SPIE Conference Proceedings: Visual Communications and Image Processings (VCIP-2001)

San Jose, CA, pp. 273-271, January 2001

Multi-View Image Coding with Depth Maps and 3-D Geometry for Prediction

Marcus Magnor[a], Peter Eisert[a], Bernd Girod[b]

[a]Telecommunications Laboratory, University of Erlangen-Nuremberg
91058 Erlangen, Germany

[b]Informations Systems Laboratory, Stanford University
Stanford, CA 94305-9510

ABSTRACT

Block-based disparity compensation is an efficient prediction scheme for encoding multi-view image data. Available scene geometry can be used to further enhance prediction accuracy. In this paper, three different strategies are compared that combine prediction based on depth maps and 3-D geometry. Three real-world image sets are used to examine prediction performance for different coding scenarios. Depth maps and geometry models are derived from the calibrated image data. Bit-rate reductions up to 10% are observed by suitably augmenting depth map-based with geometry-based prediction.

Keywords: Disparity Compensation, Geometry-Based Coding, Image-Based Rendering, Multi-View Images

1. INTRODUCTION

In recent years, image-based rendering (IBR) techniques have been developed with the goal to attain natural-appearing rendering results of arbitrary scenes.[1,2] Besides its photo-realistic rendering capabilities, the advantages of IBR over conventional rendering approaches are its great versatility and scene-independent rendering frame-rate. However, IBR methods require a large number of images as input. For high-quality rendering results, hundreds of megabytes of image data need to be acquired, stored, transmitted and processed for a single scene. Compression is therefore an essential aspect of IBR. Fortunately, the images used in IBR represent highly redundant data which depict static scenes from many different view points. A number of different coding techniques for IBR systems have been proposed, ranging from vector quantization,[3] transform[4] and subband coding[5] to more efficient yet also more complex predictive coding schemes.[6–8] Hierarchical disparity-compensated coding is an efficient technique for encoding the entire content of arbitrary multi-view imagery.[9] If, on the other hand, only objects depicted in the images are of interest, available 3-D geometry models allow to achieve even better coding performance. In geometry-based coding,[10] object geometry is used to encode image regions depicting modeled object surfaces.

In this paper, different strategies are investigated to combine depth-map based prediction with prediction aided by 3-D geometry to achieve improved coding performance for multi-view imagery. In Section 2, scene depth estimation and image prediction using depth maps is explained. Geometry-based prediction is described in Section 3. Section 5 explains the hierarchical coder framework that is used to measure the coding results presented in Section 6.

2. IMAGE PREDICTION USING DEPTH MAPS

When comparing two multi-view images, a point on the surface of the depicted object often appears in both images but at different positions: multi-view images differ mainly due to disparity. The amount of disparity depends on the distance between object surface and camera as well as on the distance between both images' recording positions. For

correspondence to: magnor@LNT.de; www.LNT.de/˜magnor

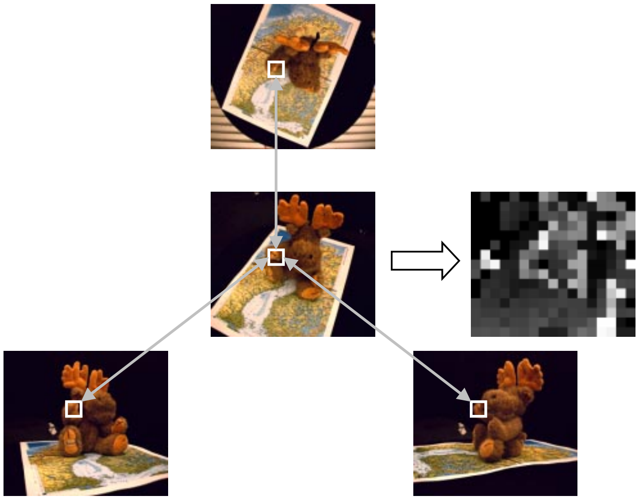


Figure 1. Block-based depth map estimation.

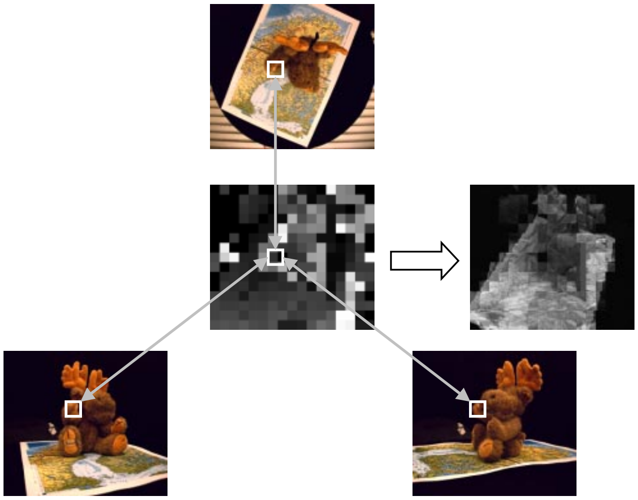


Figure 2. Disparity-compensated image prediction.

two images being recorded from arbitrary positions $\underline{P}_1$, $\underline{P}_2$ and camera orientations $\underline{\underline{R}}_1$, $\underline{\underline{R}}_2$, a pixel at coordinates (s_1, t_1) corresponds to the pixel at coordinates (s_2, t_2) in the other image following the relation

$$\begin{aligned} s_2 &= \frac{N_S}{2}\,(1 - \frac{f\ x_2}{z_2 \Delta s}) \\ t_2 &= \frac{N_T}{2}\,(1 - \frac{f\ y_2}{z_2 \Delta t}) \end{aligned}$$

with

$$\begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} = \underline{\underline{R}}_2 \underline{\underline{R}}_1^{-1} \left[\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} - \underline{P}_1 \right] + \underline{P}_2 \tag{1}$$

and

$$\begin{aligned} x_1 &= -\left(\frac{2\ s_1}{N_S} - 1\right) \frac{z_1}{f}\ \Delta s \\ y_1 &= -\left(\frac{2\ t_1}{N_T} - 1\right) \frac{z_1}{f}\ \Delta t, \end{aligned}$$

where f is the focal length, N_S, N_T denote the respective number of pixels of size Δs, Δt along horizontal and vertical direction, and z_1 is the local scene depth at pixel coordinates (s_1, t_1). While the internal and external camera parameters are known from image calibration, the scene depth is not available a priori, and z_1 must be derived from the image data to be able to perform disparity-compensated image prediction.

2.1. Depth Map Estimation

Exact disparity compensation can be performed only if scene depth is accurately known for each image pixel. Many different, and often computationally demanding, algorithms have been proposed to estimate depth from stereoscopic as well as multi-view images.[11,12] The derivation of correct depth at pixel resolution, however, proves to be an elusive problem. Fortunately, with regard to disparity-compensated image prediction, determining true scene depth is not imperative. Rather, the depth value yielding the best prediction result must be found.

To estimate dense depth maps, the multi-view images are divided into blocks. As depicted in Fig. 1, each image block is compared to the reference images used later for prediction. Different depth values within a predefined search range are considered. The step size is chosen to correspond to one-pixel disparity between target and reference images. The disparity-compensated blocks from all reference images are copied and averaged, and the Sum-Squared-Error (SSE) between the original block and the block's prediction is measured. For each depth value, the pixels within the block are individually warped to the corresponding coordinates in the reference images following (1), taking into account perspective distortion of the rectangular block. The reference images' pixel values are averaged, and the SSE of the predicted block is measured. The value resulting in the smallest SSE-value is selected as the best disparity-compensating depth value for the block.

2.2. Depth Map Coding

A fixed Huffman code table is used to encode the depth-map entries. The block size chosen during depth-map estimation determines the resolution of the maps. While small blocks allow more accurate image predictions, large blocks require a lower depth-map bit-rate. The optimal block size for a given multi-view image set must be determined experimentally.

2.3. Disparity Compensation from Multiple Reference Images

Multi-directional disparity compensation consists of simultaneously copying and averaging pixels from several reference images. Given its depth map, an image can be directly predicted. Fig. 2 illustrates depth map-based disparity compensation using multiple reference images. To predict an image, the depth value for each pixel is extracted from the corresponding depth map, and the disparity-compensated pixel coordinates in each reference image are calculated following (1). All reference-image pixels are copied and averaged, obtaining a dense image prediction.

3. IMAGE PREDICTION USING 3-D GEOMETRY

Available 3-D scene geometry enables accurate disparity compensation as well as the detection of occluded image regions, yielding potentially more accurate prediction results than block-based disparity compensation. However, 3-D object geometry must first be derived from the calibrated multi-view image set.

3.1. Geometry Reconstruction

To robustly reconstruct a 3-D model of the depicted scene, a volumetric reconstruction algorithm is used.[13] In contrast to methods that rely on feature-matching, no point correspondences have to be found. A voxel model is built directly from the calibrated images without image segmentation. The algorithm yields accurate object geometry models from well-calibrated images. In the presence of small calibration inaccuracies, the reconstructed model becomes smaller than the depicted object due to clipping at the object's slightly misaligned silhouette in different images.

The reconstructed solid volumetric model typically consists of more than one million voxels. To efficiently compress object surface geometry, a triangle-mesh description is more desirable. The Marching Cubes algorithm[14] is used to triangulate the voxel model surface. The resulting mesh contains hundreds of thousands of triangles. Because these are many more than are necessary to represent the geometry with the accuracy of the reconstructed voxel model, the Progressive Meshes (PM) algorithm[15] is employed to reduce the number of triangles until the maximum distortion of the mesh corresponds to half the size of a reconstruction voxel. This way, triangle mesh accuracy is matched to the original reconstructed voxel model, and the number of triangles in the mesh is reduced to some ten thousand triangles.

3.2. Geometry Coding

Geometry model accuracy influences image prediction as well as coding bit-rate. To determine the point of best overall coding performance, the geometry model must be encoded at different bit-rates and, subsequently, with different accuracy. To progressively encode mesh connectivity and vertex coordinates simultaneously, the Embedded Mesh Coding (EMC) algorithm is used.[16] The EMC algorithm continuously refines mesh connectivity as well as vertex positions. This way, EMC allocates available coding bit-rate evenly between mesh connectivity and vertex positional information. When using EMC in conjunction with multi-view coding, geometry bit-rate can be continuously varied to allocate optimal bit-rate between geometry and prediction-error coding.

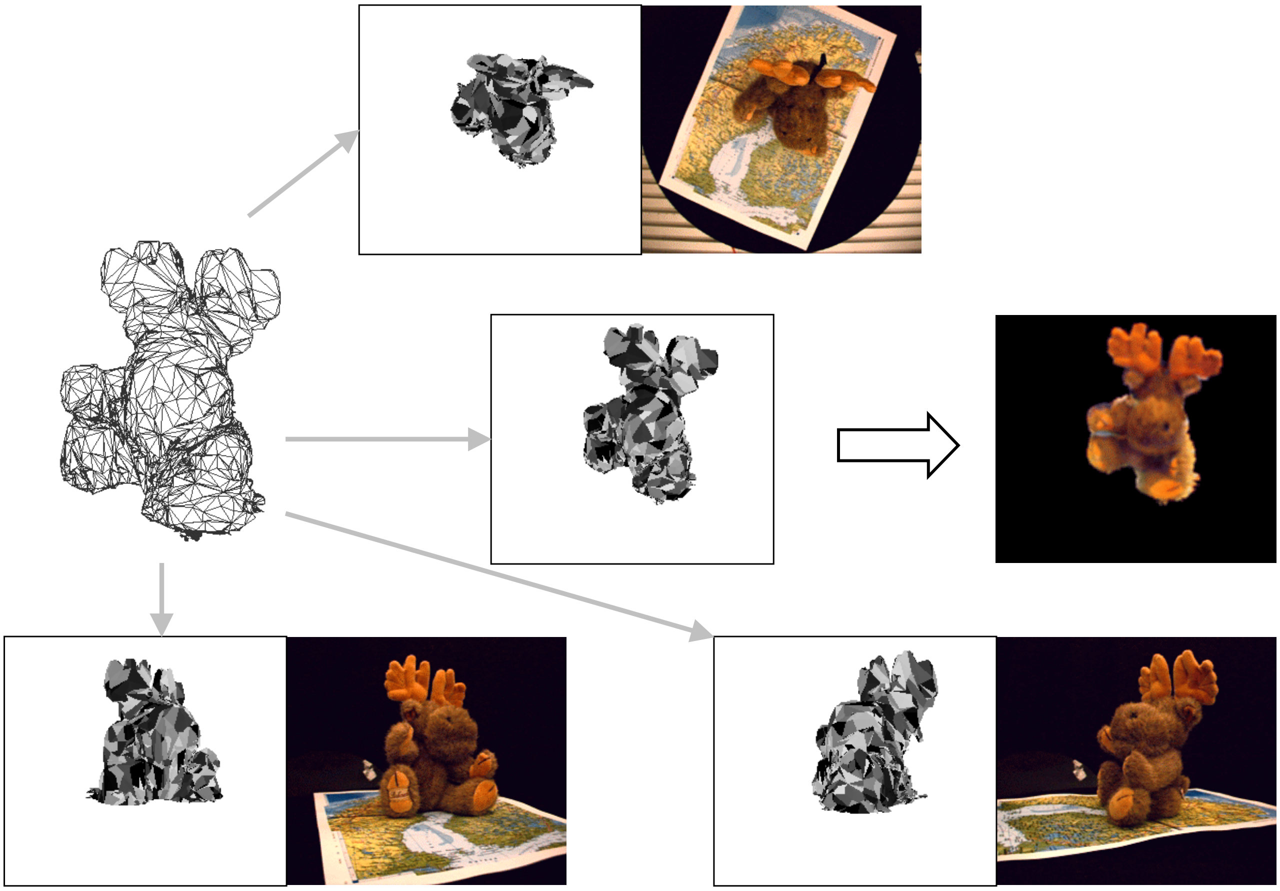


Figure 3. Geometry-based image prediction.

3.3. Image Warping based on 3-D Geometry

Fig. 3 depicts image prediction based on 3-D geometry by warping multiple images to the desired view. First, the geometry model is rendered for the image to be predicted. Each image pixel is assigned its corresponding point on the surface of the 3-D model by determining triangle index and relative position within the triangle. The geometry model is then rendered for all reference-image positions. For each pixel in the prediction image, the corresponding pixels in the reference images are sought using the triangle index and relative position. Pixels visible in several reference images are averaged. Pixels that are not visible in one reference image are detected, and partially occluded image regions are predicted using only those reference images that show the respective region. Because multiple reference images are used for prediction, the amount of completely occluded regions is small. Occluded areas are filled by interpolation using a resolution pyramid of the predicted image.

4. COMBINED PREDICTION SCHEMES

Coding performance of multi-view images can be improved by applying both depth map-based disparity compensation and geometry-based prediction. In the following, three different strategies how to combine depth map-based and geometry-based prediction are pursued.

4.1. Superimposed Prediction

To exploit available object geometry for multi-view image coding, the prediction results from depth map-based (Section 2) and 3-D geometry prediction (Section 3) can be overlaid. Because geometry-based prediction yields pixel-value estimates only for the image region within the silhouette of the projected model, pixels inside the silhouette are

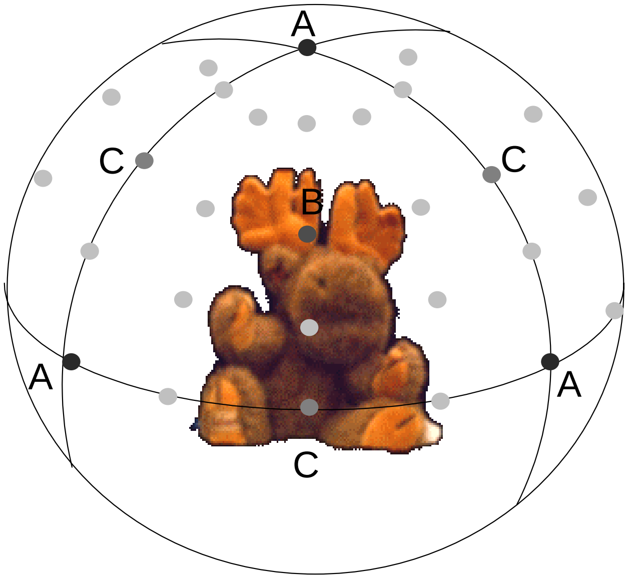


Figure 4. Hemispherical arrangement of image recording positions.

estimated using geometry-based prediction, and pixels outside the model's silhouette are predicted by depth map-based prediction. Any pixels is predicted using either geometry or depth-map information. For compression, the geometry model must be additionally encoded. In return, some depth-map blocks that are completely covered by the projected geometry model are not used for prediction and need not be coded. On the decoding side, the decision how to predict each pixel can be deduced solely from the available geometry model. To achieve improved overall coding performance, the expected increase in prediction quality must lead to sufficiently reduced residual-error bit-rate in order to offset the additional geometry bit-rate.

4.2. Silhouette-Switched Prediction

If the projected model silhouette does not exactly match the outline of the object in an image, a narrow halo around the object remains when taking the difference between the original and the predicted image. While silhouette mismatch may amount to only 1 to 2 pixels in width, the resulting prediction error can exhibit a steep gradient and represents noise of high spatial frequency. DCT-encoding this prediction error is very bit-rate expensive. To avoid silhouette-induced residual error, only image blocks that are completely covered by the projected object geometry are predicted using geometry in the silhouette-switched prediction scheme. The number of depth-map blocks to be encoded is the same as in superimposed prediction. The prediction mode decision can again be deduced from the encoded geometry model.

4.3. Block-Adaptive Prediction

A third way to combine depth map-based and geometry-based prediction consists of individually selecting the best prediction mode on an image-block basis.[17] While all image blocks outside the model silhouette can only be predicted based on depth maps, image blocks that are partially or entirely covered by the projected object geometry can be predicted using depth maps as well as geometry during coding. The SSE between both block-prediction candidates and the original block is measured, and the mode yielding the smaller error is selected. In addition to encoding the geometry model, the depth-map values outside the silhouette and some depth-map entries inside the silhouette, the prediction mode decision must be encoded for each image block touching the modeled object. The bit-rate overhead is larger than in superimposed or silhouette-switched prediction which must be compensated by improved prediction performance.

5. HIERARCHICAL IMAGE CODING ORDER

Depth maps as well as 3-D object geometry enable the prediction of images from arbitrary reference images. To achieve optimal coding performance, however, the images should be encoded in such an order that images are

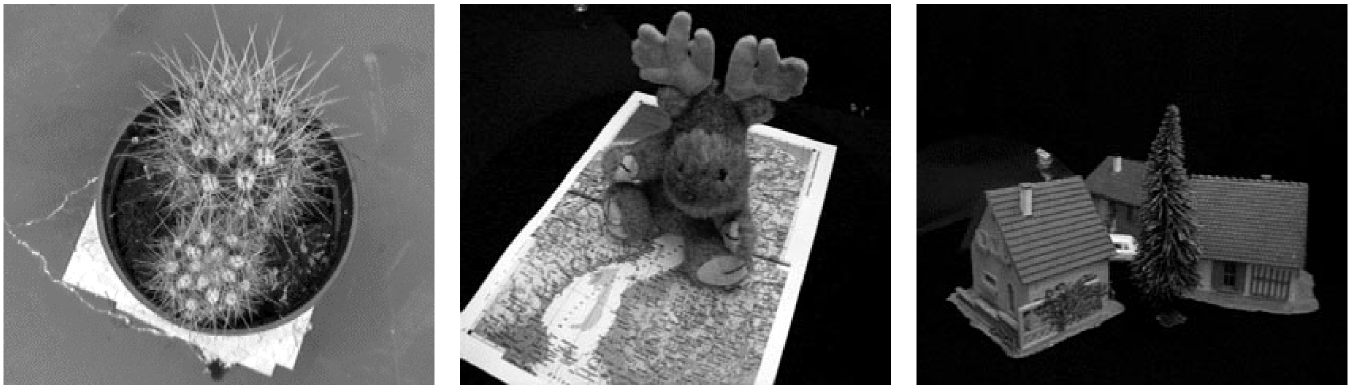
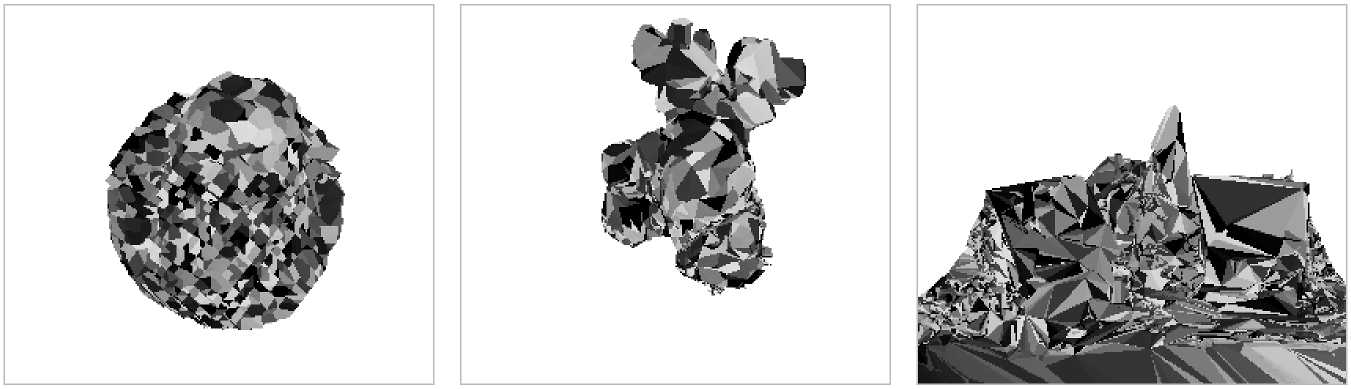


Figure 5. Images of the multi-view sets *Cactus*, *Moose*, and *Village*.



2144 triangles, 19083 bits

12633 triangles, 112246 bits

26801 triangles, 205414 bits

Figure 6. Reconstructed geometry models of the *Cactus*, *Moose*, and *Village*.

predicted most frequently from nearby reference images.

Fig. 4 illustrates the hierarchical image coding order that is applied to achieve image prediction over short distances while keeping fast access to arbitrary images.[6] First, all image recording positions are projected onto a sphere around the scene. As objects typically stand on a rigid, opaque platform during image acquisition, the recording directions gather on one hemisphere. The image closest to the pole of the sphere, and four images evenly spaced around the equator are intra-coded using the block-DCT scheme familiar from still-image compression (images A in Fig. 4). For each image, the quantization parameter Q is individually adjusted to ensure that the reconstructed image meets a preset minimum reconstruction quality $q_{\min}$. The five intra-coded images are arranged into four groups, each consisting of the polar and two equatorial images, subdividing the half-sphere into four quadrants. In each quadrant, the image closest to the center position (image B in Fig. 4) is predicted from the three corner images. The residual prediction error is DCT-encoded if image quality does not meet the desired reconstruction quality $q_{\min}$. The three images closest to the mid-positions of the quadrant's sides (images C in Fig. 4) are predicted and encoded likewise from the center and two corner images. After all quadrants have been considered, each quadrant is divided into four sub-regions. Because the corner images in each subdivided region are already encoded, they are available for prediction of the center and side images of the subdivided region. Again, the prediction error is encoded if necessary to meet $q_{\min}$. Subdivision continues recursively until all images are considered. This hierarchical image coding order allows access to multiple resolution levels of the image set. For rendering purposes, the multi-resolution representation of the images allows adjusting rendering quality to available computational resources.

The different prediction schemes are evaluated in conjunction with the above-described coder framework. In the following experiments, image coding order and residual-error encoding remain the same.

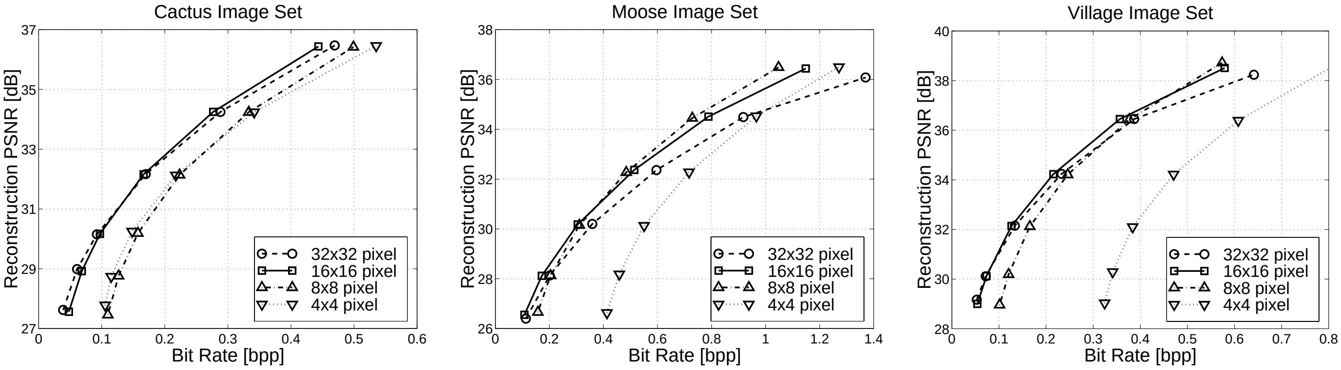


Figure 7. Rate-distortion curves for depth map-based image prediction using different depth-map block sizes.

6. RESULTS

Fig. 5 depicts images of the three test data sets. Each image set consists of 257 calibrated RGB images, recorded from a real-world scene using a turn table and a digital camera on a lever arm. The *Cactus* images are composed of 368 × 280 pixels, while the *Moose* and *Village* data sets contain 272 × 240-pixel images. For encoding, the images are converted to YC_bC_r color space, and the chrominance components are downsampled by a factor 2 along both horizontal and vertical direction.

Fig. 6 depicts the geometry models used for geometry-based prediction. The reconstructed *Cactus* geometry covers 17% of all image pixels. Due to small image calibration inaccuracies, the prickles are lost during model reconstruction, providing only approximate geometry for geometry-based prediction. The compact *Moose* object is more accurately reconstructed, extending over 22% of total image area. The *Village* geometry, finally, covers more than 55% of all image pixels. However, the *Village* model also consists of twice as many triangles as the the *Moose* geometry. The multi-view image sets represent different coding scenarios with regard to geometry accuracy and the amount of image area covered by the geometry model.

6.1. Depth Map-based Prediction

Depth map-based image prediction is used as reference to which the combined prediction schemes are compared. During depth-map estimation, the block size used for disparity compensation determines the resolution of the depth maps. While small blocks provide better prediction accuracy, larger block sizes require less bit-rate to encode the depth maps. Fig. 7 depicts depth map-based coding performance for different block sizes. Image reconstruction quality is expressed in decibel (dB) of the Peak-Signal-to-Noise-Ratio (PSNR), averaged over all multi-view images. The bit-rate includes depth-map encoding and is given in bits per pixel (bpp).

At low bit-rates, larger block sizes provide best compression. 32×32-pixel blocks allow encoding the *Cactus* image set at 0.039 bpp and 27.6 dB mean reconstruction PSNR. The *Moose* images are encoded best with 16 × 16-pixel blocks, yielding 26.6 dB PSNR at 0.108 bpp. At 29.2 dB PSNR, bit-rates of 0.055 bpp are attainable for the *Village* data set given 32 × 32-pixel blocks. Smaller depth-map blocks achieve better compression for high reconstruction quality. At 36.4 dB PSNR, the *Cactus* images are encoded with 0.44 bpp using 16 × 16-pixel blocks. For the *Moose* image set, blocks of 8 × 8 pixels yield 1.05 bpp total coding bit-rate at 36.5 dB PSNR. 8 × 8-pixel blocks also allow encoding the *Village* images with 0.57 bpp at 38.7 dB PSNR.

6.2. Performance Comparison

Fig. 8 depicts rate-distortion performance for depth map-based and geometry-based prediction alone as well as for superimposed, silhouette-switched, and block-adaptive prediction. Various different geometry accuracies and block sizes have been tested to determine optimal coding parameters. The curves in Fig. 8 reflect the individually best coding results.

Geometry-based prediction alone requires by far the highest bit-rate because the unmodeled background must be DCT-encoded for all images. Superimposed as well as silhouette-switched prediction also prove to be less efficient than depth map-based prediction alone. Only block-adaptive prediction yields a bit-rate reduction of 10% over depth

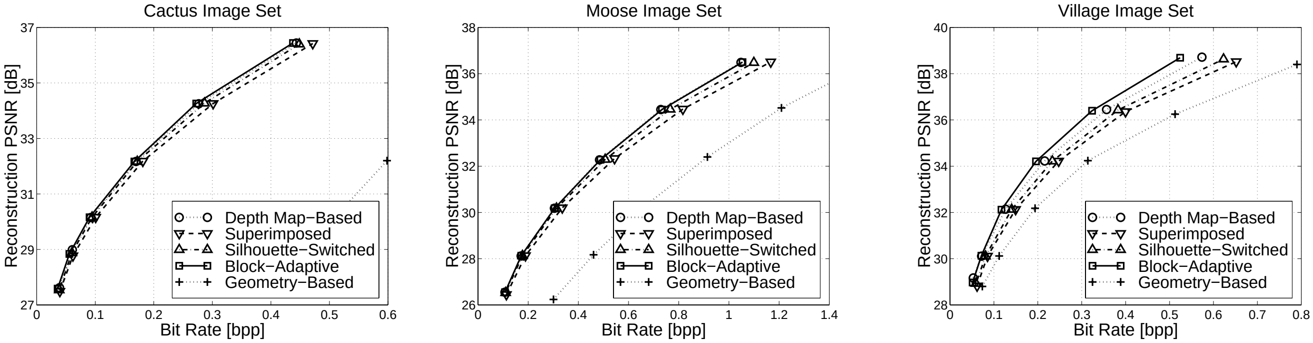


Figure 8. Best rate-distortion performance of the proposed image prediction schemes; for comparison, the coding results for depth map-based and geometry-based prediction are included.

map-based prediction at low bit-rates, encoding the *Cactus* images with 0.036 bpp at 27.6 dB reconstruction PSNR. For high bit-rates, the block-adaptive coding gain diminishes. The *Moose* images do not benefit from combined prediction at all, as depth map-based prediction slightly outperforms block-adaptive prediction at all bit-rates. For the *Village* image set, however, block-adaptive prediction is superior, especially towards high bit-rates. At 38.7 dB PSNR, block-adaptive prediction yields up to 9% better performance, encoding the data with 0.52 bpp.

The comparison of silhouette-switched and superimposed prediction suggests that silhouette inaccuracies indeed cause higher residual-error coding bit-rate. However, the disappointing results of both superimposed and silhouette-switched prediction indicate that geometry-based prediction accuracy is not, in general, significantly better than block-based disparity compensation. The depth-map values are generated by optimizing the resulting image prediction, while potential prediction capabilities are not considered during geometry model reconstruction. Additionally, depth maps are derived individually for each image, while the geometry model is reconstructed globally and has to comply to all images simultaneously.

Block-adaptive prediction, on the other hand, proves useful despite the additional bit-rate needed to encode the prediction mode per block. At low bit-rates, approximate object geometry can enhance coding performance if geometry coding bit-rate is small. More accurate geometry can be beneficial at medium and high bit-rates if the geometry model covers a substantial portion of all image pixels. An accurate geometry model that represents only a relatively small image region, however, requires too much bit-rate which cannot be offset by improved prediction quality. The superior performance of block-adaptive prediction over both other combined prediction schemes indicates that geometry-based prediction does not always provide better image predictions than depth map-based prediction. For the *Cactus* and the *Village* image sets, only about half of the blocks are selected to be predicted using 3-D geometry.

7. CONCLUSIONS

Three different schemes have been investigated how to combine depth map-based disparity compensation with geometry-based prediction. Different coding scenarios are evaluated by using three multi-view image sets depicting real-world scenes whose geometry can be reconstructed to different degrees of accuracy. Of the three combined prediction schemes examined, only block-adaptive prediction has been found to achieve better coding results than depth map-based prediction alone. For block-adaptive prediction, up to 10% bit-rate reduction has been observed.

The experimental results suggest that improved coding performance can be attained using combined depth map-based and geometry-based prediction if the geometry model requires low bit-rate (*Cactus*), or if the geometry model covers large parts of the image area to be coded (*Village*). At low bit-rates, approximate geometry appears to be sufficient to achieve enhanced coding results (*Cactus*), while accurate geometry is necessary at high bit-rates to obtain significantly better image predictions than depth map-based prediction (*Village*). Accurate geometry that covers only a small portion of the image set, however, might require too much bit-rate to achieve a reduction in total bit-rate (*Moose*).

REFERENCES

1. J. Lengyel, “The convergence of graphics and vision,” *Computer* **31**, pp. 46–53, July 1998.
2. M. Magnor and W. Heidrich, “Image-based rendering,” in *Principles of 3D Image Analysis and Synthesis*, H. N. B. Girod, G. Greiner, ed., pp. 232–241, Kluwer Academic Publishers, 2000.
3. M. Levoy and P. Hanrahan, “Light field rendering,” *Proc. ACM Conference on Computer Graphics (SIGGRAPH’96),* New Orleans, USA , pp. 31–42, Aug. 1996.
4. G. Miller, S. Rubin, and D. Ponceleon, “Lazy decompression of surface light fields for precomputed global illumination,” *Proc. Eurographics Rendering Workshop’98,* Vienna, Austria , pp. 281–292, Oct. 1998.
5. P. Lalonde and A. Fournier, “Interactive rendering of wavelet projected light fields,” *Proc. Graphics Interface’99,* Kingston, Canada , pp. 107–114, June 1999.
6. M. Magnor and B. Girod, “Data compression for light field rendering,” *IEEE Trans. Circuits and Systems for Video Technology* **10**, pp. 338–343, Apr. 2000.
7. X. Tong and R. Gray, “Coding of multi-view images for immersive viewing,” *Proc. IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP-2000),* Istanbul, Turkey **4**, pp. 1879–1882, June 2000.
8. C. Zhang and J. Li, “Compression and rendering of concentric mosaics with reference block codec (RBC),” *Proc. SPIE Visual Communications and Image Processing (VCIP-2000),* Perth, Australia **1**, pp. 43–54, June 2000.
9. M. Magnor and B. Girod, “Hierarchical coding of light fields with disparity maps,” *Proc. IEEE International Conference on Image Processing (ICIP’99),* Kobe, Japan **3**, pp. 334–338, Oct. 1999.
10. M. Magnor, P. Eisert, and B. Girod, “Model-aided coding of multi-viewpoint image data,” *Proc. IEEE International Conference on Image Processing (ICIP-2000),* Vancouver, Canada **2**, pp. 919–922, Sept. 2000.
11. Z. Zhang, “Image-based geometrically correct photorealistic scene/object modeling: A review,” *Proc. 3rd Asian Conference on Computer Vision (ACCV’98),* Hong Kong, China , pp. 340–349, Jan. 1998.
12. K. Hata and M. Etoh, “Epipolar geometry estimation and its application to image coding,” *Proc. IEEE International Conference on Image Processing (ICIP-99),* Kobe, Japan **2**, pp. 472–476, Oct. 1999.
13. P. Eisert, E. Steinbach, and B. Girod, “Multi-hypothesis volumetric reconstruction of 3-D objects from multiple calibrated camera views,” *Proc. International Conference on Acoustics, Speech, and Signal Processing ICASSP’99* Phoenix, USA , pp. 3509–3512, Mar. 1999.
14. W. E. Lorensen and H. E. Cline, “Marching cubes: A high resolution 3D surface construction algorithm,” *Proc. ACM Conference on Computer Graphics (SIGGRAPH’87),* Anaheim, USA , pp. 163–169, July 1987.
15. H. Hoppe, “Progressive meshes,” *Proc. ACM Conference on Computer Graphics (SIGGRAPH’96),* New Orleans, USA , pp. 99–108, Aug. 1996.
16. M. Magnor and B. Girod, “Fully embedded coding of triangle meshes,” *Proc. Vision, Modeling, and Visualization (VMV’99),* Erlangen, Germany , pp. 253–259, Nov. 1999.
17. P. Eisert, T. Wiegand, and B. Girod, “Model-aided coding: A new approach to incorporate facial animation into motion-compensated video coding,” *IEEE Trans. on Circuits and Systems for Video Technology* **10**, pp. 344–358, Apr. 2000.