IEEE Transactions on Circuits and Systems for Video Technology

vol. 10, no. 3, pp. 338-343, April 2000

# Data Compression for Light Field Rendering

Marcus Magnor and Bernd Girod, Fellow IEEE

Telecommunications Laboratory

University of Erlangen-Nuremberg

Cauerstrasse 7, 91058 Erlangen, Germany

{magnor|girod}@lnt.de

**Abstract**

Two light-field compression schemes are presented. The codecs are compared with regard to compression efficiency and rendering performance. The first proposed coder is based on video-compression techniques that have been modified to code the four-dimensional light-field data structure efficiently. The second coder relies entirely on disparity-compensated image prediction, establishing a hierarchical structure among the light-field images. Coding performance of both schemes is evaluated using publicly available light fields of synthetic as well as real-world scenes. Compression ratios vary between 100:1 and 2000:1, depending on reconstruction quality and light-field scene characteristics.

## I. Introduction

Three-dimensional scenes are generally visualized using conventional two-dimensional display techniques. Rendering photorealistic 2-D views of 3-D objects at interactive frame-rates is therefore a vital aspect in 3-D video technology. Commonly, 3-D rendering relies on geometry description, texture information and illumination specifications. Rendering quality as well as frame-rate are determined by model complexity and computation time spent on simulating global illumination effects. Trading off rendering frame-rate vs. image quality is often inevitable, and 3-D geometry acquisition from real-world objects can prove problematic if surfaces are not well-defined (e.g. hair, fur, smoke).

Light Field Rendering (LFR) has been proposed as an alternative rendering technique [1], [2]: A set of conventional 2-D images is used to render arbitrary views of a static 3-D scene. The images capture the light distribution or *light field* [3] around the scene. Neither scene complexity nor illumination effects influence rendering frame-rate. Because 2-D images can be taken from any object, no constraints on scene content apply in LFR. Rendering quality depends solely on the number of images available. Unfortunately, LFR requires tens to hundreds of thousands of images to ensure photorealistic rendering results from any viewpoint. Compression is necessary to transmit the data as well as to fit all information into local memory during rendering, while fast access to arbitrary light-field segments is crucial to enable interactive rendering rates.

This paper is concerned with the compression aspects of LFR. Vector quantization [1], DCT-coding [4] and transform coding using spherical functions [5] have been applied to light-field compression. Compression ratios below 30:1 have been achieved, allowing only a fraction of the image set needed for high-quality rendering to fit into local memory. Much higher compression rates are still needed to transmit light fields in acceptable times, e.g., over the Internet.

In this paper, two schemes are presented to compress light fields more efficiently, both for transmission and rendering on standard hardware. The video compression-based coder as well as the disparity-compensating codec are designed to provide high compression ratios at medium to high image reconstruction qualities. The first coder presented has only modest memory requirements and features fast decoding of recorded light-field segments, achieving interactive rendering rates, while the disparity-compensating coder incrementally refines the light field during decoding and predicts intermediate (missing) light-field images, greatly enhancing rendering quality. Both coders' rate-distortion characteristics are evaluated for two synthetic light fields and one light field of a real-world object.

## II. Light Field Rendering

The transparent space around an illuminated object is filled with light reflected off the object's surfaces. This *light field* can be sampled by recording conventional images of the object from many viewpoints. In [1], [2], the recording positions of these images are arranged on a plane in a regular grid, parameterizing the light field as a 2-D array of 2-D images (Fig. 1). For photorealistic rendering, an object's light field must be sampled densely enough such that maximum parallax (*disparity*) between adjacent images does not exceed one pixel. Otherwise, aliasing occurs, and, if images are interpolated [1], blurred rendered images result. The number of images required to guarantee less-than-one-pixel disparity between adjacent images is proportional to the images' resolution. Simple geometrical considerations show that to accurately sample the complete light field of a scene with, e.g., $256 \times 256$-pixel images, more than 200,000 images are needed: the attempt to acquire critically sampled light fields is futile. Thus, physically recorded light fields always constitute only a sub-sampled representation of the complete light-field information. But even sub-sampled light fields may contain many thousand images to achieve decent rendering results, still yielding several Gigabytes of imagery. Data compression remains a fundamental issue for any light-field rendering application.

The first coder presented allows efficient compression of all recorded light-field data, featuring fast access to arbitrary image segments during decoding for interactive rendering rates. The second codec described in this paper has been designed to provide disparity-compensated intermediate (missing) images, augmenting the originally recorded light field by additional in-between images to enhance rendering quality.

## III. Video Compression-based Light Field Coding

The first coder extends the approach taken by video-compression schemes to the 4-D data structure of light fields: the light-field images are divided into blocks, and each block is coded using one out of several block-coding modes. The coding mode for each block is selected separately. The proposed codec will be referred to as the *V-coder* [6].

As is commonly done in video coding, all light-field images are first transformed to YUV color space, and the chrominance signal components are downsampled by a factor of 2 in horizontal and vertical direction. The coding process starts by selecting a number of images from the light-field array to be coded as intra- or *I-images*. These are compressed exploiting solely redundancy within the images using the block-based discrete cosine transform (DCT) and coefficient quantization. By selecting I-images evenly distributed over the entire light-field image array, the set of I-images constitutes a subsampled representation of the recorded light-field image set (Fig. 1).

I-images serve as reference for coding the remaining light-field images, the predicted or *P-images*. The light-field data structure exhibits favorable characteristics that can be exploited for P-image compression: when comparing two adjacent light-field images, a point on the surface of the depicted rigid object often appears in both images, but at different positions. This displacement is the point's parallax or *disparity*. From known image recording positions, the disparity *direction* between two images can be inferred, and only the amount of disparity needs to be coded. Additionally, several I-images are available to serve as reference to predict a P-image. The use of multiple reference images enhances coding performance and is closely related to multi-frame prediction in video coding [7].

The P-images are divided into square blocks of $16 \times 16$ pixels. Eight block-coding modes have experimentally been found to efficiently exploit light-field characteristics over a wide range of target bit-rates:

- CLOSEST: A $16 \times 16$ pixel block is copied from the nearest I-image with no disparity compensation (resembling P-frame prediction in video coding from the preceeding I-frame with no motion vector).
- NODISP: Similar to the CLOSEST-mode, but any one of the reference I-images can be specified for compensation.
- DISP: A reference I-image is specified, and a disparity-shifted block is copied (similar to B-frame prediction in video coding).

- AVERAGE: From all reference I-images, the blocks corresponding to no disparity shift are averaged (similar to B-frame prediction by averaging with no motion vector).
- RESIDUAL ERROR: For the AVERAGE, NODISP and DISP modes, the residual error can additionally be DCT-coded, leading to 3 more modes to code an image block.
- INTRA: If the block cannot be predicted well from surrounding I-images, the P-image block is DCT-coded, i.e., without reference to any I-image.

The block-coding modes have different operational rate-distortion characteristics. INTRA-coding a block generally requires the highest bit-rate, while for the CLOSEST and AVERAGE modes only the mode itself needs to be specified. The rate-constrained optimization problem of which coding mode to choose for each block can be elegantly solved using the method of Lagrangian multipliers [8]:

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

For each block, all coding modes $i = 1,..,8$ are considered. The resulting distortions $D_i$ and bit-rates $R_i$ are measured, and the Lagrangian cost function $D_i + \lambda R_i$ is calculated using a preset and fixed value for the Lagrangian multiplier $\lambda$. Each block is coded using the mode that results in the smallest Lagrangian cost value. The parameter $\lambda$ controls image reconstruction quality and compression efficiency by weighting the bit-rate $R$ in the cost function. A small value for $\lambda$ results in high reconstruction quality and vice versa.

The DCT quantizer parameter $Q$ determines reconstruction accuracy of the DCT-coded residual error and of INTRA-coded blocks. The value for $Q$ depends on the quality parameter $\lambda$. Experimental results in [8] show that the optimal relationship between $Q$ and $\lambda$ can be approximated by

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

The values of $\lambda$ and $Q$ are set prior to coding. On an SGI-O2 workstation, the V-coder takes about 4 seconds to code a single P-image consisting of 256 × 256 pixels.

While P-images require much fewer bits to code than I-images, overall coding efficiency depends on the number of I-images that are evenly distributed over the light-field image array. For a given quality parameter value $\lambda$, different numbers of I-images have to be tested to find the optimal I-image distribution over the light-field array. The optimal number of I-images is fixed regardless of the overall number of light-field images, because any additional light-field image can efficiently be P-image coded.

Prior to rendering, the small number of I-images is decoded and kept in local memory. Because at most a couple of hundred light-field images are coded as I-images, memory requirements are modest. The reconstructed I-images provide instantaneous access to a low-resolution version of the light field, allowing very fast rendering rates at reduced rendering quality. Tab. I depicts measured decoding times to reconstruct a P-image block ($16\times16$ pixels) for all block-coding modes. All measurements mentioned in this paper were conducted on an SGI-O2 workstation with 192 MBytes RAM. Interactive rendering rates are attainable as even the computationally most expensive decoding mode, the AVERAGE & RESIDUAL-ERROR block-coding mode, requires no more than 700 microseconds to decode a $16\times16$-pixel block.

## IV. Disparity-Compensating Light-Field Coding

The *disparity* is a scalar value associated with each pixel describing the amount of shift of the pixel's corresponding object surface point in neighboring light-field images. Due to the planar light-field recording geometry, the direction of shift can be inferred from the images' known recording positions (disparity direction). An image's *disparity map* then is an array of scalar values that lists the amount of parallax shift for each pixel in a light-field image. The second light-field coder presented relies entirely on disparity-compensating light-field images. It will be denoted the *D-coder* [9].

Prior to coding, disparity maps have to be derived for all light-field images. To retrieve disparity information, all images are divided into image blocks. Because all light-field images are arranged in a regular grid of equal spacing, an image block's true disparity magnitude is identical when comparing the block to any of its four directly neighboring images along the respective disparity direction. To find the amount of disparity shift for each block, a number of disparity values $d$ within a predefined search range are considered, Fig. 2. For each disparity value, the corresponding blocks from all four neighboring images are extracted, averaged and compared to the original block. The disparity value resulting in the smallest prediction error (e.g., the Sum-Squared-Error criterion) is chosen as the block's disparity magnitude. The disparity map for a $256 \times 256$-pixel image based on $16 \times 16$-pixel blocks is derived in less than 1 second on an SGI-O2 workstation.

To predict an image (target image), its disparity map is first estimated by compensating the disparity maps of several already coded images (reference images), Fig. 3. The estimated disparity map is low-pass filtered, filling any holes by interpolation, and all reference images are disparity-compensated. The estimated images are averaged to yield the target image's prediction. Besides optimal prediction of any light-field image from its surrounding images, the described disparity compensation approach allows estimating intermediate (missing) images. By adapting the disparity maps' block size (e.g., $4\times4$, $8 \times 8$, $16 \times 16$, or $32 \times 32$ pixels), bit-rate can be optimally allocated between disparity information and residual-error coding.

Prior to coding, a minimum reconstruction quality parameter $q_{min}$ is set. Image quality is measured as the peak-signal-to-noise ratio (PSNR) of the image's luminance signal. First, the image array's four corner images (positions A in Fig. 4) are intra-coded, identical to the I-images described in the previous section. For each image, the DCT quantization parameter is individually selected to ensure that the reconstructed image meets the reconstruction quality $q_{min}$. The corner images' disparity maps are Huffman-coded applying a fixed table. From the four reconstructed corner images and their disparity maps, the center image and its disparity map (position B in Fig. 4) is predicted. If the disparity-compensated image meets the reconstruction criterion $q_{min}$, no information regarding the center image is coded, Fig. 7. Otherwise, the center image is again predicted using its own disparity map, which is then Huffman-coded. If image quality still doesn't suffice, the residual error is additionally DCT-coded. The DCT quantizer level is adjusted to yield minimum bit-rate for the required image quality

$q_{min}$. Then, the four middle images on the array sides (positions C in Fig. 4) are predicted from the reconstructed center image and the two closest corner images. As for the center image, the residual error and disparity maps are coded, if necessary.

At this point, 9 light-field images spanning the entire recording plane are available. The image array is now divided into four quadrants. The four corner images of each quadrant are already coded and, as before, the center and side images in each quadrant can be predicted. The algorithm keeps recursing through the quadtree structure until all images are compressed. On an SGI-O2 workstation, the D-coder takes about 2 seconds per image to code the light field.

For rendering, all light-field images' disparity maps are decoded and stored in local memory. The 4 intra-coded corner images and as many light-field images from subsequent hierarchy levels as fit into local memory are reconstructed. As is the case for the I-images of the V-coder, these images are instantaneously accessible, enabling interactive rendering from a sub-sampled light-field representation. Decoding an image segment from the next-higher hierarchy level resembles the AVERAGE & RESIDUAL-ERROR block-coding mode of the V-coder: pixels from 3 or 4 reference images need to be copied and averaged, and possibly the residual error has to be decoded and added. As for the V-coder (Tab. I), this next-higher image level can still be decoded at interactive rendering rates. Due to the D-coder's hierarchical coding structure, access to still higher hierarchy-level image blocks requires decoding multiple images of in-between levels. While these images are not accessible at interactive rates, they can be used to improve rendered views during standstill. The advantage of the D-coder is that the decoder can locally refine the light field by estimating disparity-compensated intermediate light-field images that were not originally recorded. These intermediate images greatly enhance rendering results [10].

## V. Coding Performance

Both coding schemes were validated using two synthetic light fields (*Buddha, Dragon*[1]) and one light field recorded from a real-world object (*Chick*[2]), Fig. 5. Both synthetic light fields consist of 32 × 32 images, each containing 256 × 256 24-bit RGB pixels, amounting to 192 MBytes. Due to their computer-generated nature, the *Dragon* and *Buddha* light-field images exhibit perfect disparity relations. The *Chick* light field was recorded using a 256 × 256-pixel color-CCD camera on a robot arm, taking images from 17 × 17 equally-spaced locations (54 MBytes). The *Chick* light-field images show electronic noise, optical distortions and limited accuracy in image recording positions.

Light-field reconstruction quality is measured as the averaged PSNR-value of the luminance signal (Y component) over all light-field images. The light fields were coded for several reconstruction quality settings $q_{min}$ (D-coder), respectively different values of the Lagrangian multiplier $\lambda$ (V-coder).

Both coders' R-D curves are shown in Fig 6. For the V-coder, the number of I-images was varied to determine the best distribution of reference images over the light field. The V-coder compresses the *Dragon* light field down to 893 KBytes (0.11 bits per pixel) at 36 dB average reconstruction distortion. The *Buddha* light field is even more efficiently coded, requiring 434 KBytes (0.053 bpp) at 40 dB PSNR. The *Chick* light field can be compressed to 86.6 KBytes (0.037 bpp) at 37 dB. Attainable compression ratios depend on object characteristics, such as apparent size, texture variations and geometrical complexity, as well as on the distance between image recording positions.

The D-coder was tested for different disparity map block sizes. The results are also depicted in Fig. 6. The *Dragon* light field requires 562 KBytes (0.068 bpp) to code at 36 dB, while the *Buddha* light field is compressed to 223 KBytes (0.027 bpp) at 40 dB reconstruction quality. The *Chick* light field is coded with 73.8 kBytes (0.031 bpp) at 37 dB.

Comparing both coders' operational R-D curves, it is apparent that the D-coder compresses the

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

[2] URL: www.lnt.de/˜magnor/chick.tar.gz

synthetic light fields more efficiently than the V-coder, because many light-field images can be predicted well enough by disparity compensation and need no additional residual-error information (Fig. 7). For the *Chick* light field, both coders perform about equally well: in this case, the smaller total number of images, greater recording distances between images, deviations in image positions and electronic noise limit the D-coder's performance.

## VI. Conclusions

Two compression schemes were presented that are applicable for light-field rendering purposes. Both coders feature compression factors on the order of $10^2 - 10^3$ at medium to high image reconstruction-quality, easing capacity requirements to store light fields and speeding up transmission, e.g., over the Internet.

The V-coder introduces two hierarchy levels among the light-field images (I- and P-images). The presented block-coding modes allow access to P-image segments at high decoding rates, enabling interactive rendering. As in video coding, blocking artifacts become apparent at very high compression ratios. The D-coder exploits inter-image redundancy by establishing multiple hierarchy levels of dependencies between light-field images. For standstill views, or if sufficient memory is available, the decoder is able to yield near-photorealistic rendering results by augmenting the original light field with disparity-compensated intermediate images that were never physically recorded. As multiple images are disparity-compensated and averaged to predict light-field images, blocking artifacts do not occur. At very low bit-rates, the reconstructed light-field images merely lose detail (blurring).

The obtained compression factors should suffice for almost any recordable light field and LFR application. LFR performance can be further improved by accelerating the rendering process. E.g., rendering-supporting hardware can be utilized if approximate 3-D geometry of the light-field object can be determined [11]. Available geometry information can also be efficiently used for light-field coding [12]. Future work will focus on joint coding of light-field geometry and texture maps in conjunction with real-time, photorealistic rendering from light-field data.

## References

[1] M. Levoy and P. Hanrahan, "Light field rendering", in *Computer Graphics (Proc. SIGGRAPH96)*, Aug. 1996, pp. 31–42.

[2] S. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen, "The lumigraph", in *Computer Graphics (Proc. SIGGRAPH96)*, Aug. 1996, pp. 43–54.

[3] A. Gershun, "The light field", *J. Math Phys.*, vol. 18, pp. 51–151, 1939.

[4] G. Miller, S. Rubin, and Du. Ponceleon, "Lazy decompression of surface light fields for precomputed global illumination", in *Proc. Eurographics 1998*, June 1998, pp. 281–292.

[5] T.-T. Wong, P.-A. Heng, S.-H. Or, and W.-Y. Ng, "Image-based rendering with controllable illumination", in *Proc. Eurographics 1997*, June 1997, pp. 13–22.

[6] M. Magnor and B. Girod, "Adaptive block-based light field coding", *Proc. International Workshop on Synthetic-Natural Hybrid Coding and 3-D Imaging IWSNHC3DI'99,* Santorini, Greece, Sept. 1999.

[7] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction", *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 70–84, Feb. 1999.

[8] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression", *IEEE Signal Processing Magazine*, pp. 74–90, Nov. 1997.

[9] M. Magnor and B. Girod, "Hierarchical coding of light fields with disparity maps", *Proc. International Conference on Image Processing ICIP-99,* Kobe, Japan, Oct. 1999.

[10] W. Heidrich, H. Schirmacher, H. Kück, and H.-P. Seidel, "A warping-based refinement of lumigraphs", *Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization*, 1998.

[11] P. Eisert, E. Steinbach, and B. Girod, "Multi-hypothesis volumetric reconstruction of 3-D objects from multiple calibrated camera views", in *Proc. ICASSP-99,* Phoenix, USA, Mar. 1999.

[12] B. Girod, P. Eisert, M. Magnor, E. Steinbach, and T. Wiegand, "3-D image models and compression - synthetic hybrid or natural fit ?", *Proc. International Conference on Image Processing ICIP-99,* Kobe, Japan, Oct. 1999.

## List of Figures

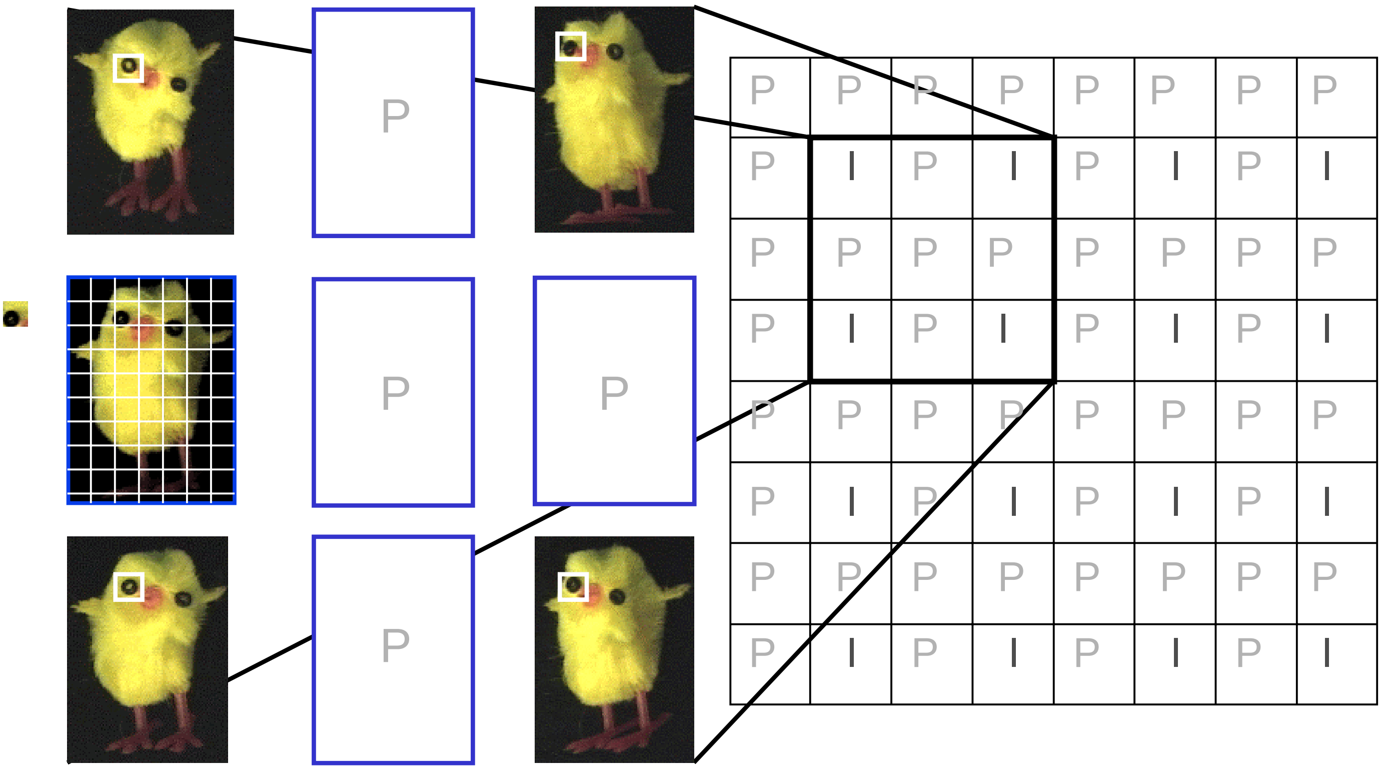

Fig. 1. The light field data structure resembles a 2-D array of 2-D images; the V-coder selects a subset of light-field images to be coded as *Intra-* or I-images that serve as reference images for coding the remaining *P-images*; P-images are subdivided into blocks, and each block is coded using one of eight different block-coding modes that make use of nearby I-images for prediction.

| Coding Mode | t ($Q = 2$) | t ($Q = 5$) | t ($Q = 31$) |
|---|---|---|---|
| INTRA | 316 $\mu$s | 239 $\mu$s | 215 $\mu$s |
| NODISP & RES | 439 $\mu$s | 338 $\mu$s | 308 $\mu$s |
| DISP & RES | 459 $\mu$s | 359 $\mu$s | 339 $\mu$s |
| AVG & RES | 700 $\mu$s | 616 $\mu$s | 577 $\mu$s |
| CLOSEST | | 218 $\mu$s | |
| NODISP | | 174 $\mu$s | |
| DISP | | 223 $\mu$s | |
| AVERAGE | | 454 $\mu$s | |

TABLE I

Overall decoding times to access and reconstruct a $16 \times 16$-pixel block (measured on an SGI-O2 workstation, 175 MHz, R 10000 CPU); decoding times of the first 4 modes depend on the DCT quantizer parameter $Q$ because the number of DCT coefficients for residual-error coding varies with $Q$.

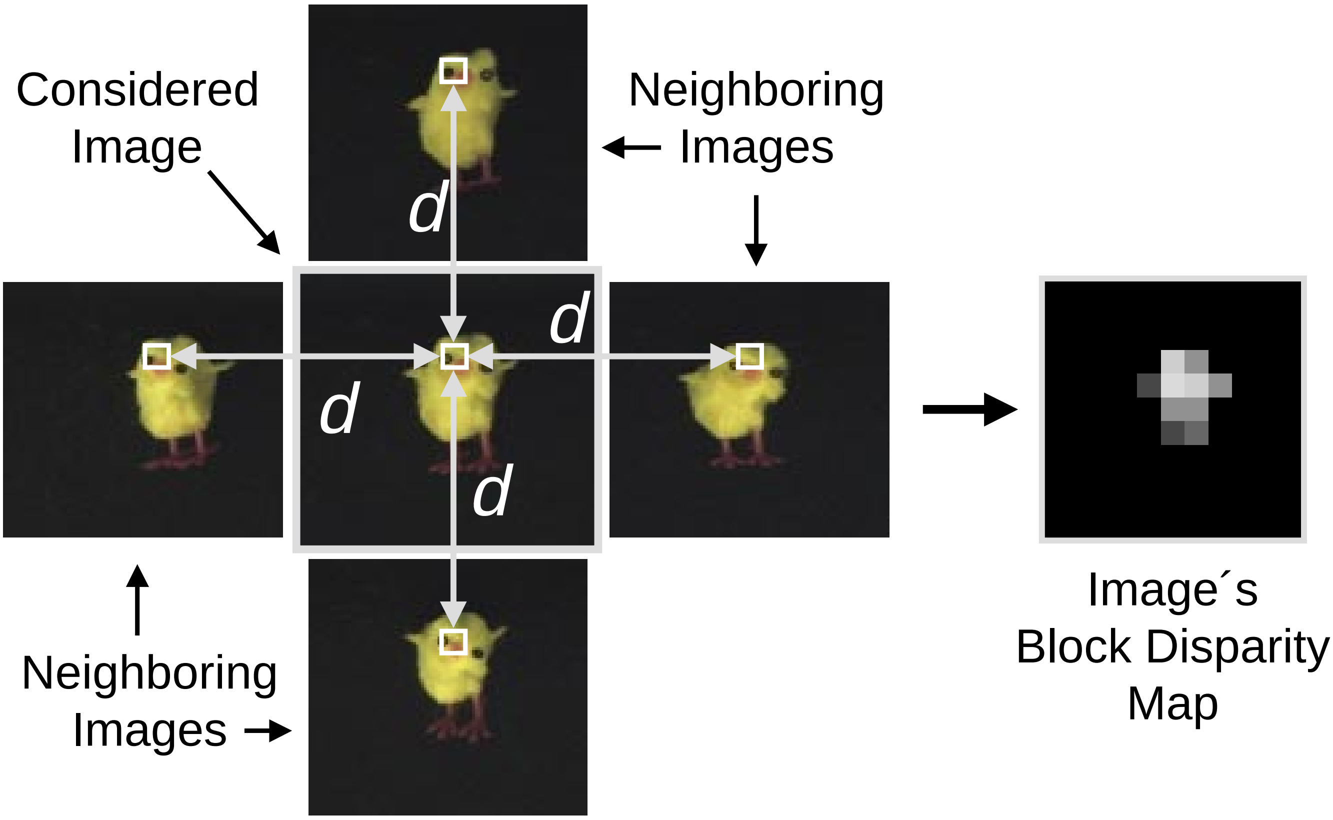

Fig. 2. A light-field image's disparity maps is estimated by comparing image blocks with neighboring images: for each block, different disparity values $d$ are tested, the corresponding blocks are extracted from the neighboring images, averaged, and the distortion (SSE) to the original image block is calculated.

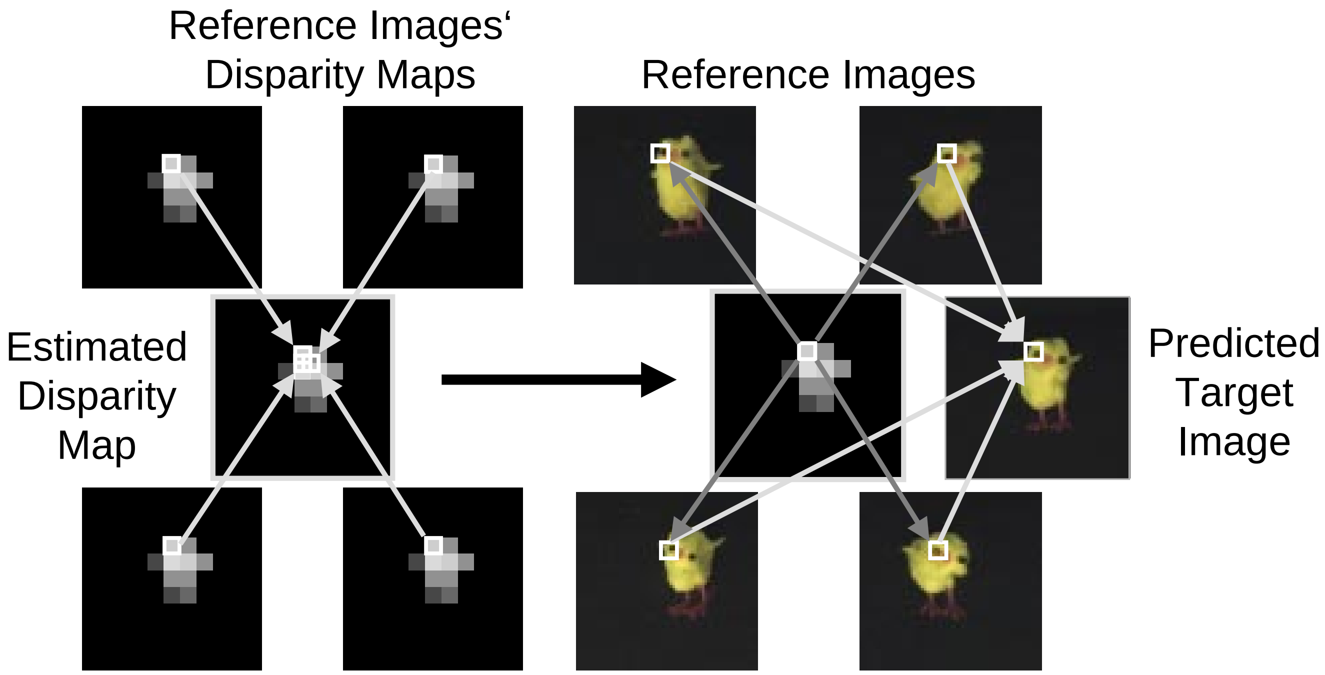

Fig. 3. Disparity compensation is performed by first estimating the target image's disparity map; The estimated disparity map is used to predict the target image from all reference images; the estimated images are averaged, yielding the target image's prediction.

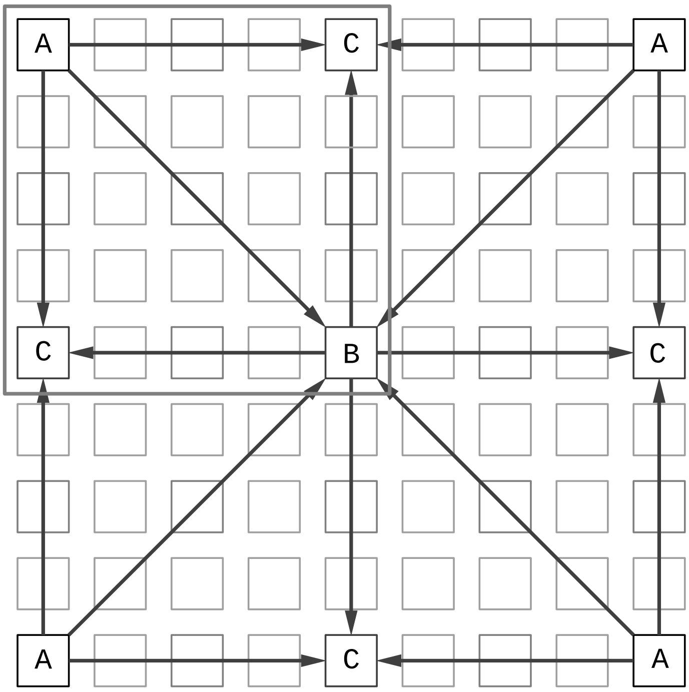

Fig. 4. Coding order of the D-coder: from the corner images (A), the center image (B) is predicted; the images at the middle of the sides (C) are predicted from the center image and the two closest corner images (A); the array is subdivided into quadrants and each quadrant is coded likewise; the algorithm keeps recursing until all images are coded.

Fig. 5. Images from the light fields *Dragon*, *Buddha* and *Chick*.

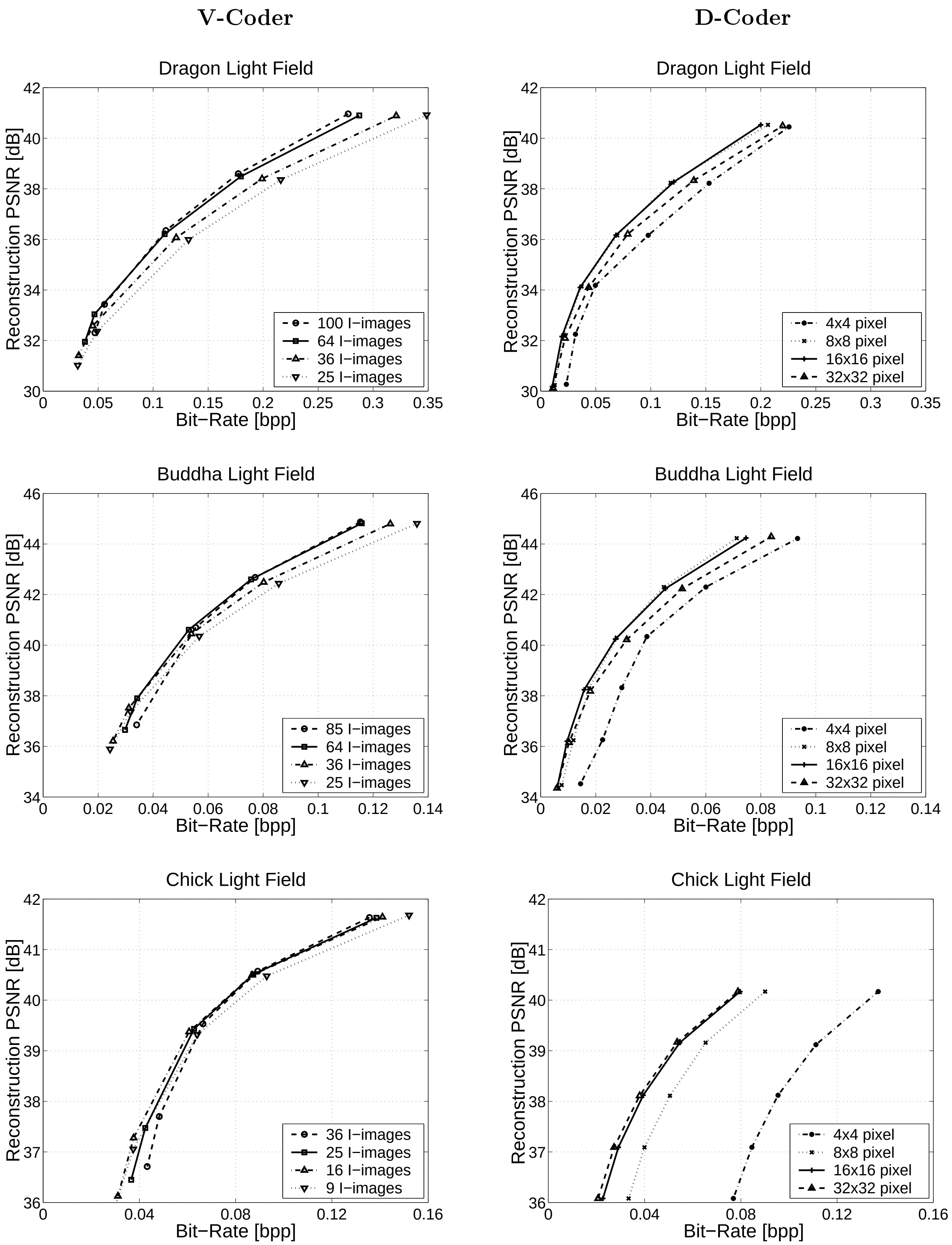

Fig. 6. R-D curves of the V-coder (left) and the D-coder (right), measured for 3 different light fields: the *Dragon* and *Buddha* light fields each consist of 1024 images, the *Chick* light field contains 289 images; for the V-coder, the number of intra-coded images (I-images) is varied, while different disparity map block sizes are considered for the D-coder's R-D curves.

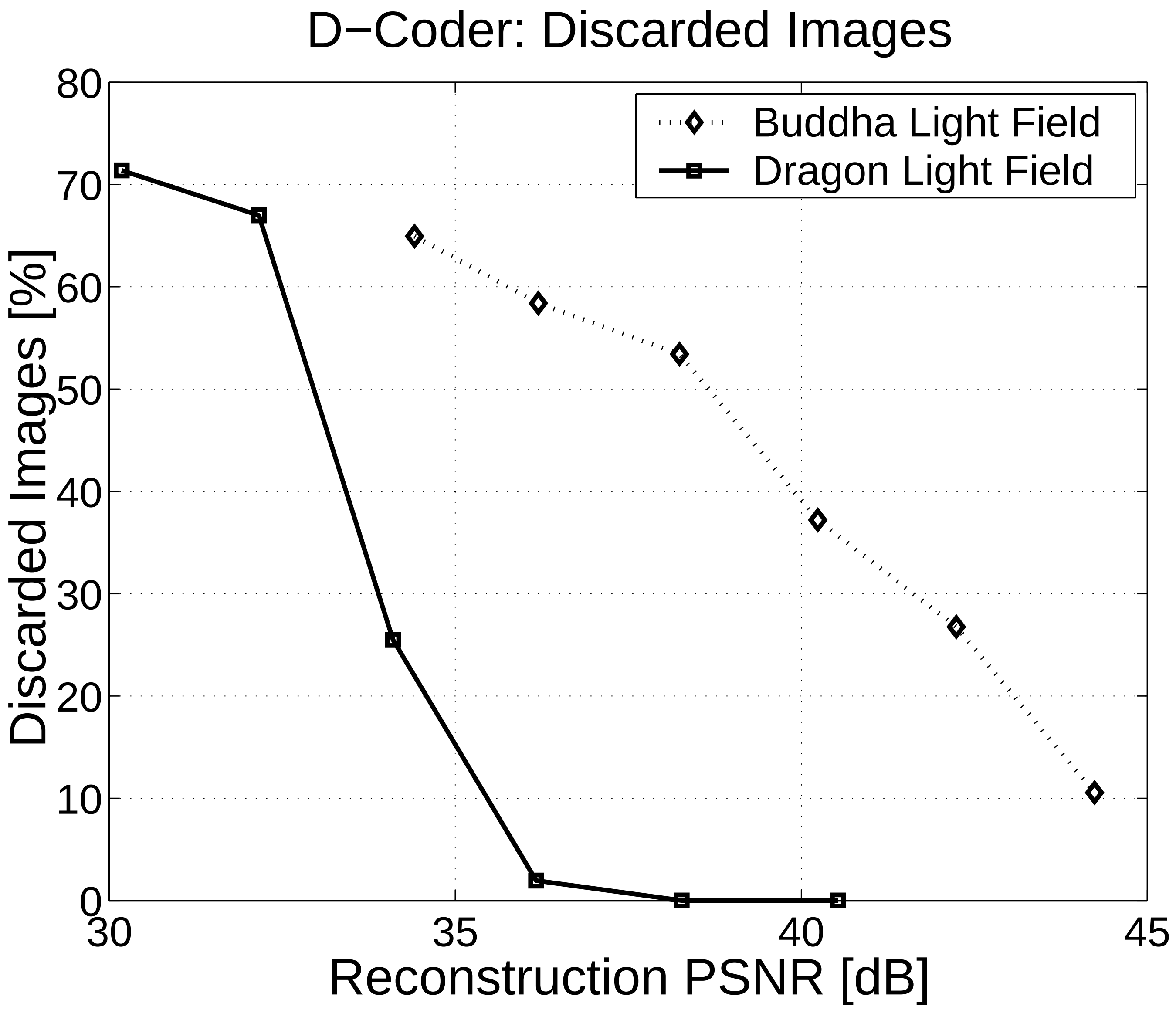

Fig. 7. D-coder: percentage of images from the *Buddha* and *Dragon* light fields that can be predicted well enough by disparity compensation alone (disparity map block size 16 × 16 pixels); no residual-error information needs to be coded for these images.