associated) color $C_i \approx c(s(x(id))) * \alpha_i$, and $\alpha_i^{'}$ and $C_i^{'}$ are the accumulated opacity and (associated) color for the i*th* step. We limit our study to the standard DVR technique [EHK*06] – ray casting in a post-classified pipeline [WMG98]. We use transfer functions which specify an opacity value for a reference sampling step size $d_r = 1$ step/voxel. To use a sampling distance, d, other then the reference a correction of the optical properties is necessary. We calculate the corrected value as $\alpha_d = 1-(1-\alpha_{d_r})^{\frac{d}{d_r}}$ [EHK*06].

2.2. Floating-point arithmetic

We describe a binary floating-point system, F, by the parameters *precision*, p, and the minimal and maximal exponent range, $[e_{min}, e_{max}]$. The set of representable numbers in $F(p, e_{min}, e_{max})$ is given by $\{\hat{x} : \hat{x} = \pm m * 2^e\}$, where e is an integer in the range $[e_{min}, e_{max}]$ and m the *mantissa*, in binary form $m = 0.d_1d_2d_3...d_p$ [Gol91]. In floating-point arithmetic as opposed to fixed-point arithmetic, an exact result is not guaranteed for the elementary operations of addition and subtraction. Errors, due to *out-shifting*, are introduced when operations involving numbers with a large difference in the exponent are performed. For a comprehensive overview of the numerical properties of floating-point systems see Higham [Hig02] or Goldberg [Gol91]. The limited number of bits in the exponent implies that $\hat{x}$ is limited in magnitude to an interval defined as the *range* of the floating-point system, F. Using a low exponent range could introduce errors due to *underflow* and *overflow*. However most floating-point formats offer a large dynamic range for the context of volume rendering, and the accumulated color and

opacity resulting from the compositing equations is a monotonically increasing function. In this work we have therefore focused our efforts towards studying the precision, p, of a floating-point representation and use a fixed large exponent range of $e_{min} = -1000$, $e_{max} = 1000$ to marginalize the effects of underflow and overflow. For example when comparing a large set of images rendered for the Carp data set (see Section 5 and Supplementary Material) we found no difference between the exponent ranges $e_{min} = -1000$, $e_{max} = 1000$ and $e_{min} = -128$, $e_{max} = 128$.

3. Numerical investigation

To select which parameters and co-dependencies to explore, we have conducted an analysis of the propagation and accumulation of errors due to a limited precision in the pipeline. We group the parameters into two categories: *scalar parameters* that can be sampled arbitrarily dense, and *scenario parameters* that do not easily lend themselves to dense sampling and exhibit more complex behavior.

Scalar parameters p, q, s - The main focus of our study is the pipeline precision, p, affecting all stages of computation in the pipeline (sample positions, interpolation, classification, opacity correction, and compositing). We also consider the co-dependencies and effects of using a finite fixed-point input data quantization, q, which previously has often been assumed to be fixed to 8 or 12 bits (despite the fact that the results of flow simulations and tomography algorithms often are of higher precision). Previous work [BNMK04] has indicated that the number of samples composited affects the requirements on a fixed-point pipeline precision. In this paper we include the effects of sampling frequency, s, denoting how many samples are taken per voxel. While s only indirectly models the total number of samples, we consider the total number of samples directly during our model building (Section 5).

Scenario parameters d, l, t - To investigate the effect of *actual* rounding errors we have explored a series of scenarios. Each scenario is defined by a data set, d, with fixed lattice resolution, l, and a transfer function, t. These parameters are important as they define the frequency and opacity content of the input signal. They are, however, difficult to vary continuously since the input data needs to be well defined and only a small subset of all possible transfer functions are useful. The investigated scenarios span a number of different data sets from different application domains and, for each data set, a set of different transfer functions with varying frequency content and opacities. The data sets and transfer functions used are described in Section 3.4.

3.1. Parameter sampling

For each scenario we render a large set of images, where the pipeline precision, p, and one of the two parameters q or s are varied while keeping all other parameters fixed. Comparing each rendered image to a reference image and encoding their difference in a scalar error, a 2D scalar field we call *error landscape*, describing the co-dependencies between p and q or s is generated. To explore the correlation between p and q we study both synthetic and real data sets. The synthetic data sets allow us to systematically vary both p and q in the range from 2 to 40 bits. However, for real data sets the upper bound of the data quantization, q, is determined by the source of data acquisition. When investigating p vs. q we typically fix the sampling rate s to 2 steps per voxel but, for comparison, we also investigate a sampling frequency of 20 steps per voxel. For an overview of the scenarios see Table 1. For the co-dependence of p and s, we consider p in the range from 2 to 40 bits and s in the range 1 to 20 steps per voxel. This was performed for several data sets, lattice resolutions and transfer function settings, see Table 2.

3.2. Measuring image fidelity

While previous work has presented techniques to measure *image quality* by e.g. user studies [GMS*07], we have limited ourselves to investigate *image fidelity*. Image fidelity measures the similarity between an image and a reference, and does not convey information about how a human observer would classify the quality of the image. The focus on image fidelity is motivated by the fact that we consider a general property of the rendering pipeline, and not a specific visualization task. Modeling the human visual system is an ongoing research challenge, and previous work has presented several models to measure perceptual error, often yielding different results [PSC00]. Instead we seek a solid understanding of the numerical issues before we move on to incorporate more advanced perceptual error norms. For numerical comparisons, we use the *Signal to Noise Ratio (dB)*: $SNR_{dB}(X,Y) = 10log_{10}(\frac{||X||_2}{||X-Y||_2})$, a standard in numerical analysis and image processing.

For image comparisons and the creation of error landscapes we would ideally want to compare to the 'true' image. However, even for synthetic data sets, there is no analytical solution we can apply [KE04]. Hence, we are forced to compute our reference image numerically. We use Riemann integration which converges to the correct result in the context of our study when using a very high sampling frequency s, pipeline precision p, and data quantization q. As we choose to vary two parameters at a time, fixing the third parameter can sometimes manifest itself as masking of the SNR increase of the two parameters under investigation. Consider, for example, an experiment where p and q are varied with a fixed $s = 2$. Comparing to a reference rendered using $s = 100$ typically yields an error landscape with a truncated pyramidal shape, see Figure 1(a). This truncation stems from the much higher s in the reference, and indicates that further increase of either q or p is masked by the difference in s. Instead comparing to a reference image rendered with the same sampling frequency, i.e $s = 2$, an error landscape without truncation is obtained, see Figure 1(b). This second type of reference still utilizes a very high data quantization and pipeline precision. Similarly, for p vs. s scenar-

Dataset (d)	Volume resolution (l)	# Considered TFs (t)	Camera position	Pipeline precision (p) in bits	Data quantization (q) in bits	Sampling frequency (s) in steps/voxel
Marchner-Lobb	40x40x40	3	(1,1,2)	2-40	2-40	2
Marchner-Lobb	40x40x40	3	(1,1,2)	2-40	2-40	20
Shepp-Logan	128x128x128	2	(2, 0, 0)	2-40	2-40	2
CFD Jet	104x129x129	2	(0, 0, 1)	2-40	2-24	2
Mouse Embryo	449x663x449	2	(2, 0, 0)	2-40	2-24	2
Golden Lady	512x512x624	2	(2, 0, 0)	2-40	2-12	2

Table 1: *Scenarios considering relationship between input data quantization q and pipeline precision p.*

Dataset (d)	Volume resolution (l)	# Considered TFs (t)	Camera position	Pipeline precision (p) in bits	Data quantization (q) in bits	Sampling frequency (s) in steps/voxel
Marchner-Lobb	40x40x40	3	(1,1,2)	2-40	8	1-20
Marchner-Lobb	40x40x40	3	(1,1,2)	2-40	12	1-20
Shepp-Logan	128x128x128	2	(2, 0, 0)	2-40	8	1-20
CFD Jet	104x129x129	2	(1,0,0)	2-40	8	1-20
Hydrogen	128x128x128	3	(0,2,0)	2-40	8	1-20
Engine	256x256x256	2	(2,0,0)	2-40	8	1-20
Mouse	449x663x449	2	(2,0,0)	2-40	8	1-20
Golden Lady	512x512x624	2	(2,0,0)	2-40	8	1-20
Golden Lady	512x512x624	2	(2,0,0)	2-40	12	1-20

Table 2: *Scenarios considering relationship between sampling frequency s and pipeline precision p.*

ios we set up the reference image, such that our error landscape would converge for the specific q considered. To study the dependencies between p and q or s separately without masking effects suppressing the landscapes, we focus on the second type of reference image rendered with the masking parameter fixed at the level that was used in the experiment.

3.3. Arbitrary precision DVR pipeline

For our investigations we have implemented a volume renderer in which all calculations are performed with arbitrary floating-point precision. It is implemented on the CPU, and utilizes the multi-precision *GNU* libraries *GMP* 4.2.4 and *MPFR* 2.4.0 [FHL*07]. This allows us to vary the floating-point precision from 2 bits to thousands of bits at any stage of the pipeline. The framework also allows for varied precision of the input data and varied sampling frequencies. As the allocation and deallocation of arbitrary precision floating-point numbers are not natively supported by the hardware they are time-consuming operations. Rendering a $256 \times 256 \times 256$ volume sampled at 2 steps per voxel with a resolution of 250×250 pixels takes 2-3 minutes on a 2.4 Ghz Intel Core 2 Duo processor and 667 MHz DDR2 SDRAM.

3.4. Data sets and transfer functions

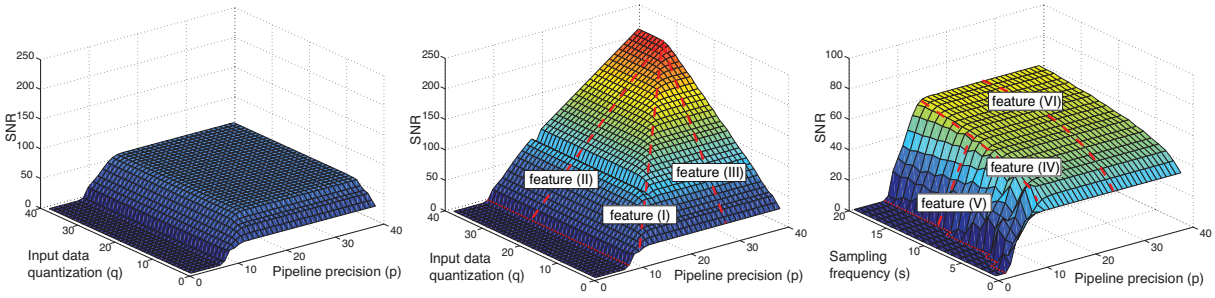
We consider two synthetic data sets: the *Marschner-Lobb test signal* [ML94], a data set with a well-defined band-limit, and the *Shepp-Logan MRI phantom* [SL74], a standard volume for assessing medical reconstruction algorithms. As an example of a simple real data set, we consider a simulation of a *hydrogen atom* given with 8-bit fixed-point precision. As examples from the engineering and medical domains, we consider two X-ray Computed Tomography (CT) scans: the *Engine* data set given in 8-bit fixed-point precision, and the higher resolution *Golden Lady* data set given at 12-bit fixed-point precision. To study the effects of increased data quantization for real data sets we consider the *Mouse Embryo* data set obtained from a set of 2D *Optical Projection X-Ray Tomography* scans, and the *CFD Jet* data set representing vorticity values from a simulation of turbulent flow, given in IEEE single precision, corresponding to 24 bit precision. To emulate fixed-point precision, we preprocess the data and quantize it with fixed-point spacing. The transfer functions used were chosen to be both realistic for common application scenarios as well as to explore the space of different frequency and opacity settings, and include both transfer functions with wide envelopes and low overall opacity (transparent-like setting), as well as higher frequency envelopes using higher opacity values (iso-surface-like setting). To explore the impact of lattice resolution, we have chosen data sets ranging from $40x40x40$ to $512x512x624$ voxels, see Tables 1 and 2. A detailed specification of the transfer functions and sample renderings can be found in the supplementary material.

4. Numerical results

For the experimental study we have rendered more than $35,000$ images. All images were rendered in greyscale with a 250×250 resolution. Details of each experiment, the generated error landscapes, and a corresponding experimental analysis can be found in the supplementary material.

4.1. Pipeline precision and input data quantization (p-q)

A typical error landscape for scenarios with varying pipeline precision, p, and data quantization, q, can be seen in Figure 1(b). Investigating the set of generated error landscapes reveals that they all exhibit a very similar topology. In this



(a) (b) (c)

Figure 1: *SNR, p-q landscape, where each image was generated using a fixed step size $s = 2$, and compared to a reference with:* ***a)*** *$s = 100$, $p = 100$ and $q = 100$, and* ***b)*** *$s = 2$, $p = 100$ and $q = 100$. **c)** A typical error landscape for scenarios with varying p and s using a fixed data quantization $q = 8$, and a reference with $s = 100$, $p = 100$ and $q = 8$.*

study, we have focused on three prominent features, denoted $(I), (II)$, and (III) in Figure 1(b).

- Feature (I) is the ridge that corresponds to the limit where, for a given data quantization, q, an increase in pipeline precision, p, will no longer improve the SNR, and vice versa.
- Feature (II) describes, for a fixed q, the gain in SNR as a function of p, until reaching feature (I).
- Feature (III) describes, for a fixed p, the gain in SNR as a function of q, until reaching feature (I).

By tracing the values of the generated error landscape, we extracted the features and analyzed their variation across experiments. Using a linear least squares fitting of the form $p = k * q + m$, the slope of Feature (I) for a general landscape can be approximated as $k \approx 1$ with an offset m along the p axis that varies in the range from 5 to 15 bits among the experiments as different data sets, transfer functions and sampling frequencies are used. The increase in SNR as a function of p or q (as represented by (II) and (III) respectively) has been well approximated by a linear least squares fitting in each scenario.

We have found that the average slope for feature (II) is fairly constant around $5-7$ db/bit with a small variation between scenarios (see Section 4.2). Note that each extra pipeline precision bit doubles the quantization levels of the resulting image representation, which in the ideal case yields an SNR increase of ≈ 6 db/bit [WK08]. We found that the slope of feature (II) is constant in each scenario for q above $7-8$ bits. Hence, the influence of q in this region is negligible. For lower quantization, the co-dependency between p and q is difficult to determine due to masking effects. Such low values of q are, however, not the focus of this paper, as most data use 8 bits or more. As can be seen in Figure 1(b), there is a flat area with zero SNR leading up to the start of feature (II). This is due to inaccuracies in the operation $(1-\alpha)$, performed both in composting and opacity correction. We have found that using a low opacity transfer function or a high sampling frequency shifts the start of feature (II) towards higher precision. For low frequency transfer functions, we have noticed that the slope of feature (II) is generally steeper than for high frequency transfer functions. This can clearly be observed when comparing the different transfer functions for the Marschner Lobb data set.

For feature (III), the variation in SNR as a function of q for fixed p, the slope is close to constant for p larger than the values suppressed by feature (I). The average slope over scenarios is ≈ 6.3 db/bit (represented by k_1 in Table 3). We have found no clear co-dependency between feature (III) and the sampling frequency used ($s = 2, 20$).

4.2. Pipeline precision vs. sampling frequency (p-s)

Figure 1(c) displays a typical SNR error landscape for scenarios studying the co-dependency between pipeline precision p and sampling frequency s. The topology of the set of generated error landscapes is very similar, and three prominent features can be noted:

- Feature(IV) is the ridge, where, for a fixed s, an increase in p has no further effect on the SNR, and vice versa.
- Feature (V) is the gain in SNR with higher p for a fixed s, until reaching feature (IV)
- Feature (VI) is the gain in SNR with higher s for a fixed p, until reaching feature (IV) or the end of the landscape.

Feature (V) starts at the end of the flat zero signal SNR region introduced at low p. It can be observed that this starting position varies with s, and that this variation follows a staircase behavior with increasing step length. Feature (V) is close to linear for a fixed s, but the parameters for a linear least squares fitting varies for different s. Note, that this behavior was also noticed for feature (II), which corresponds to feature (V) for a fixed sampling frequency. This shows that there exists a co-dependance between s and p for both feature (II) and feature (V). We have also noted that the co-dependency between p and s for feature (V) is more emphasized for high resolution volumes, that is when l is large. This conforms to the theory, see Section 3, since the number of samples given by $s \cdot l$ affects the precision require-

ments during compositing. The observed numerical values for both the slope and co-dependency effects are discussed more closely in Section 5.

Feature (VI) clearly exhibits a non-linear behavior, and varies between scenarios. We have found that this feature gives especially large errors for combinations of both high data frequency and high transfer function frequency (sharper envelopes), for example the Marschner Lobb data set with the isosurface like transfer function. However, for the scope of this paper, the experimental results have not provided us with all the details needed for an exact understanding of the complex effects of transfer functions and data sets on the sampling frequency. We did not, for example, find clear evidence for the essential band-limiting frequencies in our error landscapes as discussed in previous work [BMWM06]. We have, however, found that the effect of changing s for each pipeline precision, p, can, with sufficient accuracy for our investigation, be approximated by a quadratic function. In Section 5 and the supplementary material, the parameters for this quadratic approximation are presented in detail.

The extracted ridge, feature (IV), exhibits a non-linear slope with a small variation between the experiments. This is due to the non-linear nature of feature (VI) and co-dependency between p and s in the region of feature (V). We observed a similar behaviour no matter whether the quantization of the input was 8 or 12 bit leading us to believe that increasing q has a negligible effect on the relationship between s and p.

5. Model building

Given an error landscape for a specific scenario a user is able to pick suitable rendering parameters for achieving a particular image fidelity. However, the creation of error landscapes is impractical and time-consuming. Our goal in this section is to create a set of predictive models, that require few or no previously rendered images to accurately estimate the error landscape for arbitrary real application scenarios.

From the numerical results presented in Section 4, a number of conclusions can be drawn: The overall topology of all generated error landscapes is similar for both the p vs. q and p vs. s scenarios, respectively. We have identified a number of *features*, that define these topologies. We conclude that the ridges, features (I) and (IV), are a direct consequence of feature masking, that is one parameter effect dominating the others. These masking effects allow us to model feature $(II), (III), (V)$, and (VI) separately, and base our model on the minimum of two functions either describing features (II) and (III) or (V) and (VI). For features (II) and (V) there is a co-dependence between pipeline precision, p, and sampling frequency, s, which is especially apparent for large data sets. Here, we denote p vs. q error landscapes, with fixed sampling frequency, s_f, and lattice resolution, l_f, as $SNR_{pq}(p,q)$, and p vs. s landscapes as $SNR_{ps}(p,s)$. The error landscapes $SNR_{pq}(p,q)$ and $SNR_{ps}(p,s)$ can be modeled as:

$$SNR_{pq}(p,q) = \max(0, \min(SNR_{III}(q), SNR_{II,V}(p,s_f,l_f)))$$
$$SNR_{ps}(p,s) = \max(0, \min(SNR_{VI}(s), SNR_{II,V}(p,s,l_f)))$$
$$SNR_{III}(q) = k_1 q + m_1 \quad (4)$$
$$SNR_{VI}(s) = k_3 s^2 + k_4 s + m_3 \quad (5)$$
$$SNR_{II,V}(p,s,l) = k_2(s,l)p + m_2(s,l) \quad (6)$$

where $SNR_{III}(q)$ describes feature (III) as a linear function of q, $SNR_{VI}(s)$, feature (VI) as a quadratic function of s, $SNR_{II,V}(p,s,l)$ feature (II) as a linear function of p, while keeping s fixed, and $SNR_{II,V}(p,s,l)$ feature (V) as a linear function with coefficients that change when s is varied. Since both theoretical and numerical results indicate that the co-dependency between p and s is due to the number of samples composited $(s \cdot l)$, we model the variation of feature (V) as functions of both s and l. Specifically we model the co-dependency by linear functions as $k_2(s,l) = k_{k2} \cdot s \cdot l + m_{k2}$ and $m_2(s,l) = k_{m2} \cdot s \cdot l + m_{m2}$.

The ridges ((I) and (IV)) describe when for a given q or s the increase of the pipeline precision no longer has an effect on image fidelity, and vice versa. These ridges can be modeled directly by considering where the functions (6) and (4) or (6) and (5) are equal. Specifically we can describe feature (I) with $p_{(I)}(q,s_f,l_f)$, by considering where $SNR_{III}(q) = SNR_{II,V}(p,s_f,l_f)$. Similarly we describe feature (IV), with $p_{(IV)}(s,l_f)$, by considering where $SNR_{VI}(s) = SNR_{II,V}(p,s,l_f)$. That is

$$p_{(I)} = \frac{k_1}{k_2(s_f,l_f)} \cdot q + \frac{m_1 - m_2(s_f,l_f)}{k_2(s_f,l_f)}$$
$$p_{(IV)} = \frac{k_3}{k_2(s,l_f)} \cdot s^2 + \frac{k_4}{k_2(s,l_f)} \cdot s + \frac{m_3 - m_2(s,l_f)}{k_2(s,l_f)}$$

Mean model fitting - To estimate the values of the model parameters $k_{1\ldots4}$ and $m_{1\ldots3}$, the simplest option is to average the extracted parameters over all training scenarios. In order to avoid unnecessary bias, we do not consider the scenarios where the third parameter is varied. For example for the Marschner-Lobb data set we only consider the scenario of 8 bit data and not 12 bit data, since the additional data bits have been found not to affect the SNR_{ps} significantly, see Section 4. Table 3 displays the mean, standard deviation, max and min for the extracted parameters from the conducted training scenarios (see Table 1 and 2). The mean model parameter fitting has the disadvantage of not considering the variation of parameters for different transfer functions and data sets. Such effects are very hard to model explicitly, since the parameter space of possible transfer functions and data sets does not have a reasonable parameterization. Hence, we turn instead to adaptive models, that fit the model parameters to specific data sets and transfer functions by rendering and comparing a few specific images.

Instance based model fitting - Our *instance based model* adapts itself to a specific scenario by directly sampling the

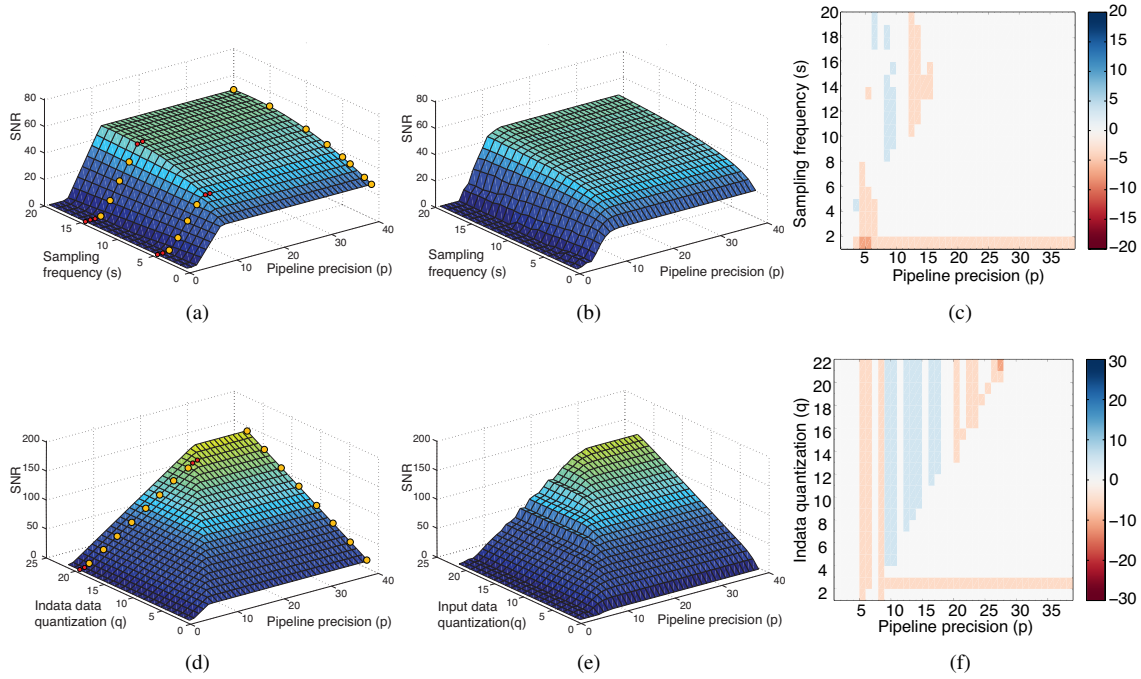


Figure 2: *The* CFD Vort *validation data set.* ***Top row:*** *SNR_{ps} landscape for pipeline precision p vs. sampling frequency s.* ***a)*** *instance based model estimation, using 25 rendered images for model adaption, red dots indicate images for tracing start of feature (V) and the ridge, orange dots indicate the sampling of the feature (V) and (VI) parameters.* ***b)*** *generated validation set where each point in the error landscape is rendered.* ***c)*** *error plot of difference between estimation a) and validation b) data in dB.* ***Bottom row:*** *SNR_{pq} for pipeline precision p vs. input data quantization q.* ***d)*** *instance based model estimation, using 20 rendered images for model adaption, red dots indicate images for tracing start of feature (II) and the ridge, orange dots indicate the sampling of the feature (II) and (III) parameters.* ***e)*** *generated validation data set where each point in the error landscape is rendered.* ***f)*** *error plot of difference between d) and e) in dB.*

parameter fitting for $k_{1\ldots4}$ and $m_{1\ldots3}$. To adapt the model parameters to a specific scenario, given by d, l, and t, a number of images need to be rendered and compared to a reference.

1. To estimate the linear effect of q, feature (I), one needs to render at least two images for large enough p to be unaffected by masking, and compare them to a reference image using the highest obtainable q.
2. To estimate the quadratic effect of s, feature (VI), at least three images need to be rendered, for large enough p to be unaffected by the masking effects below the ridge.
3. To estimate the direct effects of p, one can either model it as a linear increase if an SNR_{pq} landscape is desired or a linear function with linear co-dependency effects if an SNR_{ps} landscape is sought.

   SNR_{pq}**:** Two images at a value for $q > 8$ above the start of feature (II) is the minimal requirement. To find the start of feature (II), one can render images with increasing p until a change is detected.

   SNR_{ps}**:** At least two linear fittings of feature (V) for different s have to be extracted. The image samples for each fixed s should be placed after the start of feature (V) and before the ridge, feature (IV). The start of feature (V) and the ridge can be found by tracing the landscape using several images with a fixed s.

Feature	Parameter	mean	std. dev.	max	min
(III)	k_1	6.34	0.591	7.41	5.90
(III)	m_1	-1.53	13.7	25.8	-17.8
(II), (V)	$k_{k_2} * 10^4$	6.96	8.26	31.7	0.385
(II), (V)	k_{m_2}	4.92	1.44	7.75	3.28
(II), (V)	$m_{k_2} * 10^2$	-1.13	1.39	-0.213	-5.87
(II), (V)	m_{m_2}	-19.8	11.7	-8.43	-53.1
(VI)	k_3	-0.106	0.0434	-0.0259	-0.163
(VI)	k_4	3.50	1.64	5.86	0.666
(VI)	m_3	34.0	16.6	64.5	3.13

Table 3: *Mean, standard deviation, maximum and minimum over extracted parameters from the training scenarios.*

Some of the SNR landscapes exhibit noise, especially in the region dominated by feature (V). In this case additional images can lower the influence of noise on the model estimation. For highly anisotropic data sets, where the data signal varies with viewpoint, the model can be improved by rendering the volume from a few different viewpoints and either use a mean or a pessimistic estimate over the obtained error landscapes.

Hybrid model fitting - The drawback of the instance

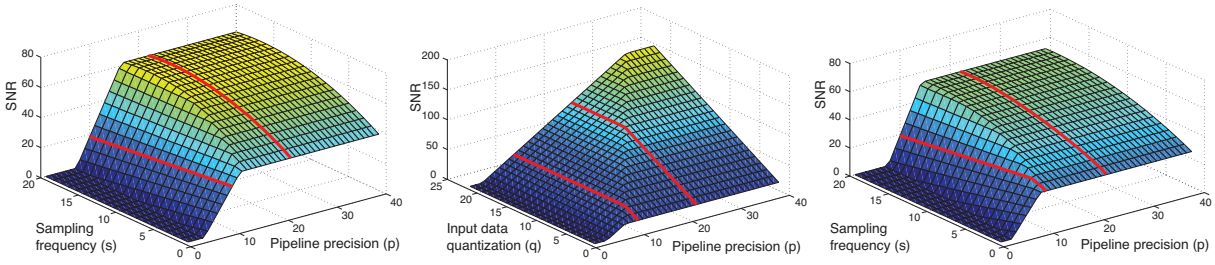


Figure 3: *For the* CFD Vort *validation data set. Mean based model for* ***a):*** SNR_{ps} *landscape. and* ***b):*** SNR_{pq} *landscape.* ***c):*** *Hybrid based* SNR_{ps} *landscape. The red lines at* $p = 11$ *and* $p = 24$ *corresponding to IEEE Half and Single (float) precision.*

	SNR_{ps} Landscapes							SNR_{pq} Landscapes				
	Instance model			Hybrid model		Mean model		Instance model			Mean model	
Dataset/TF	mean	std. dev.	#img	mean	std. dev.	mean	std. dev.	mean	std. dev.	#img	mean	std. dev.
MRI/TF1	1.12	1.00	23	1.73	2.21	34.7	18.5	2.45	2.43	11	6.45	2.40
MRI/TF2	1.23	1.08	23	2.87	3.54	42.2	21.2	1.85	2.31	11	1.85	1.93
Vort/TF1	1.16	0.86	24	3.74	3.58	34.2	15.7	2.19	1.42	19	4.39	4.94
Vort/TF2	1.77	2.32	28	5.88	4.72	43.1	21.5	1.68	1.98	20	19.2	8.64
Carp/TF1	1.36	1.27	26	2.37	2.93	30.8	18.4	1.31	1.85	13	11.2	4.37
Carp/TF2	1.22	1.08	25	2.33	3.14	31.8	18.4	3.76	2.72	12	10.3	4.21

Table 4: *Mean error in dB, standard deviation (std. dev.) and number of rendered images (#img) for model estimation of validation scenarios. For the Hybrid model 9 images have been rendered with varied setting for* s *at* $p = 24$ *for each validation scenario.*

model is the large number of images that need to be rendered using an arbitrary precision pipeline. It can be observed that the gain in SNR over the region dominated by feature (VI), and thus also the ridge represented by feature (IV), varies noticeably between different scenarios. On the other hand, feature (V) exhibits less variation between scenarios. To improve the mean based fitting for SNR_{ps} landscapes, we therefore propose a *hybrid model*. For this hybrid model the parameters $k_{3\ldots4}$ and m_3 for SNR_{VI} are adapted to the data and transfer function under investigation in the same way as for the instance based model. However, feature (V) is based on the mean parameters. Note that the reference images in this case are rendered with high sampling frequency, but at the same pipeline precision as the rendered images. The hybrid model thus has the advantage that it does not require an arbitrary precision pipeline. Instead only a standard pipeline with the possibility to vary s is necessary.

5.1. Model validation

To validate our models, we have used three additional data sets: the *Carp* data set obtained from a CT scan given in 12 bit fixed-point quantization and the *MRI Head* a MRI scan given in 12 bit fixed-point quantization. To validate our model on higher precision data we have used the *CFD Vort*, a CFD simulation data set given in 24 bit floating-point precision. For each data set we consider two transfer functions, one with a single higher frequency envelope and one with lower frequency envelopes.

For the *CFD Vort* data set, Figure 2(b) depicts a densely sampled (true) SNR_{ps} landscape where each SNR measure is based on a rendered image. Similarly Figure 2(e) depicts a densely sampled SNR_{pq} landscape. Examples of the SNR_{ps} and SNR_{pq} generated from the mean based model are displayed in Figures 3(a) and 3(b). From a comparison with Figures 2(b) and 2(e), the model captures the overall topology but lacks some of the accuracy of the true landscapes. Figure 2 displays an example of two instance based SNR_{pq} and SNR_{ps} landscapes. Figure 2(a) shows the estimated SNR_{ps}, and 2(c) an error plot of the difference between 2(b) and 2(a). Similarly, Figure 2(d) displays the estimated SNR_{pq} landscape, and 2(f) an error plot of the difference between 2(e) and 2(d). Figure 3(c) displays a hybrid fitting of the SNR_{ps} landscape.

Table 4 displays the error mean and standard deviation for the three validation data sets for the *instance, mean* and *hybrid* based parameter fittings. The error for each point in the landscapes is computed as the absolute value of the difference in SNR between the model landscape and the true landscape, where we do not consider errors in SNR_{pq} landscapes where $q < 8$. Note that most model errors are small compared to the increase in SNR gained from increasing p, q and s. A complete overview of all model errors for all conducted scenarios can be found in the supplementary material.

6. Visual artifacts

The greyscale images used in the numerical investigations and a set of rendered color images with various data sets and transfer function settings were visually inspected. One of the most common visual artifacts introduced by a low pipeline precision are quantization errors or *banding* arti-

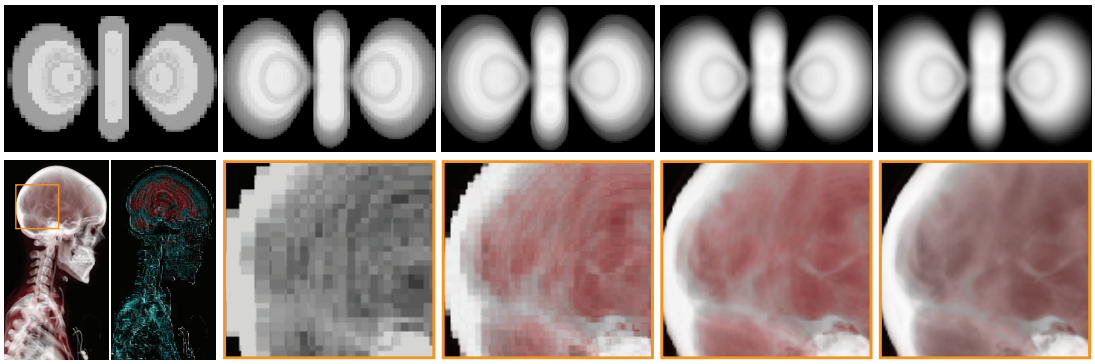


Figure 4: ***Upper row:*** *Greyscale images of Hydrogen atom rendered using* $6,7,8,9$, *and* 10 *bits of pipeline precision at two steps/voxel.* ***Bottom row:*** *Color images of Golden Lady rendered with a 512x512 pixel resolution using two steps/voxel, From the left: Rendered using 10 bits of pipeline precision, the normalized difference image between 8 and 12 bits and close ups of images using* $7,8,9$, *and* 10 *bits of precision*

facts in the images. With increased precision these bands shrink until eventually, an apparently continuos change in intensity/color is reached. This artifact can be seen in the upper row of Figure 4, which displays the hydrogen atom rendered using $6,7,8,9$, and 10 bits of pipeline precision. These effects also have a co-dependence with sampling frequency, where higher sampling frequencies tend to increase the imprint of the artifacts, this corresponds to the observed co-dependency in the error landscapes between sampling frequency and pipeline precision, also modeled in Section 5. Another artifact is related to the transparency of different envelopes. High transparency materials require higher precision to be rendered correctly. For example, when rendering the Golden Lady, the bone structures are rendered correctly for lower precisions while the skin and blood structures (having a lower assigned opacity) require a higher precision. In Figure 4 (bottom row) the difference between 7 and 8 bits of pipeline precision is shown. Examining the normalized difference image between precision settings, see Figure 4, edges and contours in the volume are often emphasized. We have also found that artifacts due to a limited pipeline precision are often very similar to so called *wood-grain* artifacts [EHK*06], assumed to be introduced by low sampling frequencies. This can e.g. be seen in Figure 4 (bottom row), in which the Golden Lady data set is rendered with varied pipeline precision. We have also noticed that this type of artifact is more noticeable when high pipeline precision is used for the computation of ray sample locations and the rest of the pipeline uses low precision.

7. Conclusion and future work

We have found that, generally, image quality improves by $\approx 6db/bit$ increased pipeline precision with a small variation for different sampling frequencies, volume resolutions, data sets and transfer functions. By studying the co-dependency effects with sampling frequency and data quantization, we found that there exists a limit, above which increasing the pipeline precision further does not increase the SNR significantly. Using our models, we can predict where such ridges occur. Given such a predictive model, a practitioner could e.g. set sampling distance, given an input data quantization while constrained to a particular pipeline precision. Based on the mean error landscapes, one can conclude that the standard IEEE half and single precision formats, see Figures 3(a) and 3(b), give acceptable numerical quality ($SNR \gtrsim 40$) for reasonable sampling frequencies and data quantizations. However, if very high sampling frequencies are used, IEEE single precision may not be enough, since there is a clear increase in precision requirements with higher sampling frequencies. For the non-truncated landscapes, Figures 3(a) and 3(b), an upper SNR value is given according to a fixed third parameter. The true SNR value (corresponding to a truncated landscape) is, however, given by the minimum SNR value from the two types of model landscapes. This corresponds to a 3D model, obtained by considering the min over the three features (4)(5)(6) presented in section 5. We can also conclude that for visualization tasks not requiring absolute accuracy, or given requirements for coarse sampling frequencies and/or data quantizations, half precision, corresponding to 11 mantissa bits, should be sufficient as the error introduced by pipeline precision is negligible compared to the low sampling frequency and/or coarse data quantization.

The paths to explore in future work include the effect of a limited precision in other parts of the pipeline such as higher order interpolation filters, shading methods etc. This study will furthermore be extended to include perceptual error metrics. We will also further explore the detailed effects of varying the sampling frequency using different data sets and transfer functions.

8. Acknowledgements

The authors would like to thank Matt Cooper and Claes Lundström for valuable discussions and proofreading. This work has been supported by the Swedish Research Council under the Linnaeus Center CADICS, and by the Strategic Research Center MOVIII, funded by the Swedish Foundation for Strategic Research, SSF, as well as the Natural Science and Engineering Research Council of Canada.

References

[BMWM06] BERGNER S., MÖLLER T., WEISKOPF D., MURAKI D. J.: A spectral analysis of function composition and its implications for sampling in direct volume visualization. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (Sep.-Oct 2006), 1353–1360. 2, 6

[BNMK04] BITTER I., NEOPHYTOU N., MUELLER K., KAUFMAN A. E.: Squeeze: numerical-precision-optimized volume rendering. In *Graphics Hardware* (2004), pp. 25–34. 2, 3

[BO04] BABUSKA I., ODEN J.: Verification and validation in computational engineering and science: basic concepts. *Computer Methods in Applied Mechanics and Engineering 193*, 36-38 (2004), 4057–4066. 2

[EHK*06] ENGEL K., HADWINGER M., KNISS J. M., REZK-SALAMA C., WEISKOPF D.: *Real-Time Volume Graphics*. A K Peters, Ltd, 2006. 2, 9

[FHL*07] FOUSSE L., HANROT G., LEFÈVRE V., PÉLISSIER P., ZIMMERMANN P.: MPFR: A multiple-precision binary floating-point library with correct rounding. *ACM Transactions on Mathematical Software 33*, 2 (June 2007), 13:1–13:15. 4

[GMS*07] GIESEN J., MUELLER K., SCHUBERTH E., WANG L., ZOLLIKER P.: Conjoint analysis to measure the perceived quality in volume rendering. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1664–1671. 2, 3

[Gol91] GOLDBERG D.: What every computer scientist should know about floating point arithmetic. *ACM Computing Surveys 23*, 1 (1991), 5–48. 2

[Hig02] HIGHAM N. J.: *Accuracy and stability of numerical algorithms*, second edition ed. SIAM, 2002. 1, 2

[JS03] JOHNSON C. R., SANDERSON A. R.: A next step: Visualizing errors and uncertainty. *IEEE Comput. Graph. Appl. 23*, 5 (2003), 6–10. 1

[KE04] KRAUS M., ERTL T.: Pre-integrated volume rendering. In *The Visualization Handbook*, Hansen C., Johnson C., (Eds.). Academic Press, 2004. 2, 3

[KWP01] KIM K., WITTENBRINK C. M., PANG A.: Extended specifications and test data sets for data level comparisions of direct volume rendering algorithms. *IEEE Transactions on Visualization and Computer Graphics 7*, 4 (2001), 299–317. 2

[Max95] MAX N.: Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics 1*, 2 (June 1995), 99–108. 2

[MHB*00] MEISSNER M., HUANG J., BARTZ D., MUELLER K., CRAWFIS R.: A practical comparison of popular volume rendering algorithms. In *Proceedings of the 2000 IEEE Symposium on Volume Visualization* (2000), ACM, pp. 81–90. 2

[ML94] MARSCHNER S. R., LOBB R. J.: An evaluation of reconstruction filters for volume rendering. In *Proceedings of the Conference on Visualization '94* (1994), IEEE Computer Society Press, pp. 100–107. 2, 4

[MMMY97a] MÖLLER T., MACHIRAJU R., MUELLER K., YAGEL R.: A comparison of normal estimation schemes. In *Proceedings of the IEEE Conference on Visualization* (Oct. 1997), pp. 19–26. 2

[MMMY97b] MÖLLER T., MACHIRAJU R., MUELLER K., YAGEL R.: Evaluation and design of filters using a Taylor series expansion. *IEEE Transactions on Visualization & Computer Graphics 3*, 2 (1997), 184–199. 2

[NA92] NOVINS K., ARVO J.: Controlled precision volume integration. In *Proceedings of the 1992 Workshop on Volume Visualization* (New York, NY, USA, 1992), ACM, pp. 83–89. 2

[PSC00] PAPPAS T., SAFRANEK R., CHEN J.: Perceptual criteria for image quality evaluation. *Handbook of image and video processing* (2000), 669–684. 3

[SL74] SHEPP L. A., LOGAN B. F.: The Fourier reconstruction of a head section. *IEEE Transactions on Nuclear Science 21*, 3 (June 1974), 21–43. 4

[WK08] WIDROW B., KOLLLAR I.: *Quantization Noise*. Cambridge University Press, 2008. 2, 5

[WMG98] WITTENBRINK C. M., MALZBENDER T., GROSS M. E.: Opacity-weighted color interpolation for volume sampling. In *Symposium on Volume Visualization* (1998), pp. 135–142. 2

[WU99] WILLIAMS P. L., USELTON S. P.: Metrics and generation specifications for comparing volume-rendered images. *The Journal of Visualization and Computer Animation 10* (1999), 159–179. 2

[YNCP06] YUAN X., NGUYEN M. X., CHEN B., PORTER D. H.: HDR VolVis: High dynamic range volume visualization. *IEEE Transactions on Visualization and Computer Graphics 12*, 4 (2006), 433–445. 1