Proc. 3rd International Workshop on Synthetic and Natural Hybrid Coding and 3-D Imaging (IWSNHC3DI'99)
Santorini, Greece, pp. 140-143, September 1999

Adaptive Block-based Light Field Coding

Marcus Magnor and Bernd Girod

Telecommunications Laboratory, University of Erlangen-Nuremberg
Cauerstrasse 7, 91058 Erlangen, Germany
Email: {magnor,girod}@nt.e-technik.uni-erlangen.de
Tel.: +49-9131-8528904
FAX : +49-9131-8528849

ABSTRACT

A compression scheme is presented that has been designed to efficiently code light fields. Based on video compression techniques, the light field image array is divided into image blocks. Different block-coding modes have been devised to exploit redundancy between images over a wide range of target bit rates. Prediction from multiple reference images further enhance coding efficiency. Rate-constrained mode selection is accomplished by employing Lagrangian optimization. The coder's operational rate-distortion performance is evaluated using different light fields. Depending on light-field scene characteristics and reconstruction quality, compression ratios up to 1000:1 are achieved.

1. INTRODUCTION

Light Field Rendering (LFR) has attracted much attention since its debut in 1996 [1, 2]. Based solely on conventional 2D images, its versatility and independence from object geometry makes LFR an attractive alternative to conventional geometry-based rendering techniques. LFR, however, requires immense amounts of image data, typically exceeding 1GByte. Data compression is therefore a vital issue in LFR for storing, processing and transmission of light fields. Vector quantization [1], DCT coding [3] and transform coding using spherical functions [4] have been employed to light field coding, yielding compression ratios less than 30:1.

In this paper a coder for light fields is presented that exploits intra-image redundancy as well as similarities between images. The coder is designed to provide high compression ratios for a wide range of reconstruction distortion values. The light field of a real-world object and two publically available synthetic light fields are used to verify coding performance. Compression ratios vary from 80:1 up to 1000:1, depending on reconstruction quality and light field characteristics.

2. LIGHT FIELD RENDERING

An illuminated, static object fills the surrounding space with light reflected off its surfaces, establishing the object's *light field* [5], also referred to as its *plenoptic function* [6]. If the object's light field is known, an observer's visual impression can be exactly reproduced for any viewpoint in space (x, y, z) and any viewing direction (θ, ϕ), as shown in Fig. 1. As light rays propagate through unobstructed, free space

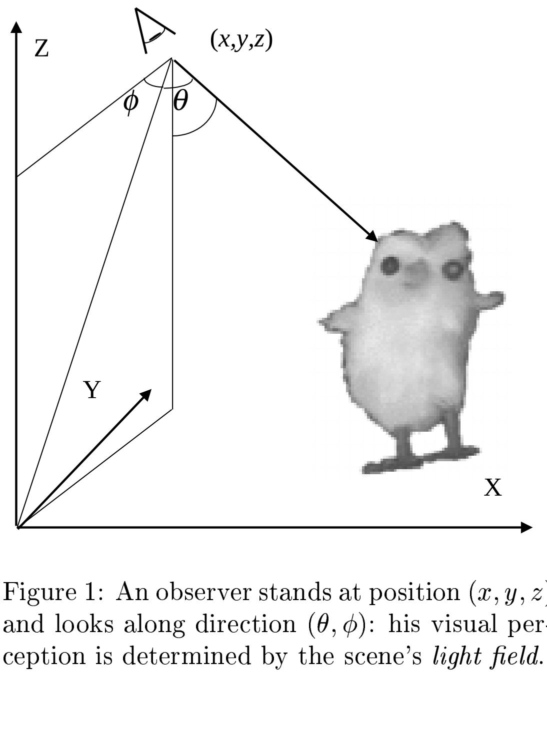
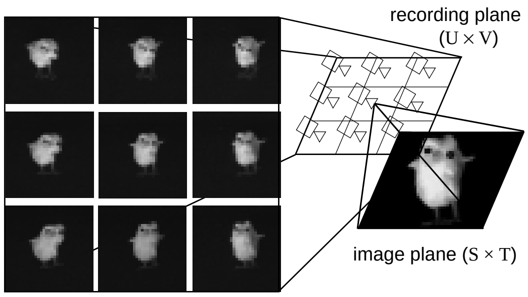
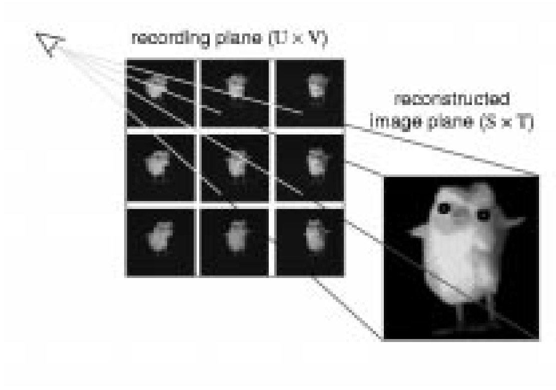


Figure 1: An observer stands at position (x, y, z) and looks along direction (θ, ϕ): his visual perception is determined by the scene's *light field.*



Recording a light field.



Rendering a new view from a light field.

Figure 3: Recording a light field in the two-plane parameterization consists of taking a number of conventional images from regularly spaced positions. Light Field Rendering requires to resample the image array by calculating the intersection points of viewing rays with the recording plane and the image plane.

unaltered, the light field's five degrees of freedom (x, y, z, θ, ϕ) can be reduced to four dimensions by suitable parameterization. In [1], the light field is described by rays of light that intersect two parallel planes, the *recording plane* (UV-plane) and the *image plane* (ST-plane) (Fig. 2). Each ray is parameterized by its intersection coordinates (u, v, s, t), establishing the light field's four-dimensional data structure. The planes are regularly sampled at $U \times V$ sampling points in the recording plane and $S \times T$ points in the image plane.

A natural object's light field can be readily recorded in the two-plane parameterization using cameras positioned at the sampling points in the recording plane: the light field resembles a 2D array of conventional 2D images (Fig. 3). To capture an object's light field from all sides, 6 recording planes are needed to enclose the object.

During rendering, rays are traced from the viewpoint (Fig. 3). The intersection coordinates with the UV-plane (u, v) and the ST-plane (s, t) can be swiftly calculated. Because intersection coordinates do not, in general, coincide with image recording positions, pixels from 4 images closest to the intersection point (u, v) are interpolated to avoid aliasing effects (quadralinear interpolation).

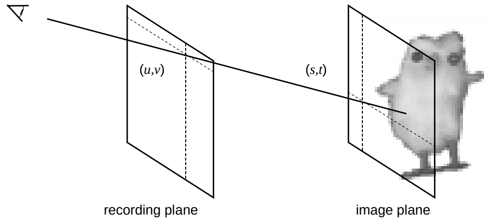


Figure 2: Two-plane parameterization of the light field: a viewing ray is parameterized by its intersection coordinates with the recording plane (u, v) and the image plane (s, t).

Rendering quality depends on image plane resolution $(S \times T)$, as well as on the number of images taken from the recording plane $(U \times V)$. For example, the *Buddha* and *Dragon* light fields used to evaluate the coder both consist of an array of 32×32 24-bit RGB images, each containing 256×256 pixels. These 1024 images cover only one of the six recording planes necessary for covering the entire scene: light fields must consist of several thousand images to achieve acceptable rendering results from all viewing positions.

To attain high compression ratios, similarities between light-field images have to be exploited, introducing dependencies among images during decoding. The coder described in the following codes a number of light-field images without reference to other images to enable fast decoding and rendering at reduced resolution.

3. THE CODER

Prior to coding, the light-field images are transformed to YUV-color space, and the chrominance signal is downsampled by a factor of 2 in both the horizontal and vertical directions.

Light-field compression starts by selecting a subset of images from the light-field array. These *I-images* are solely DCT-coded, as is common in still-image compression. Each I-image can be decoded without reference to other images, and by selecting I-images evenly distributed over the entire light-field image array, the I-images constitute a subsampled representation of the original data set.

I-images serve as reference images to code the remaining light-field images, the predicted or *P-images*. Several neighboring I-images can be used to predict a P-image. It was experimentally found that prediction from 4 I-images yields the best rate-distortion characteristics (multi-frame prediction [7]). Motion between light field images is reduced to one dimension, because the scene is static and image recording positions are known. Only *disparity* (parallax motion) has to be compensated.

The P-images are divided into blocks of 16×16 pixels. Each block is coded by applying the most efficient of eight different block-coding modes:

- INTRA: The block is DCT-coded, i.e., without reference to other images (3+x bits).
- NODISP: a block is copied from one of the 4 I-images without compensating for disparity motion (3+2 bits).
- CLOSEST: a block is copied from the nearest I-image with no disparity motion compensation (3 bits).
- AVERAGE: from all 4 I-images, blocks are averaged without disparity compensation (3 bits).
- DISP: a disparity-compensated block from one of the 4 I-images is copied (3+2+1..6 bits; disparity is coded using a fixed Huffman table).
- For the AVERAGE, NODISP and DISP modes, the residual error can additionally be DCT-coded, leading to 3 additional coding modes (each 3+x bits).

3 bits per block are needed to specify which mode is used to code a block. Some modes require additional bits to specify from which of the 4 reference images to compensate, or to code DCT coefficients. The presented block coding modes have been chosen because experiments show that these modes allow efficient coding of P-image blocks over a wide range of target bit rates.

Because the block-coding modes have different operational rate-distortion characteristics, the most efficient mode must be found for each block. The coder adapts to a desired target bit rate and solves the rate-constrained optimization problem by minimizing the Lagrangian rate-distortion functional [8]:

$$\min_{i=1..8} \{D_i + \lambda R_i\}$$

For each P-image block, all coding modes $i = 1,..,8$ are considered. The resulting distortions D_i and bit rates R_i are measured, and the cost function $J_i = D_i + \lambda R_i$ is calculated using a preset and fixed value for the Lagrangian multiplier λ. The Summed Squared Error (SSE) between original and coded block serves as distortion measure D_i. A block is coded using the mode that results in the smallest cost value J_i. The parameter λ and the DCT quantizer stepsize parameter Q control image reconstruction quality and compression efficiency. Experimental results in [8] show that the relationship between Q and λ can be approximated by

$$Q = \sqrt{\frac{\lambda}{0.85}}.$$

While P-images require much fewer bits than I-images, overall coding efficiency depends on the number of I-images distributed over the image array. For a given reconstruction quality parameter value λ, different numbers of I-images have to be tested to find the optimal I-image density over the image array.

4. CODING PERFORMANCE

The proposed coder was tested using two synthetic light fields (*Dragon, Buddha*)[1] and one light field of a natural object (*Chick*). The Buddha and Dragon light fields both consist of 32×32 images of 256×256 24-bit RGB pixels each, amounting to 192 MBytes. The Chick light field was recorded by taking 17×17 images using a CCD camera (256×256 24-bit RGB pixels) on a robot arm, totalling 54 MBytes.

The coder's rate-distortion curves are depicted in Fig. 4. Light-field reconstruction distortion is measured as the averaged PSNR of the luminance signal over all light-field images. Different numbers of I-images were tested to find the best percentage of reference images. Because the Dragon and Buddha light-field images are computer-rendered, disparity motion can be accurately compensated. At 36 dB PSNR average reconstruction distortion, the Dragon light field is compressed to 893 kBytes (0.11 bits per pixel). The Buddha light field is even coded with 434 kBytes (0.053 bpp) at 40 dB PSNR.

[1] Available at: www-graphics.stanford.edu/software/lightpack/lifs.html.

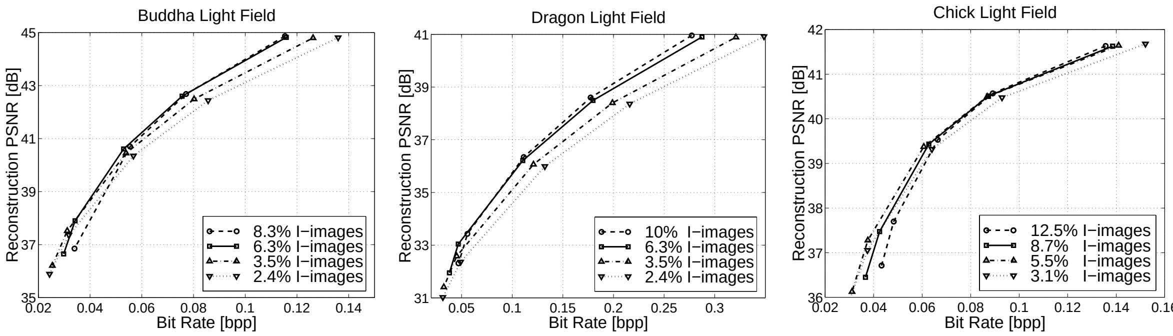


Figure 4: Rate-Distortion measurements for different I-image densities (I-image density is expressed as the percentage of I-images to the total number of images in the light field).

The Chick light field is recorded from a natural object, so camera calibration and recording positions exhibit small inaccuracies. 39 dB PSNR reconstruction distortion are achieved after compressing the light field to 140 kBytes (0.061 bpp), showing that good coding results are also obtained if recording geometry is not exactly known.

All light fields show visible quantization and block artifacts only at the lowest bit rate tested (λ=1000). At the highest tested bit rate (λ=10), compressed light-field images cannot be distinguished from the original. Intermediate compression ratios (λ=50,100,500) result in a gradual loss of fine details. Because lowpass filtering has to be applied during rendering to avoid aliasing effects (quadralinear interpolation), the loss of some high-frequency image components during compression does not affect rendering performance.

5. CONCLUSION

A coding scheme for light fields has been presented that yields compression ratios from $80:1$ to $1000:1$ at medium to high reconstruction quality, easing light field storage, transmission and processing requirements. The coder has been shown to be applicable to synthetic as well as natural light-field scenes. Much higher compression ratios than previous light field compression attempts [1, 3, 4] are attained, at the cost of a more complex decoding process. Interactive rendering frame rates can still be achieved from compressed light fields by rendering from I-images first and refining the rendered view afterwards using P-images. Taking increased decoding times into account, even higher compression rates can be achieved by coding light-field images hierarchically, using disparity maps for prediction [9].

6. REFERENCES

[1] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH 96 Conference Proceedings*, pages 31–42, August 1996.

[2] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. The lumigraph. In *SIGGRAPH 96 Conference Proceedings*, pages 43–54, August 1996.

[3] G. Miller, S. Rubin, and D. Ponceleon. Lazy decompression of surface light fields for precomputed global illumination. In *Eurographics Rendering Workshop*, pages 281–292, 1998.

[4] Tien-Tsin Wong, Pheng-Ann Heng, Siu-Hang Or, and Wai-Yin Ng. Image-based rendering with controllable illumination. In *Eurographics Rendering Workshop*, pages 13–22, June 1997.

[5] A. Gershun. The light field. *J. Math Phys.*, 18:51–151, 1939.

[6] E. H. Adelson and J. R. Bergen. The plenoptic function and the elements of early vision. *Computational Models of Visual Processing*, 1991.

[7] T. Wiegand, X. Zhang, and B. Girod. Motion-compensating long-term memory prediction. In *Proc. ICIP-97, Santa Barbara*, October 1997.

[8] G. Sullivan and T. Wiegand. Rate-distortion optimization for video compression. *IEEE Signal Processing Magazine*, pages 74–90, November 1998.

[9] M. Magnor and B. Girod. Hierarchical coding of light fields with disparity maps. In *Proc. ICIP-99, Kobe*. IEEE Signal Processing Society, 1999. accepted.