On Media Data Structures for Interactive Streaming in Immersive Applications

Gene Cheung[a], Antonio Ortega[b], Ngai-Man Cheung[c] and Bernd Girod[c]

[a]National Institute of Informatics, Tokyo, Japan;
[b]University of Southern California, Los Angeles, CA;
[c]Stanford University, Stanford, CA

ABSTRACT

Interactive media streaming is the communication paradigm where an observer, viewing transmitted subsets of media in real-time, periodically requests new desired subsets from the streaming sender, upon which the sender sends the appropriate media data corresponding to the received requests. This is in contrast to non-interactive media streaming like TV broadcast, where the entire media set is compressed and delivered to the observer before the observer interacts with the data (such as switching TV channels). Examples of interactive streaming abound in different media modalities: interactive browsing of JPEG2000 images, interactive light field or multiview video streaming, etc. Interactive media streaming has the obvious advantage of bandwidth efficiency: only the media subsets corresponding to observer's requests are transmitted. This is important when an observer only views a small subset out of a very large media data set during a typical streaming session. The technical challenge is how to structure media data such that good compression efficiency can be achieved using compression tools like differential coding, while providing sufficient flexibility for the observer to freely navigate the media data set in his/her desired order. In this introductory paper to the special session on "immersive interaction for networked multiview video systems", we overview different proposals in the literature that simultaneously achieve the conflicting objectives of compression efficiency and decoding flexibility.

Keywords: multiview video, video compression, interactive streaming

1. INTRODUCTION

An essential aspect of an immersive experience is the ability for an observer to interact naturally with a remote/virtual environment as if he is actually there. The observer may interact manually via a traditional keypad[1] or more naturally via a head-mounded tracking device,[2] but in either case an immersive communication system must in response produce quickly media data that corresponds to the observer's input; for example, if the observer tilts his head to the right, the view corresponding to the right-shifted view must be decoded and rendered for viewing in real-time. If the media data representing the environment already resides at the observer's terminal prior to media interaction, then the right subset of media corresponding to the observer's input can simply be fetched from memory, decoded and displayed. If the media data resides remotely in a server, however, then sending the entire data set over networks before an observer starts interacting with it can be prohibitively costly in bandwidth or delay; for example, the size of a set of light field data[3]—a densely sampled 2-D array of images taken by a large array of cameras[4] where a desired view is synthesized using image-based rendering (IBR)[5]—has been shown to be on the order of tens of Gigabytes,[6] while multiview video datasets have been captured using up to 100 time-synchronized cameras.[7]

Hence a more practical communication paradigm is one where the server continuously and reactively sends appropriate media data in response to an observer's periodic requests for data subsets—we call this *interactive media streaming* (IMS). This is in sharp contrast to *non-interactive media streaming* scenarios like terrestial digital TV broadcast,[8] where the entire media data set is delivered from server to client before a client interacts with the received data set (e.g., switching TV channels, superimposing picture-in-picture with two TV channels, etc). IMS has the potential of reduced bandwidth utilization since only the media subsets corresponding to the observer's requests are transmitted. However, efficiently coding the sequence of requested media subsets *prior* to the streaming session—in a stored-and-playback scenario—becomes a substantial technical challenge: while standard coding tools such as H.264[9] exploit correlation among neighboring frames using closed-loop motion

Table 1. Proposed Media Structures For IMS

Application	Structuring Technique
ROI image browsing	JPEG2000[10]
video browsing	JPEG2000+CR,[11] JPEG2000+MC[12]
light field streaming	rerouting,[13,14] SP[15] intra DSC,[16] DSC+MC+coset[17]
ROI video streaming	multi-res MC[18,19]
reversible video playback	DSC+MC[20]
multiview streaming	DSC+MC[21] rerouting,[22] redundant P-frames[22,23]

compensation for coding gain, the obvious correlation that exists among requested media subsets is difficult to exploit since at coding time, the order and selection of media subsets chosen by the observer during streaming time in the future is unknown. This is the *inherent tension between media interactivity and coding efficiency*; i.e., providing maximum "navigation" flexibility can come at the cost of lower coding efficiency.

Over the past few years, as shown in Table 1, researchers have devised novel coding structures and techniques to achieve different tradeoffs between media interactivity and coding efficiency. In this introductory paper for the special session on "immersive interaction for networked multiview video systems", we provide a detailed overview of various proposals in the literature. A key contribution of this work is to provide, for the first time, taxonomies of these methods and how they relate to each other. The outline of the paper is as follows. We first overview different proposed encoded data structures for different media modalities in Section 2. We then present two taxonomies on how different proposals are related in Section 3. We then narrow our focus to a single application—interactive multiview video streaming—and how how quantitatively different quantities can be traded off in Section 4. Finally, concluding remarks are presented in Section 5.

2. OVERVIEW OF STRUCTURES & TECHNIQUES FOR IMS

We first overview various proposed structures and techniques in the literature for IMS. IMS has been used for a wide range of media modalities and applications; see Table 1 for a list of applications and corresponding techniques to support the applications.

2.1 ROI Image Browsing

For Region-Of-Interest (ROI) image browsing, where an observer can interactively select a region of any location and scale in a possibly very large image (e.g., a geographical map), Taubman et al[10] has proposed the use of JPEG2000 image coding standard for its fine-grained spatial, resolution and quality scalability.[24] In details, given the set of subband coefficients of a discrete wavelet transform already residing at the observer's cache, the sender sends only the missing coefficients corresponding to the requested spatial location and scale of the requested ROI, in incrementally rate-distortion optimal quality layers called *packets*[10] to the observer. The image can be displayed continuously at the observer as more packets are received with gradually improving quality.

2.2 Video Browsing

Exploiting scalable nature of JPEG2000, proposals[11,12] have also been made to use JPEG2000 for video browsing, where the streaming video can be randomly accessed with complete flexibility at the frame level: random access any frame in sequence, forward/backward playback in time, playback at $K\times$ speed by decoding only every K frames, etc. Though at encoding time each frame is encoding independently, inter-frame redundancy can nevertheless be exploited. Devaux et al[11] proposed to use conditional replenishment (CR), where the coefficients of a code-block of a desired frame were sent or replenished only if the corresponding code-block of the previous frame already at the decoder did not provide a good enough approximation. Along similar line, assuming a motion model Taubman et al[12] performed motion compensation (MC) at the server, so that the transmitted motion information could be used in combination of code-blocks of previous frames to approximately reconstruct code-blocks of requested frames. The server would send new code-blocks only if the motion-compensated approximation was not good enough.

2.3 Reversible Video Playback

Another example is reversible video playback:[20] a video frame was encoded using distributed source coding (DSC), where both past and future frames were used as side information (predictors). This was done so that frames could be sent either forward or backward in time per client's request, and the client could simply decode and play back the video in the transmission order with no excess buffering. Note that this was a more limited form of media interactivity than video browsing, but unlike JPEG2000-based approaches[11,12] for video browsing, where each frame was encoded independently, the inter-frame redundancy was explicitly exploited here during actual media encoding, hence the resulting encoding rate is expected to be much lower.

2.4 Interactive Light Field Streaming

In the case of *light fields*,[3] where a subset of a densely sampled 2-D array of images is used to interpolate a desired view using image-based rendering (IBR),[5] the notion of interactive media streaming has been investigated extensively.[13–17] These works were motivated by the very large size of the original image set,[6] which will cause intolerable delay if the set must be transmitted in its entirety before user's interaction begins.

To provide random access to images in the set, Aaron et al[16] coded images independently. Aaron et al[16] (intra DSC) encoded non-key images (key images are encoded as I-frames) independently using a Wyner-Ziv encoder, and transmitted different amount of the encoded bits depending on the quality of the side information (frames already transmitted) available at the decoder. To exploit the large spatial correlation inherent among densely sampled images, however, the majority of these works[13–15,17] encoded the image set using *disparity compensation*, i.e., differential coding using neighboring view images as predictors.

For a given image, Jagmohan et al[17] and Ramanathan et al[15] used DSC and SP-frame-like lossless coding respectively to eliminate frame differences caused by usage of different predictors of different decoding paths. Specifically, Jagmohan et al[17] proposed to encode disparity information for every possible neighboring view image, *plus* DSC based coset bits so that when both are applied, the resulting image is the same no matter which neighboring image was used as predictor. Instead of DSC, Ramanathan et al[15] used a lossless coding scheme akin to SP-frames in H.264[25] to encode residues after disparity compensation so that the exact same frame was reconstructed no matter which predictor was used.

Bauermann et al[13,14] took a different approach and analyzed tradeoffs among four quantities—storage rate, distortion, transmission data rate and decoding complexity. In particular, they first assumed that each coding block of an image was encoded as INTRA, INTER or SKIP as done in H.263.[26] Then, for a requested INTER coding block to be correctly decoded, all blocks in its dependency chain* that were *not* already in the client cache must be transmitted, creating a cost both in transmission rate and decoding complexity. We denote this technique as *rerouting*, as the dependency path of blocks from desired inter block all the way back to the initial intra block must be re-traced and transmitted if not residing in the observer's cache.

2.5 ROI Video Streaming

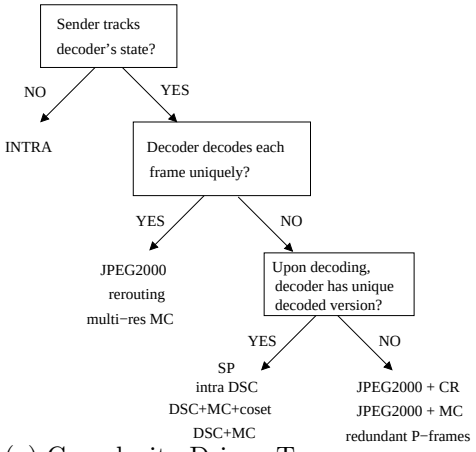
A recent application called ROI video streaming involves the transmission of high-resolution video to an observer with low-resolution display terminal. In such scenario, the observer can choose between viewing the entire spatial region but in low resolution, or viewing a selected smaller ROI but in higher resolution. Mavlankar et al[18,19] proposed a multi-resolution motion compensation (multi-res MC) technique where a low resolution version of the video called *thumbnail* was first encoded, then frames at higher resolution were each divided into different tiled spatial regions, which were motion-compensated using an up-sampled version of the thumbnail as predictor. Prediction for the high-resolution frames used only the thumbnail, so that ROIs could be arbitrarily chosen across time by observers without causing complicated inter-frame dependencies among high-resolution frames. The procedure can be repeated for even smaller spatial regions and higher resolution.

*By dependency chain we mean all the blocks that need to be decoded before the requested block can be decoded, i.e., an INTRA block followed by a succession of disparity compensated INTER blocks.

2.6 Interactive Multiview Video Streaming

While much of multiview video coding[27–29] focuses on the rate-distortion performance of compressing all frames of all views for storage or non-interactive video delivery over networks, in our previous works[21–23] we have addressed the problem of designing a frame structure to enable interactive multiview video streaming, where clients can interactively switch views during video playback in time. Thus, as a client is playing back successive frames (in time) for a given view, it can send a request to the server to switch to a different view while continuing uninterrupted temporal playback. To provide view switching capability for the observer while maintaining good compression efficiency, we have developed redundant P-frame representation[22] where multiple P-frames are encoded for the same original picture and stored at the encoder, each using as a predictor a different previous frame that was on a possible observer's navigation trajectory. Multiple representation nature of redundant P-frames means they lower transmission rate at the expense of more storage of media data. But an in-discriminatory use of redundant P-frame representation will lead to exponential expenditure in storage; to avoid such problem elegantly without resorting to bandwidth-expensive I-frame, we developed novel DSC implementations[21] to merge switches from multiple decoding paths into a single frame representation. Our most recent work[23] discussed preliminary results of using I-, P- and DSC frames in an optimized structure for interactive multiview video streaming.

3. TAXONOMY OF STRUCTURES & TECHNIQUES FOR IMS



(a) Complexity-Driven Taxonomy

	unique decoding	multiple decoding
unique encoding	rerouting[13,14,22] JPEG2000[10] intra DSC[16] multi-res MC[18,19]	JPEG2000+CR[11] JPEG2000+MC[12]
multiple encoding	SP[15] DSC+MC+coset[17] DSC+MC[21]	redundant P-frames[22]

(b) Data-Driven Taxonomy

Figure 1. Two Taxonomies categorizing Structures & Techniques for IMS

After reviewing proposed techniques for different IMS applications in the literature, in this section we discuss two taxonomies to classify previously proposed media structuring techniques that achieve different tradeoffs between interactivity and compression efficiency into logical categories. See Fig. 1 for an illustration.

3.1 Complexity-driven Taxonomy

The first taxonomy is complexity-driven and can be described as follows. First, media data can be structured such that when the sender receives a request from the observer corresponding to a specific media subset, the sender does not need to know in what state the decoder is in (i.e., what media data has already been transmitted and decoded at the observer). This is the *INTRA* structure: an encoding scheme codes each frame independently, and during the streaming session the sender simply sends the media subset that directly corresponds to the observer's request. This requires the least computation of all structure techniques. Though simplistic, when there is little correlation to exploit for coding gain between requested media subsets, this is a perfectly viable approach.

Alternatively, the sender can keep track of the decoder's state (what media data has been transmitted) and sends different subsets of media data according to both the observer's request and the decoder's state. The simple sub-category here is the class of techniques where the media is encoded such that there is only one

unique way of decoding the requested media data subset. In this case, the sender simply sends the subset of the requested media that are missing from the receiver's cache, so that the decoder can perform the required unique decoding. JPEG2000-based scheme[10] for ROI image browsing, rerouting[13,14] for light field streaming, and multi-res MC[18,19] for ROI video streaming are examples of this category.

In the next class of structures, each structure has more than one unique way of decoding each requested media subset, but there is only one unique decoded version. DSC for reversible video playback,[20] SP-frames for light field streaming,[15] DSC-based techniques for light field streaming[16,17] and DSC-based techniques for interactive multiview video streaming[21] fall in this class. This requires more computation than the previous class, since multiple ways to compute a given requested media subset means, first, an increase in pre-computation to derive the multiple encodings, and second, a more complex mapping between decoder states and stored media representations.

Finally, the last class of structures result in multiple decoded versions for a given observer request. This is the most complex class in that the multiple decoded versions lead to multiplicative increase in the number of decoder states, which the server must keep track of. JPEG2000+CR[11] and JPEG2000+MC[12] for video browsing, and Redundant P-frames[22,23] belong to this class. Typically, some methods to reduce the number of possible decoder states such as novel DSC implementations[21] must be used before the exponential growth of decoder states becomes intractable.

3.2 Data-driven Taxonomy

Another way of categorizing the proposed structuring techniques in the literature is from a data-driven perspective. First, at the encoder a particular media subset can be encoded in one unique version, or multiple versions, using different neighboring frames as predictors, for example. At the decoder, at the end of the decoding process, for a given request one can produce one or multiple decoded versions. Fig. 1(b) shows how the proposed techniques fall into different categories.

While unique encoding / unique decoding and multiple encoding / multiple decoding are conceptually straightforward, unique encoding / multiple decoding and multiple encoding / unique decoding are more subtle. For unique encoding / multiple decoding, JPEG2000+CR[11] and JPEG2000+MC[12] both encode images independently and uniquely using JPEG2000 during encoding, then during actually streaming, generate motion information for given previous frame (predictor), so that different decoded versions are resulted for different previous frames at the decoder buffer. For multiple encoding / unique decoding, different motion information are pre-encoded for different previous frames. The nature of SP and DSC frames ensures that the same unique decoded version can be recovered despite the difference in predictors.

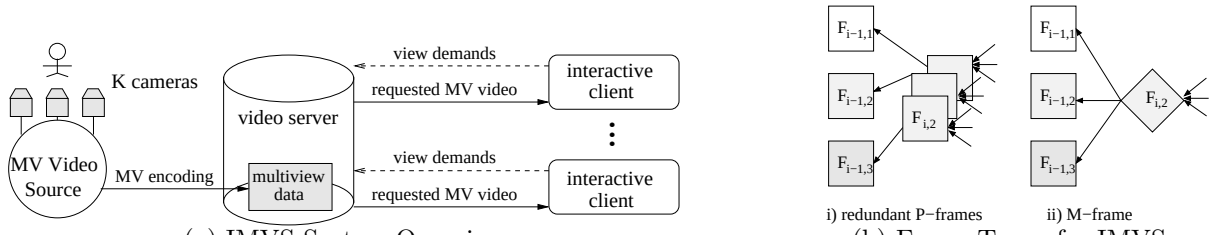
4. EXAMPLE IMS APPLICATION: INTERACTIVE MULTIVIEW VIDEO STREAMING

For a given IMS application, there are tradeoffs among several quantities that must be optimized in an application-specific manner: transmission rate of the actual interactive streaming session, storage required for pre-encoded media data, encoding and decoding complexity, level of interaction supported, etc. In this section, we focus on a specific application—interactive multiview video streaming (IMVS)[21–23]—and study how these tradeoffs can be optimized.

4.1 Overview of IMVS and IMVS-specific Coding Tools

As briefly described earlier, IMVS is an application where a multiview video sequence is first pre-encoded and stored at the server in a redundant representation, so that during a subsequent streaming session, an interactive client can periodically request view switches at pre-defined period of M frames as video is being streamed and played back uninterrupted. Fig. 2(a) shows an overview of an IMVS system. Note that an alternative approach of real-time encoding a decoding path tailored for each client's unique view traversal across time is computationally prohibitive as the number of interactive clients increase.

The challenge in IMVS is to design a frame structure for multiview video data so that streaming bandwidth is optimally traded off with storage required to store the multiview video data, to support a desired level of IMVS



(a) IMVS System Overview (b) Frame Types for IMVS

Figure 2. IMVS System Overview and Example Frame Types

interaction. By IMVS interaction, we mean both the view switching period M (small M leads to faster view switches and more interactivity), and likelihood that a user is to switch view given M. We define α to be the probability that a user would choose to switch to a neighboring view at a view switching point; large α means a user is more likely to switch views, leading to more interactivity.

As building blocks to build an IMVS frame structure, we have previously proposed to use combinations of redundant P-frames[22] and DSC-based *merge*-frames (M-frames).[21] Examples of these tools are shown in Fig. 2(b). Redundant P-frames encode one P-frame for each frame just prior to a view switch, using the previous frame as a predictor for differential coding, resulting in multiple frame representations for a given original picture. In Fig. 2(b)(i), we see that there are three P-frames $F_{i,2}$'s representing the same original picture $F^o_{1,2}$ of time instant 1 and view 2, each using a different frame in previous time instant—one of $F_{i-1,1}$, $F_{i-1,2}$ and $F_{i-1,3}$—as predictor. While redundant P-frames result in low transmission bandwidth, it is obvious that using it alone would lead to exponential growth in storage as the number of view switches across time increase.

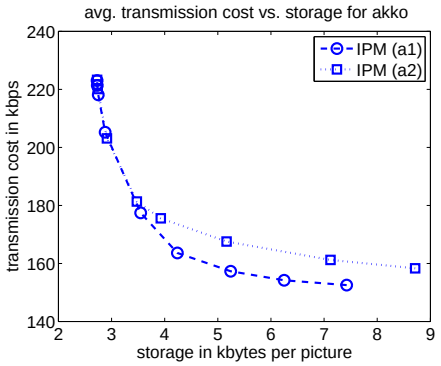
As an alternative coding tool, we have proposed an M-frame, where a single version $F_{i,j}$ of the original picture $F^o_{i,j}$ can be decoded no matter from which frame a user is switching from. In Fig. 2(b)(ii), the same $F_{i,2}$ can be decoded no matter which one of $F_{i-1,1}$ $F_{i-1,2}$ and $F_{i-1,3}$ a user is switching from. One straightforward implementation of M-frame is actually an I-frame. We have previously shown, however, that DSC-based coding tools exploiting the correlations between previous frames $F_{i-1,k}$'s and target frame $F_{i,j}$ can result in implementations that outperform I-frame in both storage and transmission rate. An M-frame, however, remains a fair amount larger than a corresponding P-frame.

It should be obvious that a frame structure composed of large portions of redundant P-frames relative to M-frames will result in low transmission rate but large storage, while a structure composed of small portions of redundant P-frames relative to M-frames will result in higher transmission rate but smaller storage. We refer readers to Cheung et al[23] for details on how an optimal structure is found for a desire transmission-storage tradeoff using combination of redundant P-frames and M-frames, for given desired IMVS interaction. We focus instead on observing how the noted tradeoffs were manifested quantitatively in the IMVS context.

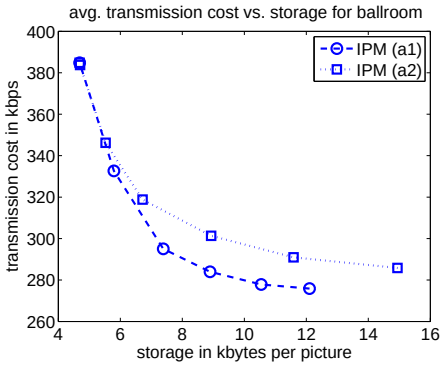
4.2 IMVS Tradeoffs

We now show quantitatively tradeoffs among transmission rate, storage and interactivity. In Fig. 3(a) and (b), we see the performance of our optimized structures as different tradeoff points between transmission and storage, for two test sequences `akko&kayo` and `ballroom`. They are coded at 320×240 resolution at 30 and 25 frames per second, respectively. Quantization parameters (QP) were selected so that visual quality in peak Signal-to-noise ratio (PSNR) was about 34dB. View switching period M was fixed at 3. For each sequence, two trials were performed where the view switching probability α was set at 0.1 and 0.2.

For both sequences, we see that at low storage, the difference between the two curves corresponding to the two view switching probabilities is quite small. This agrees with our intuition; at low storage, large portions of the frame structures are M-frames, which by design induce the same transmission rate no matter from which decoding path it is switching from. At high stroage, however, we see the difference between the two curves becomes larger. As more redundant P-frames are used, P-frames can selectively be deployed for decoding paths with high probabilities, leading to large reduction in transmission cost per P-frame encoded.

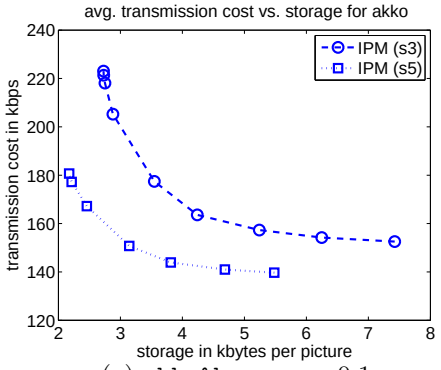


(a) `akko&kayo`, $M = 3$

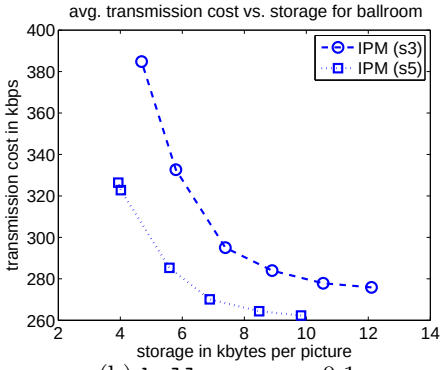


(b) `ballroom`, $M = 3$

Figure 3. Tradeoff between Transmission and Storage for different View Switching Probabilities



(a) `akko&kayo`, $\alpha = 0.1$



(b) `ballroom`, $\alpha = 0.1$

Figure 4. Tradeoff between Transmission and Storage for different View Switching Periods

In Fig. 4(a) and (b), we see the transmission-storage tradeoff of the optimized IMVS structures for the same two test sequences when the view switching period M changed from 3 to 5. Again, we see similar trend for the two sequences: performance curve for larger view switching period resides in a lower convex hull. This is intuitive as well; large view switching period M means $M - 1$ P-frames of the same view can be used between switches, leading to lower transmission rate.

5. CONCLUSION

In this introductory paper, we first argue for the advantage of interactive media streaming over non-interactive media streaming, then overview existing coding techniques for different modalities to support interactive media streaming. Each technique offers its own unique tradeoffs among different quantities: computation, transmission rate, storage, and interactivity. One of our key contributions is to present taxonomies to categorize existing techniques in a logical fashion. We also narrow our focus to the interactive multiview video streaming application, and present quantitative results that illustrated the said tradeoffs.

REFERENCES

[1] Lou, J.-G., Cai, H., and Li, J., "A real-time interactive multi-view video system," in [*ACM International Conference on Multimedia*], (November 2005).

[2] Kurutepe, E., Civanlar, M. R., and Tekalp, A. M., "Client-driven selective streaming of multiview video for interactive 3DTV," in [*IEEE Transactions on Circuits and Systems for Video Technology*], **17, no.11**, 1558–1565 (November 2007).

[3] Levoy, M. and Hanrahan, P., "Light field rendering," in [*Proc. SIGGRAPH'96*], 31–42 (August 1996).

[4] Wilburn, B., Smulski, M., Lee, H., and Horowitz, M., "The light field video camera," in [*SPIE Electronic Imaging: Media Processors'2002*], **4674** (December 2002).

[5] Shum, H.-Y., Kang, S. B., and Chan, S.-C., "Survey of image-based representations and compression techniques," in [*IEEE Transactions on Circuits and Systems for Video Technology*], **13, no.11**, 1020–1037 (November 2003).
[6] Levoy, M. and Pulli, K., "The digital michelangelo project: 3-D scanning of large statues," in [*Proc. SIGGRAPH'00*], 131–144 (August 2000).
[7] Fujii, T., Mori, K., Takeda, K., Mase, K., Tanimoto, M., and Suenaga, Y., "Multipoint measuring system for video and sound—100 camera and microphone system," in [*IEEE International Conference on Multimedia and Expo*], (July 2006).
[8] "Digital video broadcasting." http://www.dvb.org/.
[9] Wiegand, T., Sullivan, G., Bjontegaard, G., and Luthra, A., "Overview of the H.264/AVC video coding standard," in [*IEEE Transactions on Circuits and Systems for Video Technology*], **13, no.7**, 560–576 (July 2003).
[10] Taubman, D. and Rosenbaum, R., "Rate-distortion optimized interactive browsing of JPEG2000 images," in [*IEEE International Conference on Image Processing*], (September 2003).
[11] Devaux, F.-O., Meessen, J., Parisot, C., Delaigle, J., Macq, B., and Vleeschouwer, C. D., "A flexible video transmission system based on JPEG2000 conditional replenishment with multiple references," in [*IEEE International Conference on Acoustics, Speech, and Signal Processing*], (April 2007).
[12] Naman, A. T. and Taubman, D., "A novel paradigm for optimized scalable video transmission based on JPEG2000 with motion," in [*IEEE International Conference on Image Processing*], (September 2007).
[13] Bauermann, I. and Steinbach, E., "RDTC optimized compression of image-based scene representation (part I): Modeling and theoretical analysis," in [*IEEE Transactions on Image Processing*], **17, no.5**, 709–723 (May 2008).
[14] Bauermann, I. and Steinbach, E., "RDTC optimized compression of image-based scene representation (part II): Practical coding," in [*IEEE Transactions on Image Processing*], **17, no.5**, 724–736 (May 2008).
[15] Ramanathan, P. and Girod, B., "Random access for compressed light fields using multiple representations," in [*IEEE International Workshop on Multimedia Signal Processing*], (September 2004).
[16] Aaron, A., Ramanathan, P., and Girod, B., "Wyner-Ziv coding of light fields for random access," in [*IEEE International Workshop on Multimedia Signal Processing*], (September 2004).
[17] Jagmohan, A., Sehgal, A., and Ahuja, N., "Compression of lightfield rendered images using coset codes," in [*Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*], **1**, 830–834 (November 2003).
[18] Mavlankar, A., Baccichet, P., Varodayan, D., and Girod, B., "Optimal slice size for streaming regions of high resolution video with virtual pan/tilt/zoom functionality," in [*Proc. European Signal Processing Conference (EUSIPCO-07)*], (Sept. 2007).
[19] Mavlankar, A. and Girod, B., "Background extraction and long-term memory motion-compensated prediction for spatial-random-access enabled video coding," in [*Proc. International Picture Coding Symposium (PCS)*], (May 2009).
[20] Cheung, N.-M., Wang, H., and Ortega, A., "Video compression with flexible playback order based on distributed source coding," in [*IS&T/SPIE Visual Communications and Image Processing (VCIP'06)*], (January 2006).
[21] Cheung, N.-M., Ortega, A., and Cheung, G., "Distributed source coding techniques for interactive multiview video streaming," in [*27th Picture Coding Symposium*], (May 2009).
[22] Cheung, G., Ortega, A., and Cheung, N.-M., "Generation of redundant coding structure for interactive multiview streaming," in [*Seventeenth International Packet Video Workshop*], (May 2009).
[23] Cheung, G., Cheung, N.-M., and Ortega, A., "Optimized frame structure using distributed source coding for interactive multiview streaming," in [*IEEE International Conference on Image Processing*], (November 2009).
[24] "JPEG2000 interactive protocol (part 9—JPIP)." http://www.jpeg.org/jpeg2000/j2kpart9.html.
[25] Karczewicz, M. and Kurceren, R., "The SP- and SI-frames design for H.264/AVC," in [*IEEE Transactions on Circuits and Systems for Video Technology*], **13, no.7**, 637–644 (July 2003).
[26] ITU-T Recommendation H.263, *Video Coding for Low Bitrate Communication* (February 1998).
[27] Merkle, P., Smolic, A., Muller, K., and Wiegand, T., "Efficient prediction structures for multiview video coding," in [*IEEE Transactions on Circuits and Systems for Video Technology*], **17, no.11**, 1461–1473 (November 2007).
[28] Flierl, M., Mavlankar, A., and Girod, B., "Motion and disparity compensated coding for multiview video," in [*IEEE Transactions on Circuits and Systems for Video Technology*], **17, no.11**, 1474–1484 (November 2007).
[29] Shimizu, S., Kitahara, M., Kimata, H., Kamikura, K., and Yashima, Y., "View scalable multiview coding using 3-D warping with depth map," in [*IEEE Transactions on Circuits and Systems for Video Technology*], **17, no.11**, 1485–1495 (November 2007).