Distortion-sensitive synthesis of texture and geometry in interactive 3D visualization

Nicola Brusco[1,2], Pietro Zanuttigh[1,2], David Taubman[1], Guido Maria Cortelazzo[2]
[1]*University of New South Wales, Sydney, Australia*
[2]*Università di Padova, Padova, Italy*

Abstract

This paper addresses the problem of remote browsing of 3D scenes. Texture and geometry information are both available at server side in the form of scalably compressed images and depth maps. We propose a framework for the dynamic allocation of the available transmission resources between geometry and texture. Both the transmission of new images and the improvement of the already transmitted ones are taken into account. We also introduce a novel strategy for distortion-sensitive synthesis of both geometry and rendered imagery at client side.

1. Introduction

Three dimensional scenes are usually associated to a huge amount of data and their remote visualization is a new problem posing many challenging conceptual and practical issues. An efficient interactive system requires a scalable compression and transmission of the data and many approaches, based both on 3D representation and Image-Based Rendering techniques, have been developed [1],[2],[3]. Another important conceptual issue, studied also in [4], is how to allocate the available transmission resources between texture and geometry information.

We envisage a server and a client, connected via a bandlimited channel. At the client side, a user interactively determines the particular view of interest. In the proposed system the server delivers incremental contributions from two types of pre-existing data: 1) scalably compressed images of the scene from a collection of pre-defined views; and 2) a scalably compressed representation of the scene surface geometry. In [5] we developed a distortion-driven framework to decide how to combine the information from the available images into the view of interest exploiting the available geometry information. In this work we extend this approach to synthesize local surface geometry from a variety of view-dependent depth maps, thus exploiting scalable image compression and distribution techniques for both texture and geometry components, treating depth maps as images. JPIP interactive communication standard [6] may readily be exploited to implement our scene browsing paradigm. It is worth pointing out that this system can be exploited for the browsing of a 3D model even in a single computer, when the model is too large to fit in memory. Section 2 describes the synthesis of geometric information from available depth maps. Section 3 develops the synthesis of texture information for the final rendering. In Section 4 minimum distortion criteria used for geometry and image synthesis are described in details. Section 5 provides some experimental results; Section 6 draws the conclusions and outlines future research directions.

2. Synthesis of geometric information

When 3D information is acquired from the real world, usually it is not stored as a full geometry model G. Acquisitions with passive or active devices lead to a collection of depth maps Z^i, which are subsequently aligned in order to form the complete world description. Furthermore, each depth map Z^i is often associated to a corresponding image V^i. This is the case of stereo-systems or optical active scanners. Of course, the complete geometry can always be obtained by integration of available depth maps. However, shifting our focus to depth maps, rather than on the complete geometry, narrows our attention to only that part of G which is relevant to the reconstruction at hand. This is particularly important to the client-server problem, since we cannot afford to transmit a complete geometry in the case of large, complex scenes.

From the perspective of the rendering, in order to synthesize view V^*, all that is required is a depth map $Z^*[\mathbf{n}]$, identifying the depth of each location $\mathbf{n}$ in V^*.

Of course Z^* could be compressed at the server as an image and then transmitted progressively to the client; the server would have to compress a distinct depth map Z^* for each view the client may wish to render, thus heavily affecting the size of transmitted data.

Instead, in our approach the client synthesizes the depth map Z^* from a collection of available depth maps, Z^{i_1}, Z^{i_2}, ..., which the server has already delivered, with varying levels of fidelity and relevance. This scenario is depicted in Figure 1. The server and the client work with a collection of source images V^i and a collection of depth maps Z^i,

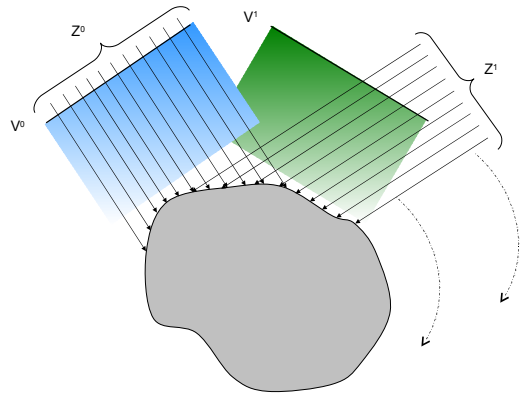

Figure 1: *Browsing environment with a fixed set of view images, V^i, and depth maps Z^i. In this example, V^0 and Z^0 share the same view point, whereas V^1 and Z^1 do not.*

compressed as images and transmitted incrementally. The server must decide how to distribute its available bandwidth between the enhancement of views V^{i_k} which are already partially available at the client, the enhancement of existing depth maps Z^{i_j} which are already partially available at the client, the delivery of a new view, more closely aligned with V^* or the delivery of a new depth map, more closely aligned with Z^*.

Each candidate depth map Z^i is transformed into a corresponding estimate $Z^{i\to *}$ for Z^*. We write this mapping as $Z^{i\to *} = \mathcal{W}(Z^i)$. Our current practical implementation involves building a local triangular mesh for the surface described by Z^i and then warping this mesh into view V^* – a task which may be accomplished using the hardware acceleration features of many popular graphics cards. A depth map can always be mapped to a triangular mesh dividing each sample $Z^*[\mathbf{n}]$ into a pair of triangles. For practical reasons, a decimation of triangles is performed. Thus, $Z^*[\mathbf{n}]$ is the depth of the closest 3D point of the mesh obtained by Z^i, rendered on the sample $\mathbf{n}$. Expansion or contraction in the local affine warping operators associated with this procedure affect the way in which distortion is mapped from subbands in the compressed Z^i images into the synthesized depth candidate, $Z^{i\to *}$.

It must be noticed that each Z^i provides only a part (i.e., what is visible from the point of view of Z^i) of $Z^{i\to *}$. We denote with $\mathcal{O}^i$ the set of indexes k such that $\mathbf{n}_k^i$ is observable (i.e. has a valid depth value) in Z^i :

$$\mathcal{O}^i = \left\{k \mid \mathbf{n}_k^i \text{ is observable in } Z^i\right\}$$

If we denote by $\mathcal{R}^* = \bigcup_{k\in\mathcal{O}^*} \mathbf{n}_k^*$ the silhouette of the object and by $\mathcal{H}^{i\to *} = \bigcup_{n\in\mathcal{O}^*\setminus\mathcal{O}^i} \mathbf{n}_k^*$ the part of $\mathcal{R}^*$ not visible in Z^i, the holes in the output depth map can be divided into two main categories: the pixels outside $\mathcal{R}^*$ (*exterior holes*) and the pixels visible in Z^* but not in Z^i (*interior holes*).

For each sample $\mathbf{n}$ in Z^* we select a single "most appropriate" original depth map from which to select the depth value. We refer to this as "stitching" writing i_k^* for the "best stitching source" for the k^{th} sample, and constructing the synthesized view as a patchwork of these best stitching sources, according to

$$Z^* = \sum_{k\in(\cup_i\mathcal{O}^i)\cap\mathcal{O}^*} \left(Z^{i_k^*\to *}[\mathbf{n}_k]\right) = \sum_{k\in\mathcal{O}^*} \left(Z^{i_k^*\to *}[\mathbf{n}_k]\right) \tag{1}$$

Local distortion estimates in each $Z^{i\to *}$ are used to guide the stitching procedure for Z^*. Distortion estimates and selection of the stitching source are exactly the same as the procedures described in Section 3; the only difference is that the geometry stitching is done in the full image resolution domain, while for image synthesis stitching is performed in the DWT domain. The reason for this is that we need to preserve discontinuities in Z^*, whereas these tend to be destroyed by multi-resolution stitching.

It is worth considering explicitly how "holes" manifest themselves in the depth synthesis problem. Each individual depth map Z^i can generally be expected to exhibit strong discontinuities in regions of object occlusion and indeed these discontinuities do correspond to internal holes in the inferred depth map $Z^{i\to *}$. Compression using waveform coders such as JPEG2000 and JPEG, however, tends to produce ringing and considerable error in the vicinity of such discontinuities. This phenomenon is illustrated schematically in Figure 2. In the figure, compressed depth maps Z^1 and Z^2 are each used to construct candidates $Z^{1\to *}$ and $Z^{2\to *}$ for Z^*, where Z^* corresponds to the vertical elevation of the scene surface in this example. Evidently, Z^1 should have a discontinuity at the point where the scene surface folds back upon itself, but this discontinuity is corrupted by ringing and loss of high frequency details due to compression. As a result, it is not possible to detect the hole $\mathcal{H}^{1\to *}$ which should appear in $Z^{1\to *}$. In fact, holes in $Z^{i\to *}$ are difficult if not impossible to detect, based solely upon the information in Z^i. The missing information is completed by a second map Z^2, for which the inferred depth $Z^{2\to *}$ is shown as a dotted line in the figure.

Fortunately, our distortion-based stitching procedure recognizes that $Z^{1\to *}$ is subject to a great deal of distortion due to the stretching which Z^1 undergoes in the vicinity of surface folding. To encourage this, we ensure that the distortion estimates used for each source depth map Z^i have the property that distortion is judged to be at least as large as the local variance of that depth map, in addition to any estimates of the underlying compression noise. The rule serves to ensure that regions in which depth map Z^1 was

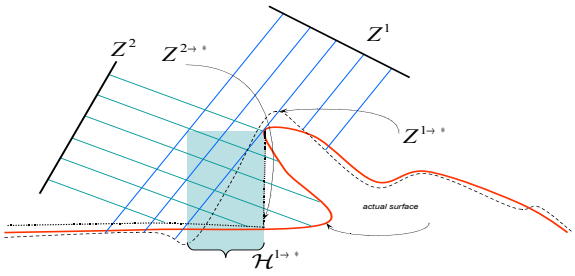

Figure 2: *Illustration of the effect of discontinuities in individual depth maps Z^i on the synthesized depth candidates $Z^{i\to *}$. In the figure, the depth of interest Z^* which we wish to synthesize is the vertical height of the surface.*

originally discontinuous will be treated with distrust, which is then further amplified by the stretching as Z^1 is mapped to $Z^{1\to *}$, encouraging the selection of the much more reliable candidate $Z^{2\to *}$ in such regions. The final stitched depth map can have much less uncertainty than either of the candidates $Z^{i\to *}$, taken in isolation, allowing for high quality view synthesis via the methods of next Section.

3. Synthesis of texture information

In this section we discuss how a set of available views of the 3D scene can be combined together to obtain the rendering of an unknown view V^* using the geometry information given by Z^*. For simplicity, we assume that Z^* is represented by a triangular mesh.

We start from the process of rendering V^* from a single original view image, V^i. By projecting the three dimensional points on the image planes of V^* and V^i by isometric or perspective projection we obtain two sets of corresponding triangles ($\{\Delta_n^*\}$ and $\{\Delta_n^i\}$). Of course not all the triangles of the required view V^* are visible in V^i. If a triangle of V^* is hidden[1] in V^i it represent a hole in the warped image. We denote with $\mathcal{O}^i$ the set of indexes n such that Δ_n^i is observable in V^i :

$$\mathcal{O}^i = \left\{ n \mid \Delta_n^i \text{ is observable in } V^i \right\}$$

Each visible triangle is then rendered by affine warping[2] of Δ_n^i. We write the overall piecewise affine mapping as $V^* = \mathcal{W}^i\left(V^i\right)$. It is important to note that $\mathcal{W}_n^i$ is a linear operator that defines the pixels of Δ_n^i but can depend on the entire domain of V^i due to the effects of interpolation kernels. It is also important to note that the warping procedure will not fill all the pixels of the destination image. If we denote by $\mathcal{R}^* = \bigcup_{n\in\mathcal{O}^*} \Delta_n^*$, the silhouette of the object and by $\mathcal{H}^{i\to *} = \bigcup_{n\in\mathcal{O}^*\setminus\mathcal{O}^i} \Delta_n^*$ the part of $\mathcal{R}^*$ not visible in V^i, the holes in the output image can be divided into two main categories: the pixels outside $\mathcal{R}^*$ (*exterior holes*) and the pixels visible in V^* but not in V^i (*interior holes*).

In the proposed scheme there are multiple views to be warped and it is necessary to combine the different renderings together. The simplest way to combine the information from multiple original view images consists in averaging the renderings $\mathcal{W}^i\left(V^i\right)$. Unfortunately the imperfections in the geometry description produce misalignment amongst the separate renderings and averaging tends to blur high frequency spatial features.

An alternative solution is to render each triangle from a single original view image V^i ("stitching"). Stitching solves the blurring problem but produces visible discontinuities at the boundaries between adjacent triangles taken from different images due to misalignments and lighting issues.

A possible solution to these problems consists in performing stitching within a multiresolution framework, such as the discrete wavelet transform (DWT). Such an approach performs much more smoothing in the lower frequencies than in the higher ones. All the images used in the proposed system are also compressed in JPEG2000, which is a based on the DWT decomposition. The procedure starts by separately warping each source image. Each of the warped image $V^{i\to *} \triangleq \mathcal{W}^i\left(V^i\right)$ is then decomposed by a D level DWT into a low resolution base image LL_D and a collection of high-pass subbands $\mathrm{HL}_d^{i\to *}$, $\mathrm{LH}_d^{i\to *}$ and $\mathrm{HH}_d^{i\to *}$, as shown in Figure 3.

Noting that the synthesis operator $\mathcal{S}$ is linear, the recursive synthesis procedure can be written as

$$\begin{aligned}\mathrm{LL}_d &= \mathcal{S}\left(\mathrm{LL}_{d+1}, \mathbf{0}, \mathbf{0}, \mathbf{0}\right) + \mathcal{S}\left(\mathbf{0}, \mathrm{HL}_{d+1}, \mathrm{LH}_{d+1}, \mathrm{HH}_{d+1}\right) \\ &= \mathcal{S}_{\mathrm{L}}\left(\mathrm{LL}_{d+1}\right) + \underbrace{\mathcal{S}_{\mathrm{H}}\left(\mathrm{HL}_{d+1}, \mathrm{LH}_{d+1}, \mathrm{HH}_{d+1}\right)}_{R_d}\end{aligned}$$

Here, $\mathcal{S}_{\mathrm{L}}$ and $\mathcal{S}_{\mathrm{H}}$ are the low- and high-pass portions of a single stage of DWT synthesis, and R_d denotes the "detail" image formed from the three high-pass subbands at decomposition level $d+1$.

Our multi-resolution stitching algorithm proceeds by separately stitching each resolution component to form

$$R_d^* = \sum_{n\in\mathcal{O}^*} R_d^{i_{n,d}^*\to *}\Big|_{\Delta_{n,d}^*}, \quad d = 0, 1, 2, \ldots, D \qquad (2)$$

Here $\Delta_{n,d}^*$ represents the support of triangle Δ_n^*, as it appears in resolution component R_d – i.e., reduced in size by

[1] Partially hidden triangles can be avoided by choosing a suitably fine mesh

[2] Affine warping does not exactly reproduce the behavior of a perspective projection inside the projected triangles but this error can be rendered arbitrarily small by reducing the size of the triangles.

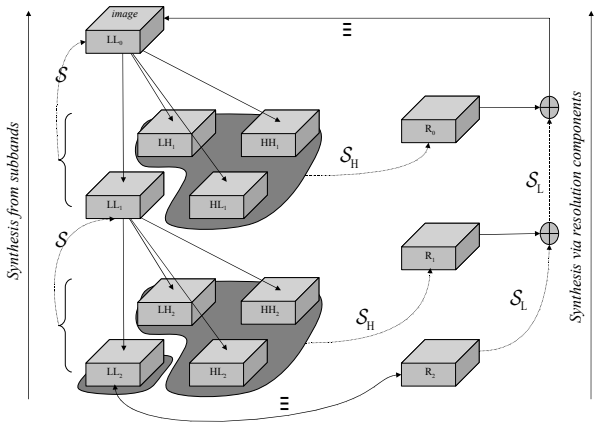

Figure 3: *Relationship between DWT subbands, resolution components and the image which they represent, for the case of a $D = 2$ level decomposition.*

the factor 2^d. We also note that the "best stitching source," $i^*_{n,d}$, need not necessarily be the same in every resolution component R_d. This is an important property, since one source view might provide high quality details at some resolutions but not at others, depending upon compression noise and the view orientation.

Finally the DWT analysis is performed in order to reconstruct the complete warped rendering.

4. Selection of the stitching sources

The biggest issue in such an approach is how to select from which image every sample in every subband must be taken. Our approach aims to minimize the distortion in the final rendered image. The distortion can be due to many different sources including image compression, geometry uncertainty and lighting issues.

4.1 Image compression

In our images JPEG2000 compression introduces a quantization error in the DWT samples of the various subbands. Then the error in every sample in subband b of the image V^i finds its way into subband b^* of $V^{i\to *}$ through DWT synthesis, warping and further DWT analysis. The quantization error in V^i can be expressed as

$$\delta V^i[\mathbf{n}] = \sum_b \sum_{\mathbf{k}} \delta B^i_b[\mathbf{k}] \cdot S^b_{\mathbf{k}}[\mathbf{n}]$$

Here, B^i_b is subband b from image V^i, $\delta B^i_b[\mathbf{k}]$ is the error in the $\mathbf{k}^{\text{th}}$ sample of this subband and $S^b_{\mathbf{k}}$ is the synthesis basis vector for that sample. After applying the warping operator $\mathcal{W}^i$ and performing DWT analysis, the quantization error in sample $\mathbf{p}$ is:

$$\begin{aligned}\delta R^{i\to *}_d[\mathbf{p}] &= \left\langle \mathcal{W}^i\left(\delta V^i\right), A^d_{\mathbf{p}}\right\rangle \\ &= \sum_b \sum_{\mathbf{k}} \delta B^i_b[\mathbf{k}] \cdot \left\langle \mathcal{W}^i\left(S^b_{\mathbf{k}}\right), A^d_{\mathbf{p}}\right\rangle \quad (3)\end{aligned}$$

where $A^d_{\mathbf{p}}$ is the analysis basis vector associated with that sample, and $\langle\cdot,\cdot\rangle$ represents the inner product between two images. Assuming that the quantization errors in the subbands are uncorrelated, the total error energy in the triangle $\Delta^*_{n,d}$ in resolution component $R^{i\to *}_d$ is then

$$\begin{aligned}D^{i\to *}_{n,d} &= \sum_{\mathbf{p}\in\Delta^*_{n,d}} \left|\delta R^{i\to *}_d[\mathbf{p}]\right|^2 \quad (4)\\ &\approx \sum_b \underbrace{\sum_{\mathbf{p}\in\Delta^*_{n,d}} \sum_{\mathbf{k}} \left|\delta B^i_b[\mathbf{k}]\right|^2 \cdot \left\langle \mathcal{W}^i\left(S^b_{\mathbf{k}}\right), A^d_{\mathbf{p}}\right\rangle^2}_{D^{i\to *}_{n,b\to d}}\end{aligned}$$

The value of $D^{i\to *}_{n,b\to d}$ depends most on the samples inside the projection of Δ^i_n into subband B^i_b. A good approximation of $D^{i\to *}_{n,b\to d}$ can so be obtained by considering a uniform quantization power inside the subband. By calling with $W^n_{b\to d}$ the average of $\sum_{\mathbf{k}} \left\langle \mathcal{W}^i_n\left(S^b_{\mathbf{k}}\right), A^d_{\mathbf{p}}\right\rangle^2$ for the samples in the triangle we obtain[3]:

$$D^{i\to *}_{n,b\to d} \approx D^i_{n,b} \cdot \frac{\left|\Delta^*_{n,d}\right|}{\left|\Delta^i_{n,b}\right|} \cdot W^n_{b\to d} \quad (5)$$

It is important to underline that the affine operator $\mathcal{W}^i_n$ stretches the synthesis basis function by an amount proportional to $|\Delta^*_n| / |\Delta^i_n|$, amplifying its energy by roughly the same amount.

Assuming an orthonormal transform, the weights can be expressed as:

$$\sum_d \sum_{\mathbf{p}} \left\langle \mathcal{W}^i_n\left(S^b_{\mathbf{k}}\right), A^d_{\mathbf{p}}\right\rangle^2 = \left\|\mathcal{W}^i_n\left(S^b_{\mathbf{k}}\right)\right\|^2 = |\Delta^*_n| / |\Delta^i_n|, \ \forall \mathbf{k}$$

and the total distortion in the triangle as:

$$\sum_d D^{i\to *}_{n,d} = \sum_d \sum_b D^i_{n,b} \cdot \frac{\left|\Delta^*_{n,d}\right|}{\left|\Delta^i_{n,b}\right|} \cdot W^n_{b\to d} = \frac{|\Delta^*_n|}{|\Delta^i_n|} \sum_b D^i_{n,b}$$

The total distortion in the warped triangle seems to be roughly independent of the affine operator $\mathcal{W}^i_n$, since the

[3] In our implementation the values of $W^n_{b\to d}$ has been pre-computed and stored in a lookup table

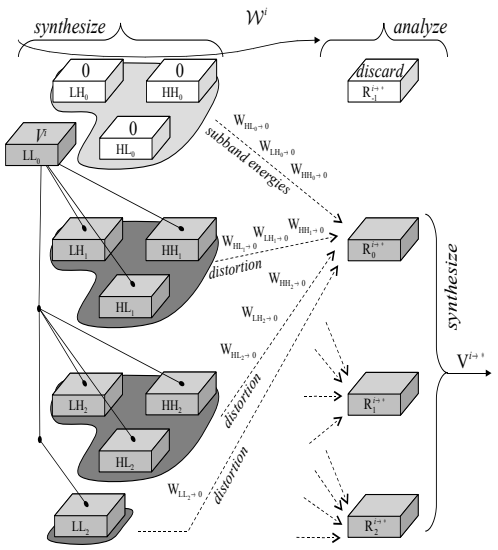

Figure 4: *Procedure for mapping both imagery and distortion information from view V^i to warped view $V^{i\rightarrow *}$. Extra, hypothetical resolutions are shown lightly shaded.*

total distortion in the source triangle is usually proportional to its area. However, two important elements are missing from this picture: the first is that $\mathcal{W}_n^i$ isn't a bandlimited warping operator and when $|\Delta_n^*| / |\Delta_n^i| < 1$ extra super-Nyquist spatial frequencies are suppressed by the discrete warping operator and V^i should yield less distortion power making it more suitable to be selected as best source. The second important oversight arises when $|\Delta_n^*| / |\Delta_n^i| > 1$ (V^i is being expanded). In this case the high frequency components of V^* at the highest resolution cannot be recovered at all. This represent a form of extra distortion that can be modeled by adding subbands from a set of hypothetical resolution above the real ones (see Figure 4). Since there is no information for these subbands, their distortions $D_{n,b}^i$ are considered equal to their energies[4] $E_{n,b}^i$.

The extra contribution introduced by missing subbands grows with $|\Delta_n^*| / |\Delta_n^i|$, thus forcing the choice of images in which the required triangle is bigger.

In a real remote application the client should know the localized quantized distortion in each subband of each source image. If the image are compressed in JPEG2000 it is possible to obtain a reasonable estimate of the distortion (up to 3 db) directly from the received code-blocks [8].

[4]One way to obtain a conservative estimate for these energies is to project each source image onto the other in turn, taking the maximum of the energy produced by such projections.

Up to this point, for simplicity reasons, we represented the geometry G in terms of triangular mesh elements. The 3D mesh can be directly replaced by the depth map Z^* obtained by the available depth maps Z^i, as described in Section 2. This permits to avoid the need for remeshing near boundaries and holes, and allows stitching decisions to follow meaningful scene features rather than artificial mesh structures.

To adapt the distortion formulation to the case of depth maps we can take the limit as triangles Δ_n^* and Δ_n^i becomes arbitrarily small.

The equations (4) and (5) can be modified in order to obtain per-sample distortion estimates:

$$\begin{aligned} D_d^{i\rightarrow *}[\mathbf{p}] &= \sum_b D_{b\rightarrow d}^{i\rightarrow *}[\mathbf{p}] \\ &= \sum_b W_{b\rightarrow d}[\mathbf{p}] \cdot D_b^i \left[\left(\mathcal{W}_{b\rightarrow d}^i\right)^{-1}(\mathbf{p}) \right] \end{aligned} \quad (6)$$

Where $W_{b\rightarrow d[\mathbf{p}]}$ is the warping operator corresponding to the location of P and $\left(\mathcal{W}_{b\rightarrow d}^i\right)^{-1}$ maps locations in the destination image back to corresponding location in the source image (it will generally fall between available distortion samples and some interpolation is required).

It is worth noting that our weighting formulation was developed based upon the assumption that the triangular mesh elements are large compared with the support of $\left\langle \mathcal{W}_n^i\left(S_{\mathbf{k}}^b\right), A_{\mathbf{p}}^d \right\rangle$ – this is certainly not true in the case of depth maps samples. The problem can be faced by applying a low-pass smoothing filter to the distortion estimates.

The warped source views $V^{i\rightarrow *}$ are subject to the appearance of exterior and interior "holes". Exterior holes correspond to pixel outside the silhouette of the object. Since DWT involves overlapping basis functions and the region of support is not limited to the silhouette, it is sufficient to to take samples for the region outside the silhouette from a suitable warped view. Interior holes instead occupy different regions in each of the $V^{i\rightarrow *}$. A hole influences a larger region, in the case of JPEG2000's 9/7 DWT it grows roughly as 2^d. Samples which corresponding DWT basis function overlap the interior holes should be avoided or at least used only if no other view is available for that sample.

4.2 Depth uncertainty

Uncertainty in the surface geometry translates into uncertainty in the locally affine warping operator. This, in turn, represents a translational uncertainty, which has been studied previously in [7]. Figure 5 shows schematically how uncertainty in depth $\delta Z_{\mathbf{n}}^*$ can be converted into a corresponding uncertainty in position $\delta\mathbf{n} = \mathbf{n}' - \mathbf{n}$, within the warped image $V^{i\rightarrow *}$. In the figure, $\mathbf{x}_{\mathbf{n}}$ denotes the 3D point corresponding to location $\mathbf{n}$ in V^*, with depth $Z_{\mathbf{n}}^* = Z^*[\mathbf{n}]$;

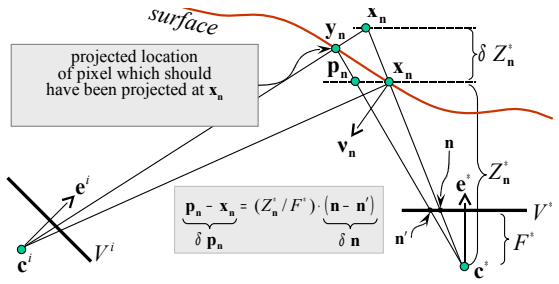

Figure 5: *Relationship between the various coordinates used to map depth uncertainty $\delta Z_{\mathbf{n}}^*$ at location n in V^* into displacement uncertainty δn in $V^{i\rightarrow *}$.*

$\mathbf{x}_{\mathbf{n}}' = \mathbf{x}_{\mathbf{n}} + \delta\mathbf{x}_{\mathbf{n}}$ identifies the true location of the observed point, assuming a depth error of $\delta Z_{\mathbf{n}}^*$. Of course, $\delta\mathbf{x}_{\mathbf{n}}$ and $\delta Z_{\mathbf{n}}^*$ are related through

$$\delta\mathbf{x}_{\mathbf{n}} = \delta Z_{\mathbf{n}}^*/Z_{\mathbf{n}}^* \cdot (\mathbf{x}_{\mathbf{n}} - \mathbf{c}^*)\,,$$

where we are using the notation $\mathbf{c}^*$ and $\mathbf{c}^i$ to denote the focal points associated with views V^* and V^i. The positional error $\delta\mathbf{x}_{\mathbf{n}}$ corresponds to a translational shift in view V^i so that the pixel information which should have been mapped to location $\mathbf{n}$ on V^* was actually mapped to location $\mathbf{n}'$, based upon an assumed scene surface which passes through $\mathbf{x}_{\mathbf{n}}$ with normal $\nu_{\mathbf{n}}$. In the figure, point $\mathbf{y}_{\mathbf{n}}$ denotes the corresponding location on this surface; its projection onto V^* yields the location $\mathbf{n}_{\mathbf{n}}'$.

It can be shown that:

$$|\delta\mathbf{n}|^2 = |\delta Z_{\mathbf{n}}^*|^2 \cdot g_{\mathbf{n}}^{i\rightarrow *}$$

where the factor $g_{\mathbf{n}}^{i\rightarrow *}$ depends upon fixed viewing parameters, together with the surface position $\mathbf{x}_{\mathbf{n}}$ and normal $\nu_{\mathbf{n}}$ seen at location $\mathbf{n}$ in view V^*.

Positional uncertainty may further be converted to amplitude distortion, using the method developed in [7]. Specifically, we obtain an additional contribution to the distortion in warped resolution component $R_d^{i\rightarrow *}$, of the form

$$|\delta\mathbf{n}|^2 \cdot \frac{1}{(2\pi)^2} \int_{\mathcal{B}_d} \Gamma_{V^{i\rightarrow *}}(\omega) \cdot |\omega|^2 \cdot d\omega \qquad (7)$$

where $\mathcal{B}_d \subset [-\pi,\pi]^2$ is the region occupied by resolution component R_d in the discrete space Fourier domain and $\Gamma_V(\omega)$ is the discrete space power density spectrum of image V. Of course, the magnitude of $|\delta\mathbf{n}|^2$ varies locally, and we are also interested in local per-sample distortion contributions $D_d^{i\rightarrow *}[\mathbf{p}]$. We accommodate this by augmenting equation (6) as follows:

$$D_d^{i\rightarrow *}[\mathbf{p}] = \sum_b W_{b\rightarrow d}[\mathbf{p}] \cdot D_b^i\left[\left(\mathcal{W}_{b\rightarrow d}^i\right)^{-1}(\mathbf{p})\right] + \overline{|\delta Z_d^*|^2}[\mathbf{p}] \cdot \overline{g_d^{i\rightarrow *}}[\mathbf{p}] \cdot \overline{|\omega_d|^2} \cdot \overline{E_d^{i\rightarrow *}[\mathbf{p}]} \qquad (8)$$

Here, $\overline{|\delta Z_d^*|^2}[\mathbf{p}]$ is obtained by reducing the resolution of an estimated local distortion field for $Z^*[\mathbf{n}]$ by the factor 2^d and applying a suitable low-pass smoothing filter. Similarly, $\overline{g_d^{i\rightarrow *}}[\mathbf{p}]$ is obtained by reducing the resolution of the $g_{\mathbf{n}}^{i\rightarrow *}$ field by factor 2^d and smoothing the result. All smoothing filters are the same as the one used for $\overline{E_d^{i\rightarrow *}}[\mathbf{p}]$, the local average power in $R_d^{i\rightarrow *}$ at location $\mathbf{p}$. In equation (7), $|\omega|^2$ serves as a frequency-dependent weighting of signal power. In the model of equation (8), this is replaced by an assumed "average" weight $\overline{|\omega_d|^2}$, for the whole of resolution component R_d. In practice, we use a simple mid-band approximation to get

$$\overline{|\omega_d|^2} = \left(\frac{3\pi}{4}2^{-d}\right)^2$$

A variety of other methods to select $\overline{|\omega_d|^2}$ may be found in [7].

Before concluding this sub-section, we note that the true depth map Z^* should be discontinuous at the boundaries of objects which occlude other objects from view V^*. Discontinuous depth cannot be accurately represented using discrete samples $Z^*[\mathbf{n}]$ and this problem only becomes worse when depth information is derived from compressed data, as discussed in Section 2. For this reason, it is reasonable to assign a depth uncertainty power, $|\delta Z_{\mathbf{n}}^*|^2$ which is at least as large as the local variance in $Z_{\mathbf{n}}^*$. In this way, we implicitly distrust the accuracy of our geometric representation in the vicinity of occluding boundaries. All things (such as source view distortion) being equal, this encourages the selection of views with smaller values of $g_{\mathbf{n}}^{i\rightarrow *}$ – these are the views which are closer to V^*.

4.3 Illuminant uncertainty

Since our surface model does not account for the illuminant-dependent effects of shading and reflection, we can expect a second distortion contribution which grows with the deviation between the orientation of views V^* and V^i. Ignoring specularity, we expect this distortion term to be proportional to the signal power, suggesting the following augmented version of equation (8).

$$D_d^{i\rightarrow *}[\mathbf{p}] = \sum_b W_{b\rightarrow d}[\mathbf{p}] \cdot D_b^i\left[\left(\mathcal{W}_{b\rightarrow d}^i\right)^{-1}(\mathbf{p})\right] + \overline{|\delta Z_d^*|^2}[\mathbf{p}] \cdot \overline{g_d^{i\rightarrow *}}[\mathbf{p}] \cdot \overline{|\omega_d|^2} \cdot \overline{E_d^{i\rightarrow *}[\mathbf{p}]} + \gamma \tan\left(\cos^{-1}\left\langle \mathbf{e}^i, \mathbf{e}^*\right\rangle\right) \cdot \overline{E_d^{i\rightarrow *}[\mathbf{p}]} \qquad (9)$$

Here, $\mathbf{e}^i$ and $\mathbf{e}^*$ are the view directions, as shown in Figure 5. In the absence of careful modeling, γ is a heuristically assigned quantity, which determines the value we place on illumination fidelity. Note that both of the distortion contributions from geometry vary with the local power $\overline{E_d^{i\rightarrow *}}$

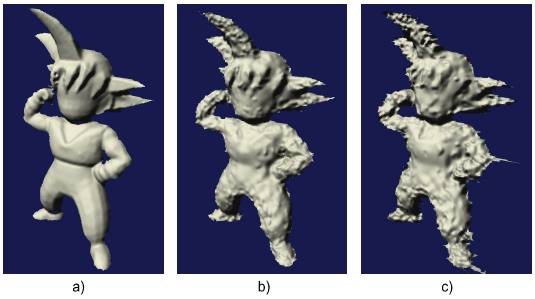

Figure 6: *Synthesized depth map Z^*: a) from 0.4 bpp b) from 0.05 bpp c) from 0.025 bpp depth maps*

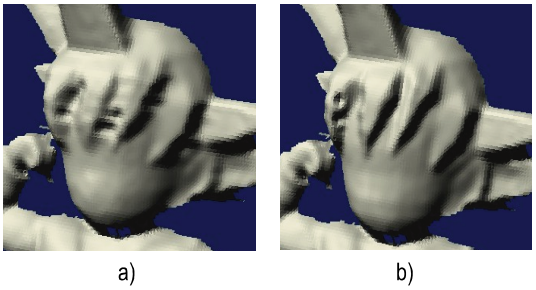

Figure 7: *Goku depth map: a) obtained using the lower depth values; b) obtained using the depth values with lower distortion.*

in the relevant resolution component. The first term varies additionally with depth uncertainty, spatial frequency and properties of the surface normal (through $\overline{g_d^{i\rightarrow *}}$), whereas the second term is affected only by the viewing angles $\mathbf{e}^i$ and $\mathbf{e}^*$.

5. Experimental results

We used both real and synthetic models for our rendering experiments. Real 3D models were obtained with a passive modeling method [9]. Passive modeling procedures lead to geometric error where texture is missing or illumination causes artifacts (such as shadows or specular reflections). That error is taken into account in Equation 8.

We also used synthetic models and we rendered them, storing both the rendering and z-buffer in order to obtain depth maps Z^i associated to images V^i. All $Z^i$and V^i have been stored in a JPEG2000 file.

When rendering takes place, a new depth map Z^* is synthesized from the existing ones Z^i. Quality of available depth maps affects the quality of the reconstructed depth map Z^*, as it is shown in Figure 6 with different bit-rates for the available Z^i. Since Z^* takes contribution from more depth maps, its quality is higher than the quality of Z^i singularly taken.

When our distortion-based stitching procedure generates Z^*, the distortion in every depth map is taken into account. High distortion, which leads to a relevant discrepancy among depth-maps, is due to low quality compression or a large angle between between the required depth map Z^* and the available map Z^i as described in Section 2. In Figure 7a) distortion is not considered, and the depth map $Z^{i\rightarrow *}$ with lowest value (i.e., closest to the viewer) is chosen for each sample. It can be seen that small details between hairs are missing, since depth maps with higher distortion affect the final result. This problem is solved selecting the source Z^i upon the basis of distortion information, as shown in Figure 7b).

Once geometry Z^* is available, rendering of V^* takes

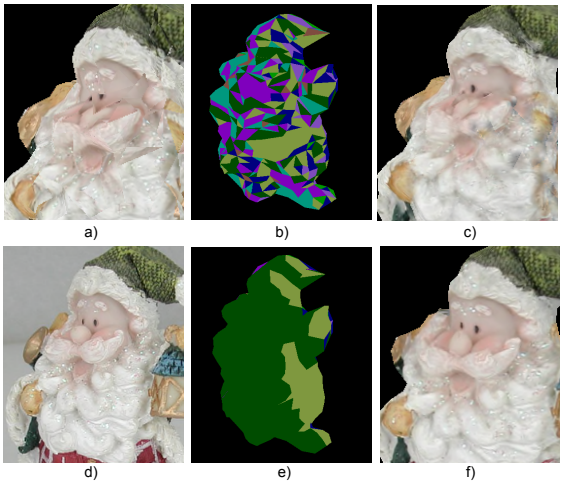

Figure 8: *Synthesized depth map Z^*: a) from 0.4 bpp b) from 0.05 bpp c) from 0.025 bpp depth maps Z^i*

place as described in Section 3. At the client side, geometry of Santa Claus is available with twelve photos V^i at the same bit-rate (0.4 bpp). In Figure 8a) a detail is shown, with the result of stitching in the image domain: discontinuities are visible. When the stitching decision is applied only on the basis of image distortion alone (Equation 6), other artifacts appear: a detail of the rendering V^* is shown in Figure 8c) and the stitching source are shown in Figure 8b) as patches of different colors. Artifacts are mainly due to geometric error, which causes misalignment between different patches: switching among many different sources produces a lot of noise. When the full distortion of Equation 8 is used, many triangles are taken from the most parallel view (which is the most consistent) as shown in Figure 8e). The resulting V^* is much nicer, as shown in Figure 8f), and it is more similar to the original object as it appears in Figure 8d).

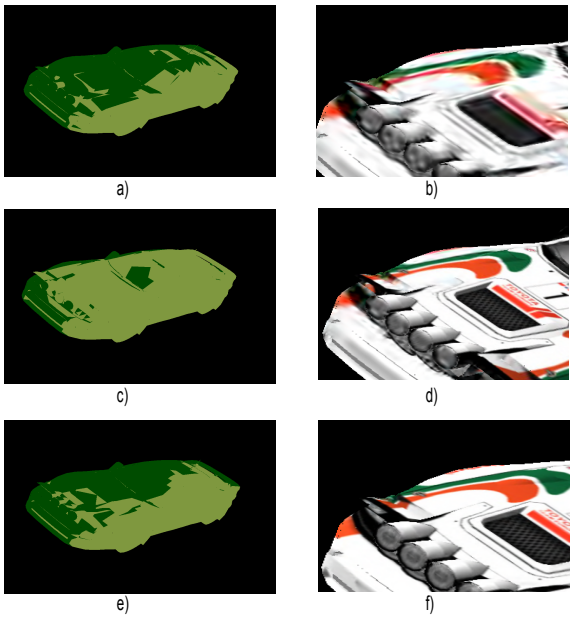

Figure 9: *Synthesized depth map Z^*: a) from 0.4 bpp b) from 0.05 bpp c) from 0.025 bpp depth maps*

If images are sent in a progressive manner, usually they are not available at the client side with the same quality. Images $V^{1'}$ and $V^{2'}$ of the synthetic Car model are sent to the client at low bit-rate (0.025 bpp). Figure 9a) shows the stitching decision and Figure 9b) shows a detail of the final rendered image V^*. Triangles are equipartitioned between $V^{1'}$ and $V^{2'}$ since the images distortions are similar. Then, the server sends image $V^{2"}$ which is taken from the right side of the model at a highest bit rate (0.4 bpp). As it can be seen in 9c), more triangles are taken from $V^{2"}$ since it shows a lowest distortion. The detail of the rendering of Figure 9d) shows a high improving in quality, since the most of the low quality image is discarded. Finally, the server sends image $V^{1"}$ and a new stitching decision, shown in 9e) is computed. The detail of rendering of Figure 9f) shows high quality details all over the surface.

6. Conclusions and future research

In this paper we developed a novel approach to the interactive transmission and rendering of 3D scenes, that lies between standard 3D visualization and Image-Based Rendering techniques. The representation of both texture and geometry information as a set of compressed images permits to fully exploit the scalable compression features of JPEG2000 to realize an interactive visualization system. We also introduced a novel framework for estimating the distortion in the rendered views and experimentally validated a client-side rendering system which aims to minimize this distortion. The proposed approach permits to synthesize both geometry and texture information starting from an arbitrary set of compressed depth maps and images, and also permits to add new views or depth information at any time during the interactive navigation of the scene.

The next step will be the development of a server policy to decide how to distribute the available transmission resources between the various views and geometry information.

References

[1] M. Levoy, "Polygon-assisted JPEG and MPEG compression of synthetic images," in *Proc. SIGGRAPH*, vol. 3, Aug 1995, pp. 21–28.

[2] P. Ramanathan and B. Girod, "Receiver-driven rate-distortion optimized streaming of light fields," in *Proc. IEEE Int. Conf. Image Proc.*, vol. 3, Sep 2005, pp. 2528.

[3] D. Cohen-Or, "Model-based view-extrapolation for interactive VR web systems," in *Proc. Computer Graphics International*, Jun 1997, pp. 104–112.

[4] I. Cheng and A. Basu, "Reliability and judging fatigue reduction in 3D perceptual quality," in *IEEE Int. Symp. on 3DPVT*, Sep 2004.

[5] P. Zanuttigh, N. Brusco, D. Taubman, G.M. Cortelazzo, "Greedy Non-Linear Optimization of The Plenoptic Function For Interactive Transmission Of 3D Scenes" *International Conference of Image Processing, ICIP05, Genova*, 2005.

[6] D. Taubman and R. Prandolini, "Architecture, philosophy and performance of jpip: internet protocol standard for JPEG 2000," *Int. Symp. Visual Comm. and Image Proc.*, vol. 5150, pp. 649–663, July 2003.

[7] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Proc.*, vol. 13, no. 8, pp. 1029–1041, Aug 2004.

[8] D. Taubman, "Localized distortion estimation from already compressed JPEG2000 images" *submitted to Proc. IEEE Int. Conf. Image Proc.*, 2006.

[9] L. Ballan, N. Brusco, G.M. Cortelazzo "3d passive shape recovery from texture and silhouette information" *CVMP05, 2nd European conference on Visual Media Production, London*, 2005.