

# Bandwidth-Efficient Parallel Visualization for Mobile Devices

Andreas Helfrich-Schkarbanenko, Vincent Heuveline, Roman Reiner, Sebastian Ritterbusch  
 Engineering Mathematics and Computing Lab (EMCL)

Karlsruhe Institute of Technology (KIT)  
 Karlsruhe, Germany

{andreas.helfrich-schkarbanenko, vincent.heuveline, roman.reiner, sebastian.ritterbusch}@kit.edu

**Abstract**—For visual analysis of large numerical simulations on mobile devices, we introduce a remote parallelizable visualization method for low-bandwidth and high-latency networks. Based on a mathematical model for multi-layered planar impostor representation of arbitrary complex and unbounded scenes, we derive optimal impostor placement from a derived metric. Using stochastic usage models, we prove the optimal bandwidth consumption order for choosing corresponding viewport impostor sets, leading to bandwidth-efficient remote visualization concepts for high performance computing simulation results.

**Keywords**-Remote Visualization; Mobile Visualization; Optimal Impostor Placement.

## I. INTRODUCTION

Remote visualization is vital wherever local storage, data transfer rates or graphical capabilities are limited. Even though the capabilities of modern smartphones are increasing rapidly, many desirable applications are impeded by limitations of the current hardware [1].

Image-based rendering techniques [2] are widely used to reduce the geometric complexity of virtual environments by replacing parts of a scene with a textured representation approximating the original geometry. Since these so-called *impostors* have a significantly simplified geometry, parallax errors [3] occur when rendering the approximation. An impostor is generated for an initial *viewport* (that is, a position and viewing direction) and is said to be *valid* as long as the visual difference to the (hypothetically rendered) original geometry is below a certain threshold.

In our application, these impostors are rendered remotely on render servers and streamed to a mobile device where they are used to approximate the scene. One substantial advantage of the impostor approach [4] is that the render time on the device only depends on the number of impostors and the resolution of the textures, not on the amount of data they display. As long as servers can generate and transfer the impostor textures sufficiently fast, every scene can be displayed remotely, regardless of its actual complexity. In this setting, network bandwidth is the bottleneck and a careful analysis of bandwidth consumption becomes mandatory.

We develop a mathematical model that allows to quantify the display error and propose an approximation method that proves to be optimal with respect to the derived error metric.

We can show that our method significantly reduces the total amount of image data that needs to be transferred. The key aspects of our method are illustrated in Figure 1: In this simplified two-dimensional case, a traditional remote visualization using one layer would need at least 32 images to provide the same visual accuracy as one layer set of 5 images. This effect is amplified by each additional degree of freedom of the viewer.

In the following Section II, we discuss related work. Then we introduce the underlying mathematical model in Section III, on which we derive the fundamental error metrics. In Section IV, this leads us to the optimal impostor placement and directly corresponding bounds for the visualization error of one impostor set. The practical outcome of the findings, using as many impostor sets as needed, is proven and evaluated in Section V, which is leading us to the conclusions in Section VI.

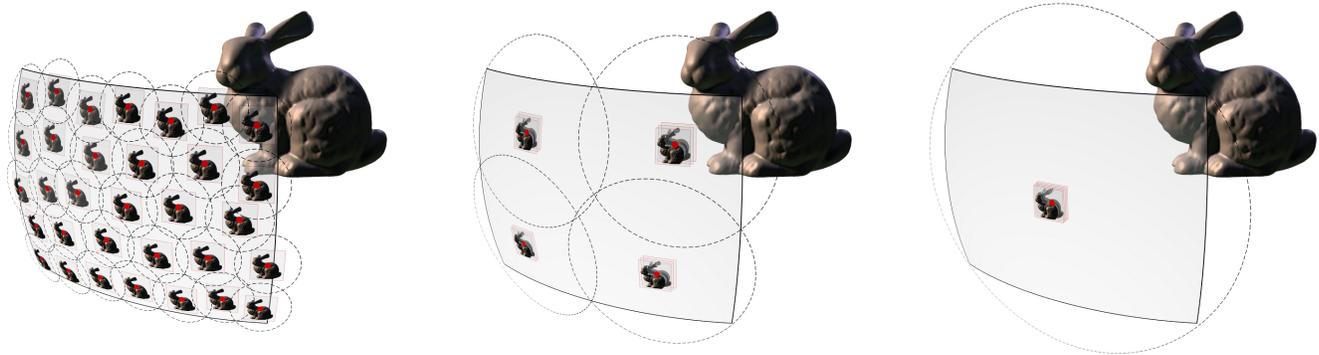
## II. RELATED WORK

A variety of image-based rendering techniques are reviewed in [4] and [2]. The first paper focuses mainly on techniques using planar impostors but also mentions more exotic approaches like depth images (planar impostors with per-pixel depth information) and light fields. These and other techniques, such as view morphing and view dependent textures, are examined in more detail in the second paper.

In the majority of cases, planar impostors stacked with increasing distance to the observer are used (see [3], [5], [6]), usually to approximate distant parts of the scene or single objects. In contrast, our approach uses impostors to represent the full scene.

For large objects, different parts of continuous surfaces can end up on different impostors which makes them tear apart when viewed from a shallow angle. Avoiding this particular problem was one focus of the method developed in [3]. Another interesting use of planar impostors is [7], which treats the rendering of volume data on mobile phones.

Several approaches using geometrically more complex impostors can be found in [6], [8] and [9]. In [4], so-called *billboard clouds* are used to approximate the shape of an object using several intersecting planar impostors. While the impostor creation process for this approach is quite



(a) 32 impostor sets with one layer each

(b) Four impostor sets with three layers each

(c) One impostor set with five layers

Figure 1. An impostor representation is only valid inside a small region around the initial viewport for which it was originally created. For observer viewports within this validity region (indicated by the dotted line) the display error does not exceed a given maximum value. To faithfully approximate the scene for all observer viewports inside the shaded area, several impostor sets have to be transmitted.

The validity regions can be enlarged (while keeping the maximum error unaltered) by increasing the number of layers per impostor set. As the number of required impostor sets decreases faster than the number of layers per set increases, this significantly reduces the total number of layers needed to approximate the scene to a given accuracy .

costly, the result allows examination from different viewing directions.

A very current example is Street Slide [10]. Street Slide sticks photos of front facades of urban environments to “panorama strips” that can be browsed by sliding sideways.

The need for accurate analysis of bandwidth and accuracy estimates is discussed in [4], [5], without further specifying how to choose which viewports to load. A more in-depth analysis on the subject of pre-fetching is given in [11] and [12]. The former defines a so-called benefit integral, indicating which parts of the scene – quality-wise – contribute most to the final image, the latter deals with rendering an indoor scene remotely. The task of remote rendering on mobile devices is addressed in [13] and [14], which mostly focuses on the technical aspects of the server-client communication.

Usually, depending on the complexity of the approximation, an impostor is either easy to generate but only valid inside a small region and thus needs to be updated very often, or it is valid inside a large domain but complex and difficult to generate and display [2]. Since the former strains bandwidth and the latter strains render speed, any image-based rendering approach is usually a trade-off between these limiting factors.

### III. VISUALIZATION MODEL AND ERROR METRICS

To begin with, a mathematical model describing viewports and projections thereon needs to be established, with which the rendering and approximation processes can be described. This yields an error function describing the maximum parallax error of a scene as a function of the observer movement, called *domain error*.

Finally, modelling the observer movement as a probability distribution, we can describe the expected value of this error.

This *interaction error* will be the cost function that we intend to minimize.

#### A. Perspective projection

Using homogeneous coordinates and projective transformations [15], we can express perspective projection as a  $4 \times 4$  matrix multiplication on the projective space  $\mathbb{P}^3$ :

**Definition 1.** The perspective projection onto the plane  $x_3 = d$  towards the origin is a function

$$\pi_d : \begin{cases} \mathbb{P}^3 \setminus \{(0, 0, 0, 1)^\top\} & \longrightarrow & \mathbb{P}^3 \\ x & \longmapsto & P_d x \end{cases}$$

with the parameter  $d > 0$  defining the proximity of the projection plane.

From the intercept theorems, one can easily see that the perspective projection of a point  $v = (v_1, v_2, v_3)^\top \in \mathbb{R}^3$ ,  $v_3 \neq 0$  onto the plane  $x_3 = d$  is given by  $(\frac{d}{v_3}v_1, \frac{d}{v_3}v_2, d)^\top$  which, using homogeneous coordinates, equals  $(v_1, v_2, v_3, \frac{v_3}{d})^\top$ . This yields the projection matrix

$$P_d := \left( \begin{array}{ccc|c} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{array} \right).$$

#### B. Viewports

Any viewport can be described by five values  $c_1, c_2, c_3 \in \mathbb{R}$ ,  $\vartheta \in [-\pi/2, \pi/2]$ ,  $\varphi \in [-\pi, \pi]$ , defining an affine transformation  $\chi$ , which is the combination of a translation by the vector  $(c_1, c_2, c_3)^\top$  followed by a rotation around the  $x_1$ -axis with the angle  $\vartheta$  and a rotation around the  $x_2$ -axis with the angle  $\varphi$  (cf. Figure 2). Actually, there is a sixth value which represents a rotation around the viewing direction.

Such a rotation, however, does not change the image besides rotating it. We assume the rotation to be lossless, which is why we do not need it for our purposes.

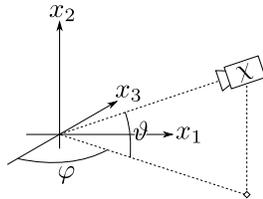


Figure 2. The angles  $\varphi$  and  $\vartheta$  of a viewport  $\chi$

We condense all five values into a single vector  $c := (c_1, c_2, c_3, \vartheta, \varphi)^\top$ . When describing viewports, we will use this vector  $c$  and the associated transformation  $\chi_c$  interchangeably. In particular, we will identify sets of viewports with subsets of  $\mathbb{R}^5$ :

**Definition 2.** The set

$$X := \mathbb{R}^3 \times [-\pi/2, \pi/2] \times [-\pi, \pi) \subset \mathbb{R}^5$$

will be called the *viewport set*. For all practical purposes, however, we want to restrict to viewports inside a given set of *feasible viewports*  $\Lambda \subset X$ .

Projective matrix representations of  $\chi_c$  and its inverse are

$$Q_c = \left( \begin{array}{c|c} B_{\vartheta, \varphi} & B_{\vartheta, \varphi} c \\ \hline 0 & 1 \end{array} \right) \quad \text{and} \quad Q_c^{-1} = \left( \begin{array}{c|c} B_{\vartheta, \varphi}^\top & -c \\ \hline 0 & 1 \end{array} \right)$$

where

$$B_{\vartheta, \varphi} := \begin{pmatrix} \cos \varphi & -\sin \varphi \sin \vartheta & -\sin \varphi \cos \vartheta \\ 0 & \cos \vartheta & -\sin \vartheta \\ \sin \varphi & \cos \varphi \sin \vartheta & \cos \varphi \cos \vartheta \end{pmatrix}.$$

We can now calculate a matrix representation of a projection onto an arbitrary viewport, by combining the matrices above with the matrix representations of the default projection  $\pi_d$ .

**Definition 3.** Let  $\chi$  be a viewport with an associated matrix representation  $Q$  and let  $\pi_\chi$  denote a projection onto the viewport  $\chi$ . Then, a matrix representation of  $\pi_\chi$  is given by  $P_{\chi, d} = Q P_d Q^{-1}$ , where  $P_d$  is the perspective projection matrix defined in Definition 1.

### C. Rendering process

Let renderable objects be located in a domain  $\Omega$ . We aim to simplify the scene by dividing  $\Omega$  into  $m$  disjoint parts  $\Omega_i$  called *cells*, replacing each with a planar representation of their contained objects. These so-called *impostors* will be created for the same initial viewport(s), that is, for a certain viewport we will create an *impostor set* with one impostor per cell, all for that particular viewport. This will be done

for  $n$  initial viewports resulting in  $n$  impostor sets with  $m$  impostors each.

As long as the current viewport matches the initial viewport for which the impostors have been created, the impostor representation coincides with the image of the actual scene. Changing the viewport, however, will introduce parallax errors, since depth information is lost in the impostor creation process.

To determine this error, we will first regard a single cell  $\Omega_i$  and a single vertex  $v \in \Omega_i$ . For a fixed initial viewport  $\chi_1$  we calculate the impostor representation  $\bar{v}$  of the actual point  $v$ . Then we consider a variable viewport  $\chi$  and calculate the screen coordinates  $v'$  of  $v$  and  $\bar{v}'$  of  $\bar{v}$  as functions of the viewports  $\chi$  and  $\chi_1$  (cf. Figure 3).

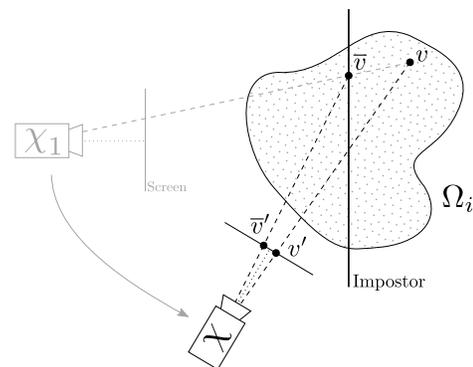


Figure 3. Rendering process for changed viewport

### D. The domain error

If we reiterate the procedure above, we obtain two images for each point in  $\Omega$ : one image of itself ( $v'$ , depending on  $\chi$ ) and one of its impostor representation ( $\bar{v}'$ , depending on both  $\chi$  and  $\chi_1$ ). The screen distance of these two, measured in (sub-)pixels is called the *screen space error*. As we are not interested in the error of a single point, but rather in error functions expressing the error of the entire scene, for example the mean error or the maximum error, we aggregate the screen space error over all point in  $\Omega$ . As the distribution of vertices inside  $\Omega$  is supposed to be unknown, we assume a uniform distribution and integrate the screen space error over the entire domain  $\Omega$ . We will be using the maximum error which replaces the integral with a supremum.

**Definition 4.** Denote the number of cells with  $m$ . For an initial viewport  $\chi_1$  we define the *domain error*

$$\begin{aligned} D(\chi, \chi_1) &:= \sup_{v \in \Omega} \|v'(\chi) - \bar{v}'(\chi, \chi_1)\|_2 \\ &= \max_{0 \leq i \leq m} \left\{ \sup_{v \in \Omega_i} \|v'(\chi) - \bar{v}'(\chi, \chi_1)\|_2 \right\}. \end{aligned}$$

This domain error depends on a variable observer viewport  $\chi$  and the fixed viewport  $\chi_1$ , for which the displayed impostor set was initially created. The dependence on  $\chi$

implies that we cannot evaluate our impostor approximation without knowledge of the observer movement. Clearly, we want to optimize our setup a priori, and hence we need to find a way to evaluate it without knowledge of  $\chi$ .

### E. The interaction error

Assume that we have  $n$  impostor sets at hand for viewpoints  $\chi_1, \dots, \chi_n \in \Lambda \subset X$ . As before, we denote the observer's viewport with  $\chi \in \Lambda$ . Since we can choose from several impostor sets, we display that set whose initial viewport  $\chi_k$  satisfies

$$D(\chi, \chi_k) = \min_{1 \leq j \leq n} D(\chi, \chi_j).$$

For  $1 \leq k \leq n$  let  $\Xi_k$  denote that subset of  $\Lambda$ , on which  $D(\chi, \chi_k)$  is the smallest of all domain errors:

$$\Xi_k := \{ \chi \in \Lambda \mid D(\chi, \chi_k) = \min_{1 \leq j \leq n} D(\chi, \chi_j) \}. \quad (1)$$

Next, we define a probability distribution  $P$  with an associated probability density function  $\mu$  on  $\Lambda$ , for instance, a uniform distribution over  $\Lambda$  or a normal distribution around the current viewport  $\chi$ . These distributions represent the probability for the respective viewport to occur, thus modeling the expected observer movement. We can then calculate the expected value of the error by integrating the domain error  $D$  over  $\Lambda$  with respect to the probability distribution  $P$ .

**Definition 5.** Let  $n \geq 1$ . We define the *interaction error*  $I : \Lambda^n \rightarrow \mathbb{R}$ , where

$$\begin{aligned} I(\chi_1, \dots, \chi_n) &:= \int_{\Lambda} \min_{1 \leq j \leq n} D(\chi, \chi_j) dP(\chi) \\ &= \sum_{j=1}^n \int_{\Xi_j} D(\chi, \chi_j) dP(\chi). \end{aligned} \quad (2)$$

The following Lemma shows that the interaction error will decrease as we add more viewpoints.

**Lemma 1.** Let  $\chi_1, \dots, \chi_n \in \Lambda$ . Then

$$I(\chi_1) \geq I(\chi_1, \chi_2) \geq \dots \geq I(\chi_1, \dots, \chi_n).$$

*Proof:* For  $1 \leq k \leq n$ , it is

$$\begin{aligned} I(\chi_1, \dots, \chi_k) &= \int_{\Lambda} \min_{1 \leq j \leq k} D(\chi, \chi_j) dP(\chi) \\ &\leq \int_{\Lambda} \min_{1 \leq j \leq k-1} D(\chi, \chi_j) dP(\chi) \\ &= I(\chi_1, \dots, \chi_{k-1}). \end{aligned}$$

■

## IV. IMPOSTOR PLACEMENT AND ERROR BOUNDS

The efficiency of the proposed method is based on an optimal choice of initial viewports for the impostor sets, as well as an optimized cell partition for each set.

**Theorem 2.** Given renderable objects located in

$$\Omega := \{ (x_1, x_2, x_3, 1)^\top \in \mathbb{P}^3 \mid 0 < a_0 < x_3 < a_{m+1} \leq \infty \},$$

the optimal cell boundaries for viewport translations are given by  $a_i = (1/a_0 - i\delta)^{-1}$ ,  $i = 1, \dots, m$  for a suitable  $\delta(m) > 0$ , and the optimal impostor placement with respect to the error metric is

$$d_i = \frac{2a_i a_{i+1}}{a_i + a_{i+1}}.$$

Note that  $m$  is finite even for domains with infinite depth, that is, when  $a_{m+1} = \infty$  for which  $d_m = 2a_m$ .

*Proof:* For viewport translations the minimum of the domain error  $D$  with respect to the projection plane distance  $d \in [a, b]$  can be found analytically. For details see [16, Theorem 3.2]. ■

With this impostor placement, we have the following asymptotic behaviour of the error with respect to viewport translations:

**Theorem 3.** For a fixed maximal screen space error  $\varepsilon > 0$ , the radius  $r$  of maximal permissible viewport change is proportional to the number of impostors per set  $m$ .

*Proof:* This property emerges during the proof of Theorem 2. For details see [16, Remark 3.5]. ■

This Theorem shows that increasing the number of impostors per set will strongly decrease the interaction error, but the number of displayable impostors is bounded by the graphical capabilities of mobile devices. Due to such limitations, several impostors sets have to be transmitted.

Denote the number of impostor sets with  $n$ . Under certain assumptions we can show that the inspection error can be bounded by

$$C_1 n^{-1/5} \leq I(\chi_1, \dots, \chi_n) \leq C_2 n^{-1/5},$$

for constants  $C_{1/2} = C_{1/2}(\Lambda, m)$ . Proving these bounds will be the endeavor of the next section.

## V. MODEL EVALUATION

**Proposition 1.** Using the  $\mathbb{R}^5$ -parametrization of the viewport space, we can regard the domain error  $D(\chi, \chi_k)$  as a continuous function  $f : \mathbb{R}^5 \times \mathbb{R}^5 \rightarrow \mathbb{R}$  which, for moderate viewport changes, behaves almost linear.

More precisely, we can find positive constants  $a_1, \dots, a_5$  and  $\bar{a}_1, \dots, \bar{a}_5$  such that

$$\|A_1(x - y)\| \leq f(x, y) \leq \|A_2(x - y)\| \quad (3)$$

where  $A_1 := \text{diag}(a_1, \dots, a_5)$  and  $A_2 := \text{diag}(\bar{a}_1, \dots, \bar{a}_5)$ .

**Proposition 2.** The matrices  $A_1$  and  $A_2$  depend on the number of cells  $m$ . For viewport translations they are proportional to  $m^{-1}$  as a direct consequence of Theorem 3.

Before proceeding, we need the following Lemmata.

*Remark 1.* In the following  $A = B + C$  means that the set  $A$  is the direct sum of the sets  $B$  and  $C$ , that is,  $A = B \cup C$  and  $B \cap C = \emptyset$ . In particular,  $\text{vol}(A + B) = \text{vol}(A) + \text{vol}(B)$ .

Similarly,  $A = B - C$  means that  $B = A + C$ , that is,  $C \subset B$  and  $\text{vol}(B - C) = \text{vol}(B) - \text{vol}(C)$ .

**Lemma 4.** Let  $G$  be a bounded, measurable,  $d$ -dimensional subset of  $\mathbb{R}^d$  and let  $B$  be a  $d$ -dimensional ball (with respect to a norm  $\|\cdot\|$ ) of equal volume (cf. Figure 4a). Then

$$\int_G \|x\| dx \geq \int_B \|x\| dx.$$

*Proof:* Denote the radius of  $B$  with  $R$ . Due to  $G = G \cap B + G \setminus B$  and  $B = G \cap B + B \setminus G$ , we can express  $G$  as  $G = (B - B \setminus G) + G \setminus B$ . As the volumes of  $G$  and  $B$  are equal, this also implies  $\text{vol}(G \setminus B) = \text{vol}(B \setminus G)$ .

Moreover, the distance from the origin to all points in  $G \setminus B$  is larger than  $R$  while for all points in  $B \setminus G$  it is smaller. Hence,

$$\int_{G \setminus B} \|x\| dx \geq \int_{G \setminus B} R dx = R \text{vol}(G \setminus B)$$

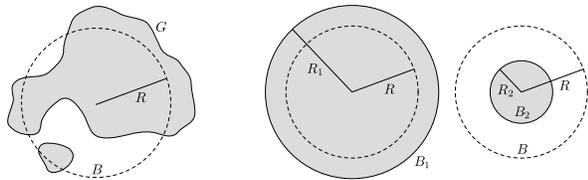
and, conversely,

$$\int_{B \setminus G} \|x\| dx \leq \int_{B \setminus G} R dx = R \text{vol}(B \setminus G).$$

This implies

$$\begin{aligned} \int_G \|x\| dx &= \int_B \|x\| dx - \int_{B \setminus G} \|x\| dx + \int_{G \setminus B} \|x\| dx \\ &\geq \int_B \|x\| dx - \underbrace{R(\text{vol}(B \setminus G) - \text{vol}(G \setminus B))}_{=0}. \end{aligned}$$

■



(a) Lemma 4.

(b) Lemma 5.

Figure 4. Accompanying illustrations for the lemmata.

**Lemma 5.** Let  $B$  and  $B_1, \dots, B_n$  be  $d$ -dimensional balls (with respect to a norm  $\|\cdot\|$ ), such that the volume of  $B$  is the arithmetic mean of the volumes of  $B_1, \dots, B_n$ . Then

$$\sum_{k=1}^n \int_{B_k} \|x\| dx \geq n \int_B \|x\| dx.$$

*Proof:* We first regard the case  $n = 2$ . Without loss of generality, let  $R_1 \geq R \geq R_2$ .

We define  $G := (B_1 - B) + B_2$ . Then,  $\text{vol}(G) = \text{vol}(B_1) - \text{vol}(B) + \text{vol}(B_2) = \text{vol}(B)$  and Lemma 4 yields

$$\begin{aligned} \int_B \|x\| dx &\leq \int_G \|x\| dx \\ &= \int_{B_1} \|x\| dx - \int_B \|x\| dx + \int_{B_2} \|x\| dx. \end{aligned}$$

From this, the general case follows by induction. ■

**Lemma 6.** Let  $B$  be a 5-dimensional ball with radius  $R$ . Then

$$\int_B \|x\|_2 dx = \frac{4}{9} \pi^2 R^6.$$

*Proof:* Straightforward calculation using 5-dimensional polar coordinates. ■

With these Lemmata, we can prove the following estimation of the inspection error:

**Theorem 7.** Let  $\Lambda$  be bounded and assume a uniform distribution of observer viewports. Then, the interaction error can be bounded from below by

$$I(\chi_1, \dots, \chi_n) \geq C_1 n^{-1/5},$$

with the constant

$$C_1 := \frac{5}{6} \left( \frac{15}{8\pi^2} \det(A_1) \text{vol}(\Lambda) \right)^{1/5},$$

where  $A_1 := \text{diag}(a_1, \dots, a_5)$  with constants  $a_i > 0$  as in Proposition 1.

*Proof:* Let us first recall (1) and (2). Assuming a uniform distribution  $\mu(\chi) = \text{vol}(\Lambda)^{-1}$  we can rewrite (2) as

$$I(\chi_1, \dots, \chi_n) = \text{vol}(\Lambda)^{-1} \sum_{k=1}^n \int_{\Xi_k} D(\chi, \chi_k) d\chi. \quad (4)$$

On the right-hand side, we have to evaluate  $n$  integrals of the form  $\int_G f(x, y) dx$ . Using (3) we define a transformation of coordinates  $\Phi(x) := A_1(x - y)$  (which is the same for all  $n$  integrals) and obtain

$$\int_G f(x, y) dx \geq \int_G \|\Phi(x)\| dx = \frac{1}{\det(A_1)} \int_{\Phi(G)} \|x\| dx.$$

Applying this to (4) yields

$$I(\chi_1, \dots, \chi_n) \geq (\det(A_1) \text{vol}(\Lambda))^{-1} \sum_{k=1}^n \int_{\Phi_k(\Xi_k)} \|x\| dx. \quad (5)$$

Using Lemmata 4 and 5 (with  $d = 5$ ), we obtain

$$\sum_{k=1}^n \int_{\Phi_k(\Xi_k)} \|x\| dx \geq \sum_{k=1}^n \int_{B_k} \|x\| dx \geq n \int_B \|x\| dx,$$

where

$$\begin{aligned} \text{vol}(B) &= \frac{1}{n} \sum_{k=1}^n \text{vol}(B_k) = \frac{1}{n} \sum_{k=1}^n \text{vol}(\Phi_k(\Xi_k)) \\ &= \frac{1}{n} \det(A_1) \text{vol}(\Lambda). \end{aligned} \quad (6)$$

With this, the estimation (5) yields

$$I(\chi_1, \dots, \chi_n) \geq (\det(A_1) \text{vol}(\Lambda))^{-1} n \int_B \|x\| dx \quad (7)$$

Now, we choose to use the Euclidean norm  $\|\cdot\| = \|\cdot\|_2$  for which a 5-dimensional ball with radius  $R$  has the volume  $\text{vol}(B) = \frac{8}{15} \pi^2 R^5$ . Then, (6) implies

$$R = \left( \frac{15}{8n\pi^2} \det(A_1) \text{vol}(\Lambda) \right)^{1/5}.$$

Hence, using Lemma 6,

$$\int_B \|x\| dx = \frac{5}{6n} \det(A_1) \text{vol}(\Lambda) \left( \frac{15}{8n\pi^2} \det(A_1) \text{vol}(\Lambda) \right)^{1/5}.$$

Inserting this into (7) we finally obtain

$$I(\chi_1, \dots, \chi_n) \geq \frac{5}{6} \left( \frac{15}{8n\pi^2} \det(A_1) \text{vol}(\Lambda) \right)^{1/5}.$$

This theorem shows, that the efficiency of any choice of impostor sets cannot be better than the given estimate. The following theorem constructively proves, that a choice of impostor sets with the desired asymptotic dependence exists, that is, that this estimate is actually achievable.

**Theorem 8.** Let  $\Lambda$  be bounded with a uniform distribution and let  $\tilde{\Lambda} \supset \Lambda$  be an enclosing cuboid. Then, there is a set of viewpoints  $\chi_1, \dots, \chi_n$  for which the interaction error satisfies

$$I(\chi_1, \dots, \chi_n) \leq C_2 n^{-1/5},$$

with the constant

$$C_2 := \frac{\pi^2 (\max\{\bar{a}_1, \dots, \bar{a}_5\} \text{diam}(\tilde{\Lambda}))^6}{36 \det(A_2) \text{vol}(\Lambda)},$$

where  $A_2 := \text{diag}(\bar{a}_1, \dots, \bar{a}_5)$  with constants  $\bar{a}_i > 0$  as in Proposition 1.

*Proof:* To begin with, we will proof the assertion for those  $n$  which are the fifth power of a whole number, that is, for  $n^{1/5} \in \mathbb{N}$ . The general case will be derived from this case later.

First, a bounded set  $\Lambda$  can be embedded into a cuboid  $\tilde{\Lambda}$ . For an  $n$  chosen as above, there is a regular decomposition of  $\tilde{\Lambda}$  into five-dimensional cuboids  $\Xi_k$  with initial viewpoints  $\chi_k$  at their respective centers.

Using the estimation  $f(x, y) \leq \|A_2(x - y)\| = \|\Psi(x)\|$  with the same arguments as in the proof of Theorem 7, we obtain

$$\begin{aligned} I(\chi_1, \dots, \chi_n) &\leq \text{vol}(\Lambda)^{-1} \sum_{k=1}^n \int_{\Xi_k} D(\chi, \chi_k) d\chi \\ &\leq (\det(A_2) \text{vol}(\Lambda))^{-1} \sum_{k=1}^n \int_{\Psi_k(\Xi_k)} \|x\| dx \\ &\leq (\det(A_2) \text{vol}(\Lambda))^{-1} n \int_B \|x\| dx, \end{aligned} \quad (8)$$

where we used that all cuboids  $\Psi_k(\Xi_k)$  are identical and can be embedded into a ball  $B$  in the last step. For this the radius needs to be at least

$$R = \frac{1}{2} \text{diam}(\Psi_k(\Xi_k)) \geq \max\{\bar{a}_1, \dots, \bar{a}_5\} \frac{\text{diam}(\tilde{\Lambda})}{2n^{1/5}}.$$

With this and Lemma 6 we finally obtain from (8)

$$I(\chi_1, \dots, \chi_n) \leq \frac{\pi^2 (\max\{\bar{a}_1, \dots, \bar{a}_5\} \text{diam}(\tilde{\Lambda}))^6}{72 \det(A_2) \text{vol}(\Lambda)} n^{-1/5}.$$

Now, for the general case, we divide  $\tilde{\Lambda}$  into  $\tilde{n} := \lfloor n^{1/5} \rfloor^5 \leq n$  cubes. This is possible because  $\tilde{n}$  is the fifth power of a whole number ( $\tilde{n}^{1/5} \in \mathbb{N}$ ). Moreover,

$$\frac{\tilde{n}^{-1/5}}{n^{-1/5}} = \frac{n^{1/5}}{\lfloor n^{1/5} \rfloor} \leq \frac{\lfloor n^{1/5} \rfloor + 1}{\lfloor n^{1/5} \rfloor} = 1 + \frac{1}{\lfloor n^{1/5} \rfloor} \leq 2,$$

that is,  $\tilde{n}^{-1/5} \leq 2n^{-1/5}$ . Hence, by this and Lemma (1)

$$\begin{aligned} I(\chi_1, \dots, \chi_n) &\leq I(\chi_1, \dots, \chi_{\tilde{n}}) \\ &\leq \frac{\pi^2 (\max\{\bar{a}_1, \dots, \bar{a}_5\} \text{diam}(\tilde{\Lambda}))^6}{72 \det(A_2) \text{vol}(\Lambda)} \tilde{n}^{-1/5} \\ &\leq \frac{\pi^2 (\max\{\bar{a}_1, \dots, \bar{a}_5\} \text{diam}(\tilde{\Lambda}))^6}{36 \det(A_2) \text{vol}(\Lambda)} n^{-1/5}. \end{aligned}$$

*Remark 2.* As stated earlier, the matrices  $A_1, A_2$  depend on the number of cells  $m$ . With the assumptions in Proposition 2, it follows that  $I = \mathcal{O}(m^{-1} n^{-1/5})$ .

## VI. CONCLUSION

In this paper, we developed a mathematical model which allows to measure, analyze and optimize the display error of image-based approximation techniques. The error asymptotics derived for our method based on parallelized rendering shows a clear advantage over traditional remote visualization concepts like Virtual Network Computing (VNC) which, under ideal conditions, represent the scene by one image  $m = 1$ . In contrast to this,  $m = 10$  impostors with  $n = 1$  viewport cover the same volume of permissible viewports as  $m = 1$  impostors for  $n = 10000$  optimally chosen viewport sets. Considering the bandwidth  $\mathcal{O}(mn)$  needed for

transmission of impostors compared with the error contribution  $\mathcal{O}(m^{-1}n^{-1/5})$ , the method offers significant decrease of bandwidth consumption, and low latency rendering for the user.

The proposed method strongly benefits from graphical capabilities of clients, such as mobile devices, and will increase its efficiency for each new generation providing increased graphical performance. Due to the parallelization of server-sided image generation, and the proven efficiency thereof, the method is applicable to large and distributed data sets for visualization on mobile devices and thin clients, also including augmented reality applications [17].

#### ACKNOWLEDGMENT

The authors appreciate the support of the 'Federal Ministry of Education and Research' and 'Eurostars' within the Project E! 5643 MobileViz. The Eurostars Project is funded by the European Union.

#### REFERENCES

- [1] F. Lamberti and A. Sanna, "A streaming-based solution for remote visualization of 3d graphics on mobile devices," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 2, pp. 247–260, march-april 2007.
- [2] H.-Y. Shum and S. B. Kang, "A review of image-based rendering techniques," in *IEEE/SPIE Visual Communications and Image Processing*, 2000.
- [3] S. Jeschke and M. Wimmer, "An error metric for layered environment map impostors," Tech. Rep. TR-186-2-02-04, 2002.
- [4] S. Jeschke, M. Wimmer, and W. Purgathofer, "Star: Image-based representations for accelerated rendering of complex scenes," *Proc. of Eurographics*, 2005.
- [5] S. Jeschke, M. Wimmer, and H. Schuman, "Layered environment-map impostors for arbitrary scenes," *Graphics Interface*, May 2002.
- [6] W.-C. Wang, K.-Y. Li, X. Zheng, and E.-H. Wu, "Layered textures for image-based rendering," *Journal of Computer Science and Technology*, vol. 19, no. 5, pp. 633–642, September 2004.
- [7] M. Moser and D. Weiskopf, "Interactive volume rendering on mobile devices," in *13th Fall Workshop: Vision, modeling, and visualization 2008*, O. Deussen, D. Keim, and D. Saupe, Eds. Akademische Verlagsgesellschaft AKA GmbH, 2008, p. 217.
- [8] J. Cohen, D. Manocha, and M. Olano, "Simplifying polygonal models using successive mappings," in *VIS '97: Proceedings of the 8th conference on Visualization '97*. Los Alamitos, CA, USA: IEEE Computer Society Press, 1997, p. 395ff.
- [9] P. Debevec, Y. Yu, and G. Boshokov, "Efficient view-dependent image-based rendering with projective texture-mapping," Berkeley, CA, USA, Tech. Rep., 1998.
- [10] J. Kopf, B. Chen, R. Szeliski, and M. Cohen, "Street slide: browsing street level imagery," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–8, 2010.
- [11] J. Shade, D. Lischinski, D. H. Salesin, T. DeRose, and J. Snyder, "Hierarchical image caching for accelerated walk-throughs of complex environments," in *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1996, pp. 75–82.
- [12] T. A. Funkhouser, "Database management for interactive display of large architectural models," in *GI '96: Proceedings of the conference on Graphics interface '96*. Toronto, Ont., Canada, Canada: Canadian Information Processing Society, 1996, pp. 1–8.
- [13] M. Hoffmann and J. Kohlhammer, "A generic framework for using interactive visualization on mobile devices," *Communications in Computer and Information Science*, vol. 53, no. 4, pp. 131–142, 2009.
- [14] F. Lamberti, C. Zunino, A. Sanna, A. Fiume, and M. Maniezzo, "An accelerated remote graphics architecture for PDAs," in *Web3D '03: Proceedings of the eighth international conference on 3D Web technology*. New York, NY, USA: ACM, 2003, p. 55ff.
- [15] A. Beutelspacher and U. Rosenbaum, *Projective Geometry: From Foundations to Applications*. Cambridge University Press, February 1998.
- [16] R. Reiner, "Numerical methods for optimal impostor pre-fetching in scientific visualization," Diploma Thesis, Karlsruhe Institute of Technology, 2011.
- [17] V. Heuveline, S. Ritterbusch, and S. Ronnas, "Augmented reality for urban simulation visualization," in *INFOCOMP 2011, The First International Conference on Advanced Communications and Computation*. IARIA, 2011, pp. 115–119.