Multi-Viewpoint Video Coding with MPEG-2 Compatibility

Belle L. Tseng and Dimitris Anastassiou

Columbia University New York, N.Y. 10027 USA

Abstract

An efficient video coding scheme is presented as an extension of the MPEG-2 standard to accommodate the transmission of multiple viewpoint sequences on bandwidth-limited channels. With the goal of compression and speed, the proposed approach incorporates a variety of existing computer graphics tools and techniques. Construction of each viewpoint image is predicted using a combination of perspective projection of 3D models, texture mapping, and digital image warping. Immediate application of the coding specification is foreseeable in systems with hardware-based real-time rendering capabilities, thus providing fast and accurate constructions of multiple perspectives.

1 Introduction

Recent interests in 3D technologies prompt the addition of depth impression onto the otherwise common 2D video signals. Two major processes to perceiving 3D can be categorized. The first is due to the two slightly different perspectives of the world offered simultaneously to our left and right eyes. The human visual system then converts these two stereo images into one single fused 3D perception. The other approach to sense depth, even with only one eye-viewpoint, is through motion parallax. Due to motion from our head movements, the relative object displacements of the resulting perspective view are sufficient cues in deriving the 3D sensation. Accordingly, in our presentation, the depth appearance is contributed by both processes.

A multi-viewpoint video, multiview for short, is a 3D extension of the traditional movie sequence, in that there are multiple perspectives of the same scene at any one instance in time. Comparable to a movie made by a sequence of holograms, a multiview video offers a similar look-around capability. An ideal multiview system allows any user to watch a true 3D stereoscopic sequence from any perspective the viewer chooses. Such a system has practical uses in interactive applications, medical technologies, educational and training demonstrations, remote sensing developments, and is a step towards virtual reality.

With the development of digital video technology, a video data compression standard, namely the second Motion Picture Experts Group specification (MPEG-2), has been adopted by the International Standards Organization (ISO) and the International Telecommunications Union (ITU). MPEG-2 specifies the coding process for one video sequence; detailed descriptions can be found in [1]. Recently, MPEG-2 has also been shown to be applicable to two sequences of stereoscopic signals through the use of spatial and temporal scalability extensions[2, 3, 4, 5]. However, extending the number of video viewpoints beyond two cannot be done practically by using the same methodology. For this motivation, a novel multiview codec is presented to complement the MPEG-2 standard.

2 Geometric Definitions and Notations

An ordinary 2D video sequence offers only one perspective of the acquired scene. For a 3D viewing experience, at least 2 viewpoints are required to obtain the depth impression from the left and right perspectives of a stereoscopic signal. In a multiview system, multiple viewpoints are involved. Let N be the number of viewpoint sequences, where the minimum number of viewpoints is equal to 3 for the presented specification, i.e., $N \geq 3$. The number of viewpoints must be extendable in the future, thus accommodation of additional viewpoints is an essential feature.

One central viewpoint image is designated to be the principal view in which the other multiview images are predicted from. The central image is denoted by I_C from viewpoint V_C. The central viewpoint, usually the middle viewpoint, is chosen so that its image has the highest collection of overlapping objects with each of the other viewpoint images. In this manner, the central viewpoint image can be used to interpolate and predict most of the other views.

The other multiview images are designated by I_X from viewpoints V_X. An example of four additional views is illustrated in Figure 1, where cameras positioned at viewpoints $V_X = \{V_L, V_R, V_T, V_B\}$ capture respective images $I_X = \{I_L, I_R, I_T, I_B\}$ corresponding to the Left, Right, Top, and Bottom. These camera-captured images available at the encoder are referred to as *real* views, whereas images not directly taken by a camera, but derived by prediction or interpolation methods, are called *virtual* views. Virtual pictures include those viewpoint images seen between two real cameras, thus having such virtual constructions allow the viewer to see a smoother video transition between two real views.

The image coordinate system corresponding to each viewpoint is defined as (X_i, Y_i), where viewpoint index $i = \{C, L, R, T, B\}$. Let the global rectangular coordinate system (X, Y, Z) be

defined as corresponding to the image coordinate system (X_C, Y_C) of the central viewpoint, where Z defines the orthogonal axis from the central image plane.

The camera position is represented by the global coordinates (vx_i, vy_i, vz_i), and the camera zooming parameter is described by va_i. In addition, the camera rotations are given by the horizontal panning angle vb_i and the vertical tilting angle vc_i. The total viewpoint vector for some viewpoint index i is denoted as $V_i = [vx_i, vy_i, vz_i, va_i, vb_i, vc_i]$. The other viewpoint vectors are obtained relative to the central viewpoint, $V_C = [0, 0, 0, 1, 0, 0]$. Given the acquisition camera configurations, the other viewpoint vectors are relative translations and rotations with respect to the central viewpoint. If camera configurations are unknown, then locating a couple of fixed points between multiviews allows determination of the relative orientation transformation[6, 7].

3 Depth and Disparity Estimation

Possessing the depth of an object in one image allows for geometric prediction of the object location in all other viewpoint images. Consequently, our desire to determine the depth of every object in a scene permits construction of the scene from any viewpoint. The depth of an object can be geometrically calculated if two or more perspectives of the object are given, as in a collection of multiviews. First, the positions of the object in each of the available viewpoint images must be located; this problem is widely known as the *correspondence problem.* After locating the object positions from two views, the difference in image coordinates is termed *disparity*. Following, it can be shown mathematically that the depth is inversely proportional to the derived disparity[7].

Under the block-based motion characterization constraint of MPEG-2, one disparity vector corresponding to each block of an image can be incorporated for prediction of a second image. Block-based disparity compensation is the dominant approach for the coding of stereoscopic video sequences. These approaches are sufficient for coding and transmission of two stereo signals without entailing true depth details to the decoding system. For acceptable predictions of multi-viewpoint images however, accurate depth information for every pixel is required for multiple spatially-continuous interpolations of intermediate viewpoint images. Thus a depth map is required covering every pixel, whose quantized values can be transmitted as the gray-level intensity of a second "image". Since the majority of the depth image is quite flat, the depth map can be considerably compressed. To obtain a dense depth map, many efficient techniques have been developed for stereo image pairs as well as for multiple views, including [7, 8, 9, 10].

4 MPEG-2 Compatible Multi-Viewpoint Video Coding

The delivery of two viewpoint video sequences is accomplished by the scalability extensions of the MPEG-2 video coding standard, where spatial predictions are obtained by disparity-compensation on a macroblock basis. As mentioned in Section 3 however, for multiview video coding a dense depth map is required for accurate predictions of multiple views. Seeking compatibility with the compression standard, an ad hoc group of MPEG-2 is established to investigate a new profile for multi-viewpoint systems[11]. For conformity, the multiview profile is to supplement the main profile of MPEG-2 so that all standardized systems are capable of deciphering at least one video sequence from one viewpoint. Consequently, the central viewpoint is selected to be processed and transmitted in accordance with the main profile. The remainder of this section is devoted to describe the proposed encoding and decoding systems for the processing of the other viewpoints.

4.1 Multi-Viewpoint Encoding System

A block diagram of the proposed multiview encoding system is illustrated in Figure 2 for five multi-viewpoint sequences: $I_C^t, I_L^t, I_R^t, I_T^t$, and I_B^t. First, determine all viewpoint vectors, $V_C^t, V_L^t, V_R^t, V_T^t$, and V_B^t, from the acquisition camera configurations positioned in Figure 1. Find the central viewpoint image, in our case I_C, achieving the best predictions for the other views. The central image sequence I_C^t is then processed in the main profile of MPEG-2. Following, the encoded bitstream for I_C^t is ready for transmission, and in parallel is decoded to determine the receiver-reconstructed central images, denoted $\widehat{I_C^t}$.

Starting with the central viewpoint image I_C^t at time t, a depth value $z_C = D_C^t(x_C, y_C)$ is calculated for every pixel $I_C^t(x_C, y_C)$, thus forming a depth map image, named D_C^t, corresponding to the central viewpoint image. Afterwards, encode the depth map D_C^t as a secondary highly compressible image, and transmit the encoded bitstream for D_C^t. Simultaneously, the encoded bitstream for D_C^t is decoded to determine the received-reconstructed depth map, assigned $\widehat{D_C^t}$.

Obtain a 3D mesh model representation[12], called M^t, for the central image $\widehat{I_C^t}$ of time t by associating each coordinate pair (x_C, y_C) with its corresponding depth value $\widehat{D_C^t}(x_C, y_C)$. Consequently, the 3D surface texture is derived from the 2D image intensity $I_C^t(x_C, y_C)$, and a graphical model of the scene is obtained. This geometric representation offers ease in interpolating different viewpoints. Similar to rendering approaches in computer graphics[13, 14], given a 3D geometrical model and its associated texture intensity, every viewpoint can be effectively and efficiently

constructed by simple geometric transformations followed by 3D texture mapping. Furthermore, interpolating virtual viewpoint images is facilitated and feasible.

Select a non-central viewpoint, designated as V_X^t, chosen from the set of original real viewpoints in which the best construction of its image I_X^t is desired for time t. The selection of viewpoint V_X^t is based on a round-robin schedule where every viewpoint is successively selected in a cycle. In Figure 3, a round-robin rotational scheme for four non-central views is shown where the selection of each image is related to time t in the following manner: I_L^t, I_R^{t+1}, I_T^{t+2}, I_B^{t+3}, I_L^{t+4}, I_R^{t+5}, I_T^{t+6}, etc, distinguished by a double border. Following, transmit the selected viewpoint vector V_X^t for time t.

The next step is to interpolate the selected non-central viewpoint image, referred to as predicted image PI_X^t, by rendering the 3D mesh model M^t in the specified viewpoint V_X^t. This interpolation step requires rendering a wireframe image of the desired viewpoint by simple geometric transformations with perspective projections of the mesh model, followed by texture mapping the corresponding areas of the central image onto the appropriate wireframe substructures.

Subsequently, calculate the prediction errors PE^t required for the final reconstruction of the selected view, by examining the difference between the original image I_X^t and the predicted image PI_X^t. Finally, encode and transmit the residual prediction errors PE^t. Before moving on to the next frame, determine if the allocated bit rate allows transmission of an additional non-central image. If bandwidth allows, the prediction errors for another view can be determined and send. Alternatively, a new round-robin schedule can be adopted where two viewpoints are selected for every time, thus perfect reconstruction is always achievable with unlimited bandwidth. Subsequently, the entire process starts all over for the next frame.

4.2 Multi-Viewpoint Decoding System

A block diagram of the proposed multiview decoding system is illustrated in Figure 4 for construction of any viewpoint images. First, the received bitstream of the central view is decoded on the main profile of MPEG-2, whose decoded images, denoted $\widehat{I_C^t}$, are stored in memory. Second, the bitstream of the depth map is decoded and hereafter referred to as $\widehat{D_C^t}$. Next, obtain a 3D mesh model, called M^t, for the central image $\widehat{I_C^t}$ as performed in the encoding system. Consequently, store the model M^t for time t in memory.

Upon receiving the selected viewpoint vector V_X^t for time t, store the vector in memory. Following, the prediction errors associated with the selected viewpoint V_X^t is decoded, denoted as $\widehat{PE^t}$. Now, determine the viewpoint requested from the user at the decoding system for time

t, assigned V_U^t. Subsequently, the user-requested viewpoint image (whether real or virtual) is interpolated, referred to as predicted image PI_U^t, where the index U designates the appropriate viewpoint V_U^t. The prediction PI_U^t is constructed by rendering the 3D mesh model M^t in the specified viewpoint V_U^t by the same viewpoint image interpolation procedure as in the encoder.

Determine if the encoder-selected non-central viewpoint V_X^t is similar to the user-requested viewpoint V_U^t. If the same, $V_U^t = V_X^t$, then the user can immediately construct the desired image. The final reconstructed image $\widehat{I_X^t}$ is obtained by combining the prediction errors $\widehat{PE}^t$ with the predicted image PI_X^t. This kind of reconstruction is thereafter defined as *Type I Prediction*. Following, the reconstructed non-central image $\widehat{I_X^t}$ is stored in memory for later reference. Satisfying the user's request for this viewpoint image, the whole process is repeated for the next time frame.

On the other hand, if the user did not request the same viewpoint as the one selected by the encoder, $V_U^t \neq V_X^t$, the following steps are carried out to generate such a view. Given the user-requested viewpoint V_U^t, retrieve from memory the image $\widehat{I_U^{t-f}}$ corresponding to the nearest past image I_U reconstructed by a Type I Prediction. Similarly, retrieve the image $\widehat{I_U^{t-b}}$ corresponding to the nearest future image I_U reconstructed by a Type I Prediction. For example, referring to Figure 3, if the user requests viewpoint V_R at time $t+2$, then the front image is I_R^{t+1} ($f = 1$) and the back image is I_R^{t+5} ($b = 3$). Due to the round robin schedule for one non-central viewpoint selection during each time frame, the maximum value of f and b is limited by $N-2$, where N is the number of viewpoints. Note that a delay may be required in order to access the future image.

Following, load the 3D mesh model M^{t-f} created at time $t-f$ from $\widehat{I_C^{t-f}}$ and $\widehat{D_C^{t-f}}$. In parallel, load the 3D mesh model M^{t+b} created at time $t+b$ from $\widehat{I_C^{t+b}}$ and $\widehat{D_C^{t+b}}$. Next, generate the front mesh image MI_U^{t-f} by locating a grid of fixed points on the image $\widehat{I_U^{t-f}}$. Also, generate the back mesh image MI_U^{t+b} by locating a corresponding grid of fixed points on the image $\widehat{I_U^{t+b}}$. Afterwards, generate the intermediate mesh-predicted image MPI_U^t for time t by digitally image warping between the front mesh image MI^{t-f} and the back mesh image MI^{t+b}. The image warping technique and generating mesh images by locating appropriate fixed points are well explained in [15].

Finally, a construction for $\widehat{I_U^t}$ is obtained by combining the intermediate mesh-predicted image MPI_U^t with the predicted image PI_U^t. The method for this final combination is left to the decoding system using any image fusion techniques[16][17], e.g., XOR operator, average, etc. Construction of this final image $\widehat{I_U^t}$ is termed as *Type II Prediction*. At this point, if other viewpoint images are requested by the users, the prediction steps may be repeated to generate other multiviews. Otherwise the decoding process starts at the beginning to decode the next time frame.

5 Conclusions

The proposed MPEG-2 compatible video coding specification for multi-viewpoint systems offers many advantages. In addition to providing a structural framework for the new multiview profile of the MPEG-2 coding standard, this codec retains the same encoding and decoding processing for one basic viewpoint sequence based on the main profile. Furthermore, because the standard only defines the bit-stream syntax and the decoding process, there is freedom in designing very high quality encoders and very low-cost decoders. Similarly, the presented specification suggests the same flexibility, providing for creativity and enhancements in accordance with advanced technologies.

Our novel contribution to the multiview coding research is mainly due to the concept of combining 2D image processing with 3D computer graphics. Many existing computer graphical tools and animation facilities, currently available in hardware, offer speedy rendering capabilities. Thus interpolation of any viewpoint image sequence can be generated quickly and efficiently. Also, construction of non-transmitted virtual viewpoints is possible due to availability of 3D models.

In rendering an image of a 3D surface structure, the textured image data are mapped onto corresponding three-dimensional geometric meshed surfaces. Consequently, texture mapping automatically incorporates the fore-shortening effect of 3D surface curvatures. In addition, since the texture is a 3D function of its position in the object, tracking of image points is facilitated by knowing the real world 3D coordinates of each point. Furthermore, additional enhancements on the constructed images may be obtained by incorporating other graphical tools, e.g., illumination and shading. Thus, innovative ideas can be freely augmented onto our system as technologies progress.

References

[1] "MPEG Draft International Standard. ISO/IEC 13818-2: Generic Coding of Moving Pictures and Associated Audio: Video," 1995.

[2] B. L. Tseng and D. Anastassiou, "Compatible video coding of stereoscopic sequences using MPEG-2's scalability and interlaced structure," in *Int'l Workshop on HDTV '94*, (Torino, Italy), Oct. 1994.

[3] A. Puri, R. Kollarits, and B. Haskell, "Stereoscopic video compression using temporal scalability," in *Proceedings of SPIE Visual Communications and Image Processing '95*, (Taipei, Taiwan), May 1995.

[4] T.-H. Chiang and Y.-Q. Zhang, "Stereoscopic video coding," in *Symposium on Multimedia Communications and Video Coding*, (New York City), to appear in Oct. 1995.

[5] B. L. Tseng and D. Anastassiou, "Perceptual adaptive quantization of stereoscopic video coding using MPEG-2's temporal scalability structure," in *International Workshop on Stereoscopic and Three Dimensional Imaging IWS3DI '95*, (Santorini, Greece), Sept. 1995.

[6] W. Burger and B. Bhanu, "Estimating 3-D egomotion from perspective image sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 1040–1058, Nov. 1990.

[7] B. K. P. Horn, *Robot Vision*. Cambridge, Massachusetts: The MIT Press, 1986.

[8] S. Barnard and M. Fischler, "Computational stereo," *ACM Computing Surveys*, vol. 14, pp. 553–572, Dec. 1982.

[9] S. Mitra, "Vision models for 3D surfaces," *SPIE Vol 1826 Intelligent Robots and Computer Vision XI*, pp. 182–188, 1992.

[10] B. L. Tseng and D. Anastassiou, "A theoretical study on an accurate reconstruction of multiview images based on the viterbi algorithm," in *International Conference on Image Processing ICIP '95*, (Washington, D.C.), Oct. 1995.

[11] T. Homma, "MPEG Contribution 95/N0861: Report of the Ad Hoc Group on MPEG-2 Applications for Multi-Viewpoint Pictures," Mar. 1995.

[12] G. Bozdagi, A. M. Tekalp, and L. Onural, "3-D motion estimation and wireframe adaptation including photometric effects for model-based coding of facial image sequences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, pp. 246–256, June 1994.

[13] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice*. Reading, Massachusetts: Addison Wesley Publishing Company, second ed., 1990.

[14] J. Neider, T. Davis, and M. Woo, *OpenGL Programming Guide*. Addison-Wesley Publishing, 1993.

[15] G. Wolberg, *Digital Image Warping*. Los Alamitos, California: IEEE Computer Society Press, 1990.

[16] C.-P. Yeh, "Depth perception based on fusion of stereo images," *SPIE Vol. 1778 Imaging Technologies and Applications*, pp. 221–226, 1992.

[17] Y. T. Zhou, "Multi-sensor image fusion," in *International Conference on Image Processing*, (Austin, Texas), Nov. 1994.

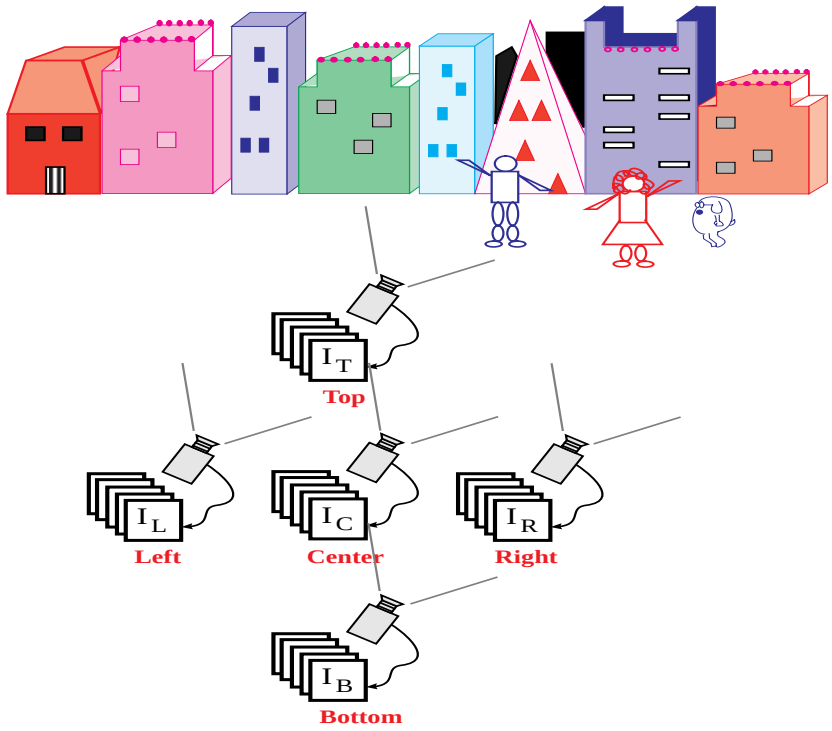

Figure 1: Camera Configuration of Multiple Viewpoint Images

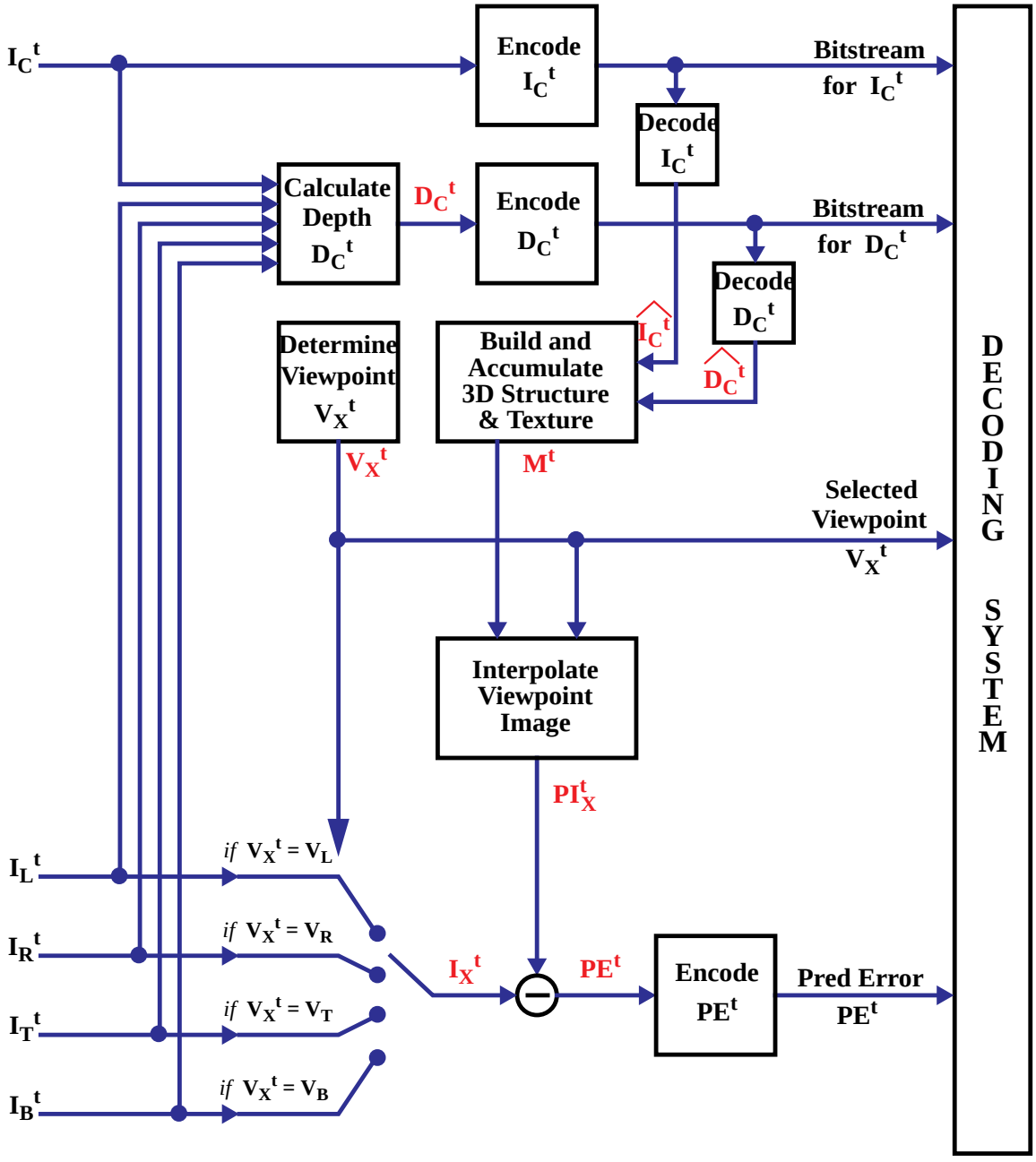

Figure 2: Block Diagram of Multiview Encoding System

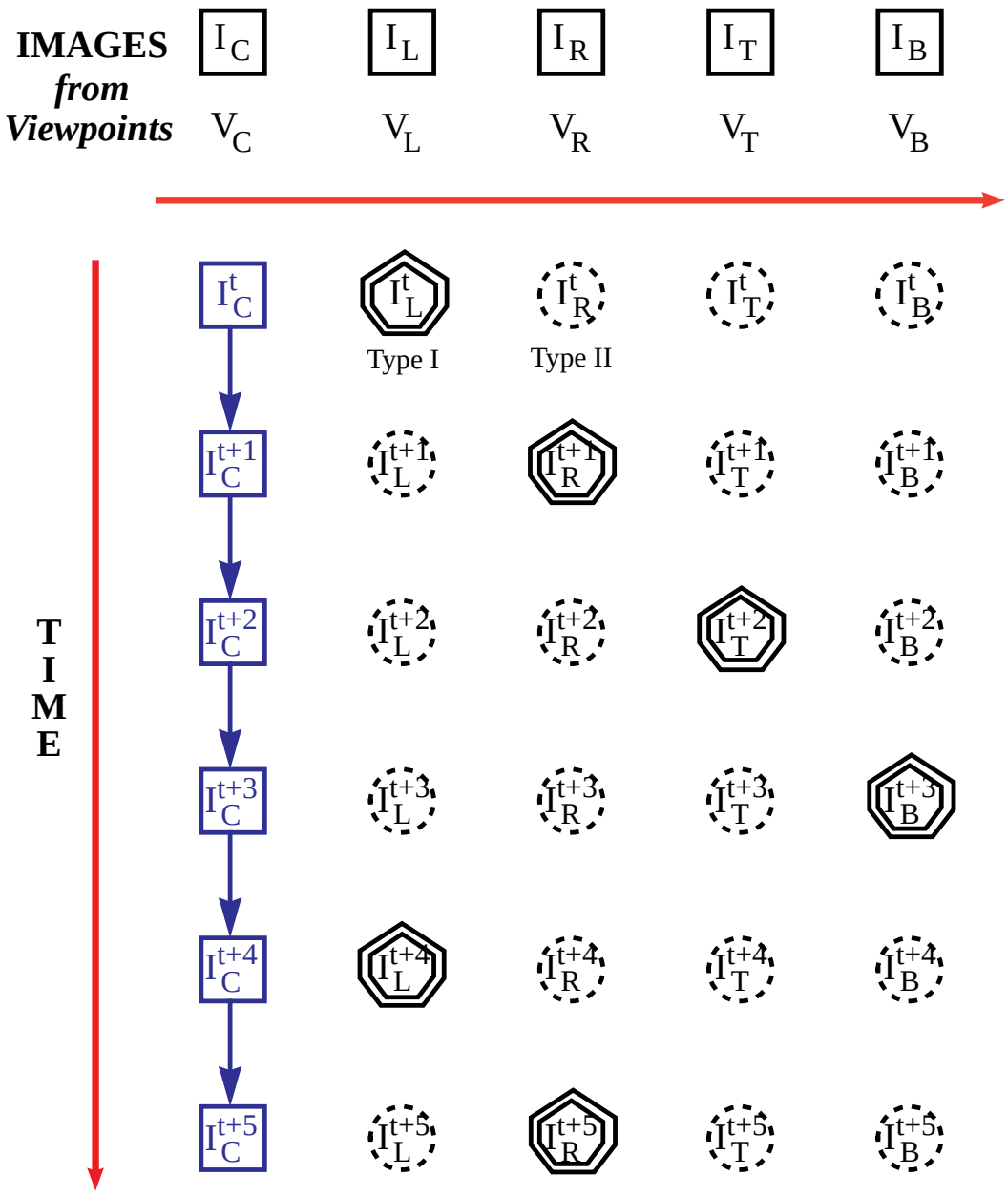

Figure 3: A Round-Robin Rotational Schedule for Encoder Selection of One Viewpoint

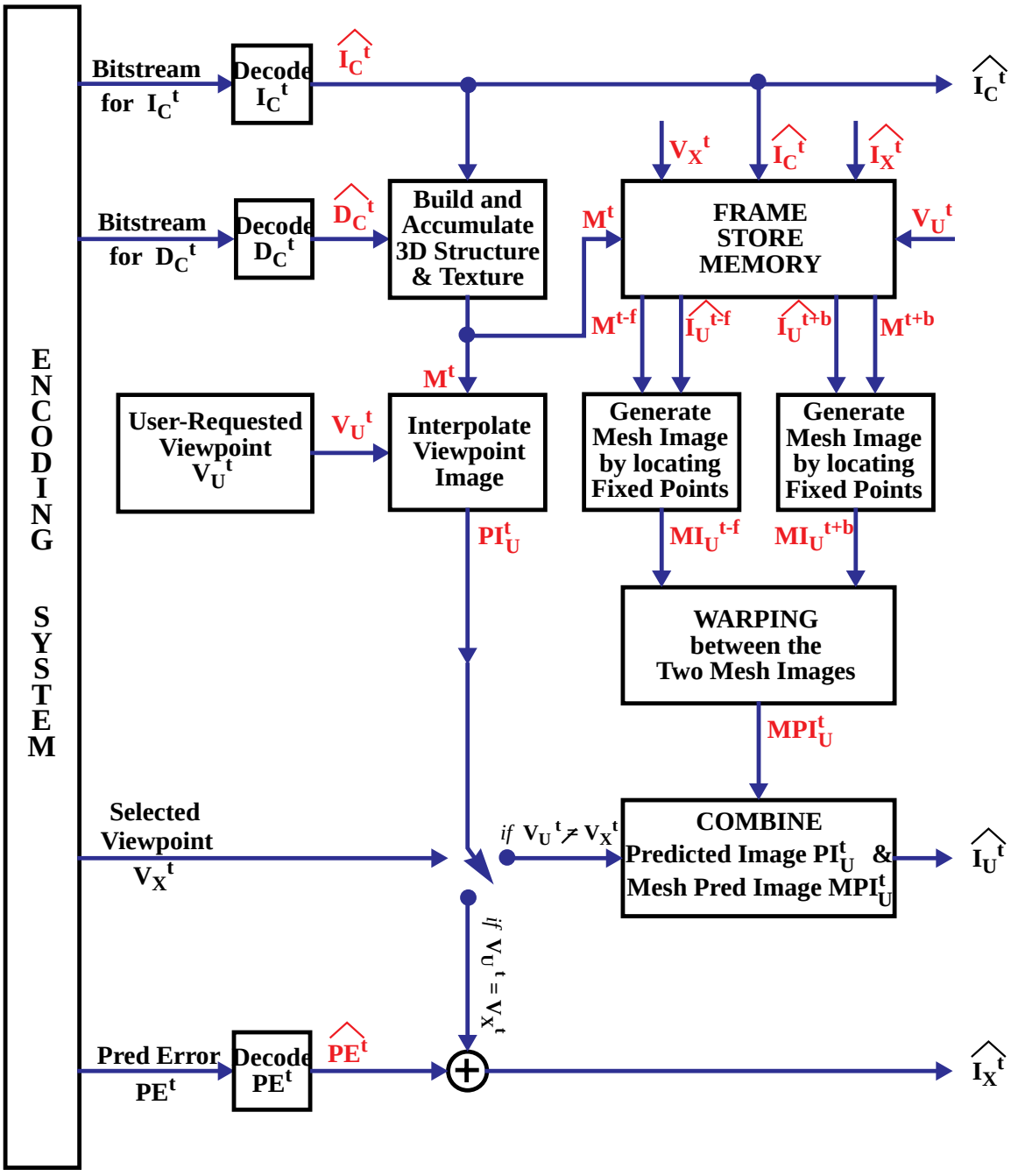

Figure 4: Block Diagram of Multiview Decoding System