Cubic-Panorama Image Dataset Compression for Image-Based Virtual Environment Navigation

Kehua Jiang

Ottawa-Carleton Institute for Electrical and Computer Engineering
School of Information Technology and Engineering (SITE)
University of Ottawa
Ottawa, Ontario, Canada

December 2008

A thesis submitted to the Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of Doctor of Philosophy
in Electrical Engineering

Contents

Abstract

This thesis addresses the key issues of image dataset compression, especially the cubic-panorama image dataset compression, used in image-based virtual environment navigation to develop effective and efficient compression techniques and schemes. To my knowledge, this is the first work to be published to investigate and design compression schemes for cubic-panorama image datasets.

A spatially consistent representation of cubic-panorama image datasets is proposed. With this spatially consistent representation, unrestricted search for displacement vectors as well as matching reference blocks is extended beyond side image boundaries in all directions. A block padding algorithm for constructing reference blocks is presented for displacement estimation and compensation. Optimized matching reference blocks are obtained to reduce the prediction errors and improve the compression efficiency. Superior coding performance is achieved with the spatially consistent representation compared with the generic planar representation of cubic-panorama image datasets.

For cubic-panorama image dataset compression, a scalable lifted wavelet-based coding scheme with displacement compensation is developed. This scheme is based on the framework of lifted wavelet transforms with cross-image displacement-compensated enhancement combined with embedded entropy coding. The wavelet transforms efficiently generate hierarchically structured decomposition coefficients and provide the potential of spatially scalable coding required by image dataset compression. The lifting operations put wavelet analysis and synthesis into fast memory-saving in-place computations. The displacement compensation significantly improves the wavelet transform coding efficiency across basis images.

A new embedded entropy coding approach named independent block with layered data partition *(IBLDP)* is proposed. It combines embedded independent coefficient block coding with layered bitplane data partitioning. To provide the required spatial resolution scalability and spatial random access, no inter-subband and inter-block dependencies are exploited. Each subband is encoded independent of other subbands. With the compact layered bitplane data representation and the simplified coding structure, *IBLDP* features reduced computational and implemental complexity and hence is more suitable for the efficient and interactive image rendering application.

For cubic-panorama image dataset compression, a specific random access mechanism is designed. A new hierarchical data structure is proposed to accommodate the random access mechanism. The corresponding bit-stream syntax is formed to support this new data structure. Multi-level index tables are embedded in the bit-stream to easily facilitate the spatial image random access. A proper compromise between the coding efficiency and the random access flexibility is reached.

A novel global displacement estimation and compensation approach with scaled block-depth estimation is developed. Based on a six-parameter perspective projection model, a displacement-compensated image prediction algorithm is presented, taking into consideration all kinds of camera motions. A unique block-based scaled depth estimation technique is proposed for image prediction. The displacement-compensated predicting images are generated by using the global model parameters combined with the estimated scaled block-depth map. More accurate predicting images with less-data presentations are obtained compared with the traditional *BMA*. The proposed approach is more efficient when applied for cubic-panorama image dataset compression.

Experimental results of the proposed techniques and schemes applied to encoding the testing cubic-panorama image datasets are demonstrated. Superior coding performance are achieved over that of the corresponding comparable techniques and schemes applied for cubic-panorama image dataset compression.

Acknowledgments

In my deep heart, I am full of gratitude to my supervisor Prof. Eric Dubois, who has always been very responsible, helpful, patient and gave me a lot of precious guidance in various aspects of study and research during the whole period of my study for a PhD degree. He has always been insisted to give the in-time directions in the regular meetings all these years even when he was quite busy while also working as the director of School of Information Technology and Engineering. I am sure that I will continue to benefit from the instructive guidance he gave me after I graduate from this university.

My thanks go to Prof. Rafik A. Goubran and Prof. Pierre Payeur for their serving in my advisory committee and giving me valuable advices during the comprehensive examination and the thesis proposal defence. I wish to acknowledge the financial support from NSERC through the Navire and Lornet research projects. Also, my thanks are given to my colleagues in the VIVA lab for their help while we worked together in these research projects.

I would like to thank my parents for their continuous encouragements during my whole study period. Also, I would like to thank my wife for her patience and support in these years. Finally, my thanks go to my daughter, who has been making progresses in her study and life all the time and often brought her refreshing good news to my monotonous study life.

List of Figures

List of Tables

List of Symbols

Superscripts

Br	Bayer format image
Rgb	RGB format image
Yuv	YUV format image

Notations

$\mathbf{A}_{i,\,j}$	Significance state array of coefficient block $CB_{i,\,j}$
$\mathbf{B}_{c,\,n}$	n^{th} sample block in the current image
$\mathcal{C}_{\mathcal{B},k}$	DCT coefficients of $\mathcal{I}_{\mathcal{B},\,k}$
$C_{i,\,j}$	Wavelet decomposition coefficient in $CB_{i,\,j}$
$\mathcal{C}_{M,\,k}$	k^{th} model context
$c_{H,\,l}$	l^{th} high-pass decomposition coefficient
$c_{L,\,m}$	m^{th} low-pass decomposition coefficient
$CB_{i,\,j}$	j^{th} decomposition coefficient block in the i^{th} subband
$D_{i\rightarrow j}$	Displacement-compensated mapping from $\mathcal{I}_{\mathcal{B},\,i}$ to $\mathcal{I}_{\mathcal{B},\,j}$
$\bar{\mathbf{D}}_{\mathcal{B}}$	Basis image domain
$\bar{\mathbf{D}}_{\mathcal{B},\,j}$	j^{th} sub-domain of $\bar{\mathbf{D}}_{\mathcal{B}}$
$\bar{\mathbf{D}}'_{\mathcal{B},\,j}$	Selective decoding sub-domain corresponding to $\bar{\mathbf{D}}_{\mathcal{B},\,j}$
$\bar{\mathbf{D}}_{\mathcal{E}}$	Real-world environment domain
$\bar{\mathbf{D}}_{\mathcal{R}}$	Raw image domain
$\bar{\mathbf{D}}_{\mathcal{S}}$	Two-dimensional synthesized image domain

$\mathbf{d}_s$	Scaled depth
$\mathbf{d}_{s,\,n}$	Scaled block-depth of the n^{th} block
DC_i^P	i^{th} stage displacement compensation for the prediction operation
DC_i^U	i^{th} stage displacement compensation for the update operation
$\mathcal{E}_{\mathcal{B},k}$	Prediction error of $\mathcal{I}_{\mathcal{B},\,k}$
$\mathcal{E}_{MS}$	Mean square error
$\mathbf{G}_0(z),\ \mathbf{g}_0(n)$	Synthesis low-pass filter
$\mathbf{G}_1(z),\ \mathbf{g}_1(n)$	Synthesis high-pass filter
G_H, G_L	High-pass and Low-pass output scaling factors
$\mathbf{H}_0(z),\ \mathbf{h}_0(n)$	Analysis low-pass filter
$\mathbf{H}_1(z),\ \mathbf{h}_1(n)$	Analysis high-pass filter
$\mathcal{I}_{\mathcal{B},\,c},\ \mathcal{I}_{\mathcal{B},\,r}$	Current and reference basis images
$\mathcal{I}_{\mathcal{B},\,k}$	k^{th} basis image in an image dataset
$\hat{\mathcal{I}}_{\mathcal{B},k}$	Reconstructed $\mathcal{I}_{\mathcal{B},\,k}$
$\tilde{\mathcal{I}}_{\mathcal{B},\,k}$	Predicting image of $\mathcal{I}_{\mathcal{B},\,k}$
$\mathcal{I}_{\mathcal{B},\,kj}$	j^{th} side image in basis cubic panorama $\mathcal{I}_{\mathcal{B},\,k}$
$\mathcal{I}_{\mathcal{B},\,k,\,j}$	j^{th} image segment in $\mathcal{I}_{\mathcal{B},\,k}$
$\mathcal{I}_P$	Image samples in the padding block
$\mathcal{I}'_P$	Modified image samples in the padding block
$\mathcal{I}_{\mathcal{R},\,l}$	l^{th} raw image in a raw image sequence
$\mathcal{I}_{\mathcal{R},\,ki}$	Raw image captured with the i^{th} lens of Ladybug in $\mathcal{I}_{\mathcal{R},\,k}$
$\mathcal{I}_{\mathcal{S},\,m}$	m^{th} synthesized image in a synthesized image sequence
$\mathbf{L}_\mathbf{C}$	Number of cross-image DWT levels
$\mathcal{L}_\mathcal{C}$	Sampling structure of the cubic-panorama basis images
$\mathbf{L}_d$	Dimensionality of the plenoptic function
L_n	n^{th} quad-tree level
$\mathcal{L}_P$	Plenoptic function of the light ray intensity distribution
$\mathcal{L}_{P5}$	Five-dimensional plenoptic function

$\mathcal{L}_{P6}$	Six-dimensional plenoptic function
$\mathbf{N_L}$	Number of spatial wavelet analysis levels
$N_{m,\,n}$	Quad-tree node in quad-tree structure $T_{m,\,n}$
$P_i(z)$	i^{th} stage of prediction operation
$\mathbb{P}_M$	Model parameter set
$\mathcal{P}_m$	m^{th} bitplane
Q_F	Frame quantization parameter
$Q_{i,\,j}$	Quantization index in $CB_{i,\,j}$
$q_{i,\,j,\,m}$	Binary symbol of quantizer index $Q_{i,\,j}$ on the m^{th} bitplane
$\mathcal{R}_C$	Compression ratio
$\boldsymbol{\tilde{\mathcal{S}}}_{\mathcal{B}}$	Basis image signal space
S_E	Sum of squared errors
$S_{H,\,k}$	k^{th} high-pass subband decomposition image
$S_{L,\,k}$	k^{th} low-pass subband decomposition image
$\tilde{\mathbb{S}}_I$	Image dataset
$\boldsymbol{\tilde{\mathcal{S}}}_{\mathcal{R}}$	Raw image signal space
$s_{i,\,j}$	Significance state variable in $CB_{i,\,j}$
SB_i	i^{th} subband
$T_{m,\,n}$	Quad-tree structure at the n^{th} quad-tree level on the m^{th} bitplane
T_x, T_y, T_z	Translation vector components
$U_i(z)$	i^{th} stage of update operation
$X_{c,\,m}, Y_{c,\,m}$	Coordinates of the m^{th} correspondence pair in the current image
$X_{r,\,m}, Y_{r,\,m}$	Coordinates of the m^{th} correspondence pair in the reference image
$\bar{\mathbf{x}}$	Position vector in the basis image domain
$\hat{\mathbf{x}}$	Position vector in the synthesized image domain
$\Delta x, \Delta y, \Delta z$	Scaled translation vector components
δ_i	Quantization step-size in the i^{th} subband
$\tilde{\delta}_i$	Scaled quantization step-size in the i^{th} subband

η_h, η_v	Horizontal and vertical sign context functions
Λ_C, Λ_L	Sampling lattices of chrominance and luminance components
$\Lambda_r, \Lambda_g, \Lambda_b$	Sampling lattices of R, G, B components
μ_d, μ_h, μ_v	Accumulated significance states of the three categories
ρ_s	Complementary state variable
τ_h, τ_v	Horizontal and vertical neighboring sign statuses
$\tau_{i,j}$	Sign of $Q_{i,j}$
τ_p	Sign prediction

$\mathcal{I}_{\mathcal{B},kj,R}$, $\mathcal{I}_{\mathcal{B},kj,G}$, $\mathcal{I}_{\mathcal{B},kj,B}$	R, G, B components of $\mathcal{I}_{\mathcal{B},kj}$
$\mathcal{I}_{\mathcal{B},kj,Y}$, $\mathcal{I}_{\mathcal{B},kj,U}$, $\mathcal{I}_{\mathcal{B},kj,V}$	Y, U, V components of $\mathcal{I}_{\mathcal{B},kj}$
$\mathcal{I}_{\mathcal{B},ku}$, $\mathcal{I}_{\mathcal{B},kb}$, $\mathcal{I}_{\mathcal{B},kl}$, $\mathcal{I}_{\mathcal{B},kf}$, $\mathcal{I}_{\mathcal{B},kr}$, $\mathcal{I}_{\mathcal{B},kd}$	Up, back, left, front, right and down side images in basis cubic panorama $\mathcal{I}_{\mathcal{B},k}$
$\mathbf{T}_u$, $\mathbf{T}_b$, $\mathbf{T}_l$, $\mathbf{T}_f$, $\mathbf{T}_r$, $\mathbf{T}_d$	Translation vectors of up, back, left, front, right and down side images
$\mathbf{R}_u$, $\mathbf{R}_b$, $\mathbf{R}_l$, $\mathbf{R}_f$, $\mathbf{R}_r$, $\mathbf{R}_d$	Rotation matrices of up, back, left, front, right and down side images

Chapter 1

Introduction

Navigation in image-based virtual environments is desirable in many applications, such as virtual museums, immersive learning, virtual sightseeing, telepresence, computer games, e-business, real-estate sales, etc. [7, 10, 31, 52, 88]. Compression of the image datasets used as input for image-based virtual environment navigation systems is indispensable for reducing the huge database sizes and making the applications feasible in practical use. As a specific instance of image datasets used for image-based virtual environment navigation, cubic-panorama image datasets with their unique features are becoming more popular and expected to be more widely utilized [7, 21, 90]. Compression of cubic-panorama image datasets confronts some substantial challenges in order to adapt the coding schemes to the features of the image datasets, satisfy the requirements of navigation systems and efficiently represent the image datasets in the compressed streaming output. This thesis addresses and resolves some of the challenges for the purposes of efficiently compressing image datasets, especially the cubic-panorama image datasets and meeting well the specific requirements of the applications.

In this introductory chapter, first the necessity for conducting research on compression of image datasets as well as the specific cubic-panorama image dataset instance used for image-based virtual environment navigation is clarified in Section 1.1. Then, an overview of the research work covered in this thesis is given in Section 1.2. After that, the organization of this thesis is outlined at the end of this introductory chapter in Section 1.3.

1.1 Motivation

An image-based virtual environment navigation system can provide users with realistic and immersive exploration experiences of real-world environments by taking image datasets as the primitive inputs. As a discrete image representation of a real-world environment, an image dataset consists of a collection of basis images in a particular image format (Fig. 1.1). Each basis image in the image dataset is associated with a distinctive viewpoint at a physical location on the designed trajectories. Therefore, an image dataset also refers to a number of image samples generated at a sequence of viewpoints in the represented environment. By making use of image-based rendering (*IBR*) techniques, novel-view images can be synthesized and new image sequences can be ren-

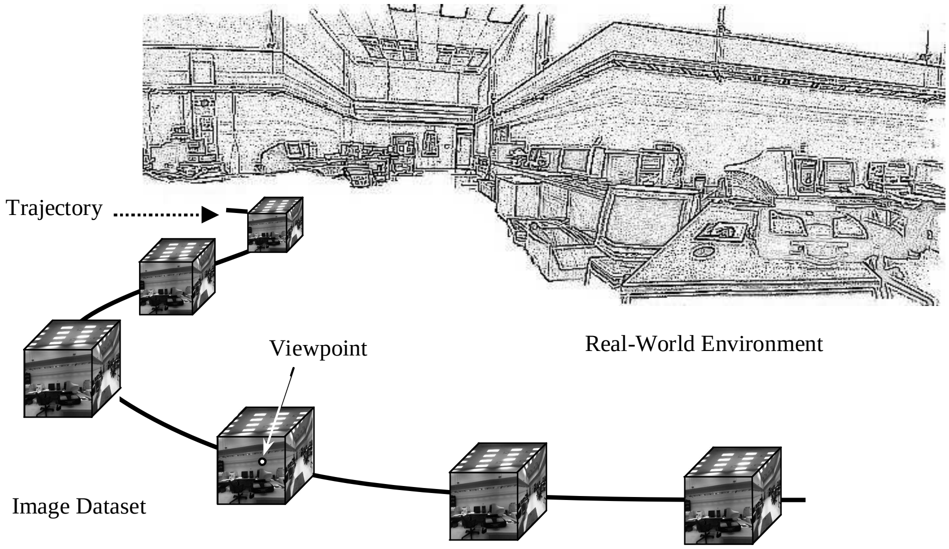


Fig. 1.1 An image dataset consisting of a collection of basis images in a particular image format as a discrete image representation of a real-world environment.

dered according to users' instructions. Remote users are able to interactively extract and observe the rendered image sequences. A resultant image sequence imitates a video sequence as would be captured by a virtual camera. This virtual camera moves with the desired orientations along the navigation trajectories controlled by users according to their exploration intents. Thus, users are able to navigate in the image-based replica of a real-world environment.

In the process of navigating in an image-based virtual environment, photo-realistic novel views are synthesized based on the image datasets representing the environments. One of the most important form of image dataset is the cubic-panorama image dataset. A cubic panorama (Fig. 1.2) consists of six planar perspective side images stitched together to provide viewers with a 360^o horizontal view and with the capability to look up to the ceiling and down to the floor. Cubic panoramas can be easily generated and manipulated. Novel-view images can be synthesized conveniently from the basis images in cubic-panorama datasets. The best general-purpose representation of environment mapping is the projection onto a cube [25]. Cubic-panorama image datasets are more suitable for being used as the image database in some applications, like navigation in image-based virtual environments. As a unique format of panoramic images and a promising form of image dataset used for image-based virtual environment navigation,

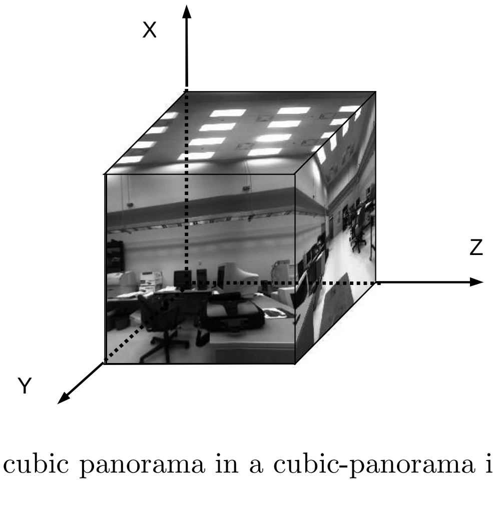


Fig. 1.2 A cubic panorama in a cubic-panorama image dataset.

cubic-panorama image datasets are becoming more popular and are expected to be more widely applied [7, 21].

Image datasets constitute the fundamental databases for image-based virtual environment navigation. In order to completely cover the represented environment, basis images in image datasets are usually densely distributed in space and a larger number of basis images are included in image datasets. They involve a huge amount of data. For example, the Lumigraph [24] and light field [53] experimental image dataset “Fruit” consisting of a two-dimensional image array with 32 x 32 basis images of resolution 256 x 256 has a size of 192 MB, and the cubic-panorama [25] experimental image dataset “Lab” consisting of a one-dimensional image array with 112 basis images of resolution 6 x 512 x 512 has a size of 504 MB. In practical applications, a much larger number of higher resolution images are desirable in order to represent large environments, enable large available navigation spaces and synthesize high-quality novel images. This results in even larger sizes of image datasets. Large image datasets occupy huge amounts of storage space on storage media and require high transmission bandwidths over networks. Compression of the image datasets is crucial to reducing the required storage space and transmission bandwidth and putting image-based virtual environment navigation into practical use.

Intensive research on compressing images and video sequences has been conducted for many years. Many image and video compression techniques have been proposed and are available for different applications. Some of the state-of-the-art compression techniques have been recommended in a variety of international image and video coding standards, ranging from ISO/IEC JPEG [33] and JPEG2000 [34] for still image coding, ISO/IEC MPEG-1 [35], MPEG-2 [36] and MPEG-4 [37] as well as ITU-T H.261 [39] and H.263 [40] for video coding up to the recent H.264 (also named MPEG-4 AVC) [41] jointly developed by ITU-T and ISO/IEC for advanced video coding. However, as these image and video compression techniques are designed for compressing generic images and video sequences, they are sub-optimal if directly applied for compressing the image datasets used for image-based virtual environment navigation.

Image dataset compression is significantly different from the generic image and video sequence compression. Making the compression techniques adaped to the specific features of the image datasets is very important to achieve high-performance image dataset compression, and meeting the special requirements for image-based virtual environment navigation is the key to the design of image dataset compression schemes. Compression techniques and schemes specially designed for compressing image datasets need to be developed.

1.2 Thesis Work Outline

This thesis addresses the challenging problems of image dataset compression, especially the cubic-panorama image dataset compression, used for image-based virtual environment navigation. Although not all aspects of cubic-panorama image dataset compression could be dealt with due to the limits of time and space, this thesis investigates a number of the very key issues. It is, to the author's knowledge, the first publicly available work on cubic-panorama image dataset compression. Several contributions to the research on cubic-panorama image dataset compression are made in this thesis.

Image dataset compression is a new research area. Many issues remain to be investigated in various aspects of image dataset compression. In this thesis, based on an intensive study of the problems of image dataset compression and an extensive relevant literature review, the common characteristics of image datasets used for image-based virtual environment navigation are systematically summarized, and the requirements, the key issues and the strategies of image dataset compression are generalized. Also, a comprehensive review and comparison of existing solution methods are presented. All these provide a valuable reference to the research on image dataset compression.

Adapting coding schemes to the special features of image datasets is very important in the design of a scheme for efficient image dataset compression. Taking the cubic-panorama image dataset as a concrete form of image dataset used in image-based virtual environment navigation, a spatially consistent representation of cubic-panorama image datasets is proposed by making use of the features of this specific form of image dataset. With this spatially consistent representation, unrestricted search for displacement vectors and matching reference blocks as well can be performed. A block padding algorithm for constructing reference blocks is presented for displacement estimation and compensation. Optimized matching reference blocks are obtained to reduce the prediction errors and improve the compression efficiency. With the proposed spatially consistent representation, superior coding performance is achieved over that with the generic planar representation of cubic-panorama image datasets.

Providing spatially scalable coding is extremely desirable for image dataset compression in order to synthesize the frequently required zooming views with steady image quality. Based on this fundamental requirement, a scalable lifted wavelet-based coding scheme with displacement compensation is specially developed for cubic-panorama image dataset compression. This scheme is based on the framework of lifted wavelet transforms with cross-image displacement compensation combined with embedded entropy coding. The wavelet transforms provide the potential of required spatially scalable coding. The lifting operations put wavelet analysis and synthesis into fast memory-saving in-place computations. The displacement compensation significantly improves the wavelet transform coding efficiency across basis images. A new embedded entropy coding approach is proposed. It combines embedded independent coefficient block coding with layered bitplane data partitioning. This new entropy coding approach features reduced computational and implementation complexity and hence is more suitable for the efficient and interactive image rendering application.

With regard to the unique requirements of random access and selective decoding for image dataset compression, a specific rectangular sub-region access mechanism is designed for cubic-panorama image dataset compression. A new hierarchical data structure is proposed to accommodate this mechanism. The corresponding bit-stream syntax is formed to support this new data structure. Multi-level index tables are embedded in the bit-stream to easily facilitate the required random access. A suitable tradeoff among coding efficiency, spatial resolution scalability and random access flexibility is reached.

Making full use of the features of image datasets is a principle strategy of designing efficient coding schemes for image dataset compression. As a specific feature of image datasets, the dominant image displacement across basis images demonstrates a certain regular pattern for each form of image dataset determined by the motion of the camera or cameras producing the image dataset. A novel global displacement estimation and compensation approach is developed based on the dominant image displacement feature of cubic-panorama image datasets. A displacement-compensated image prediction algorithm is presented, taking into consideration all kinds of camera motions. A unique block-based scaled depth estimation technique is proposed for image prediction. Displacement-compensated predicting images are generated by using the global model parameters combined with the estimated scaled block-depth map. More accurate predicting images with more compact representations are obtained compared with the traditional block matching algorithms (*BMA*). This proposed approach is more efficient when applied for cubic-panorama image dataset compression.

Experimental results of the proposed techniques and schemes applied to encoding the testing cubic-panorama image datasets are demonstrated. Superior coding performances are achieved over those of the corresponding comparable techniques and schemes applied for cubic-panorama image dataset compression.

1.3 Organization of the Thesis

This thesis is organized as follows. After this introductory chapter, Chapter 2 provides the key background information on image dataset compression used for image-based virtual environment navigation. An introduction to an image-based virtual environment navigation system is made along with brief descriptions of the main components in the system, including the image dataset compression component. Also, image-based rendering techniques, which build the basis of image-based virtual environment navigation and are closely related to the design of image dataset compression schemes, are surveyed in this background information chapter.

Chapter 3 covers the generic issues of image dataset compression for image-based virtual environment navigation. It starts out with a problem description section, summarizes the characteristics of image dataset compression, and presents a comprehensive review of developed solution methods for compressing image datasets. Lastly, comparisons of different image dataset compression schemes are made, and some conclusions are drawn in a summary section.

In Chapter 4, cubic-panorama image dataset compression with a spatially consistent representation is discussed. After an introduction to the cubic-panorama image dataset with its superior features over other forms of image datasets, the process of generating cubic-panorama image datasets is presented. A spatially consistent representation for cubic-panorama image datasets is proposed, and a reference block padding algorithm is presented and applied to unrestricted displacement estimation and compensation. Two baseline compression schemes adapted to coding cubic-panorama image datasets with the spatially consistent representation are introduced and experimental results are provided. This chapter ends with a summary of the discussion on cubic-panorama image dataset compression.

The development of a scalable lifted wavelet-based coding scheme with displacement compensation for cubic-panorama image dataset compression is covered in Chapter 5. Following a problem description of scalable coding and wavelet-based compression techniques, all the main aspects of the developed compression scheme are discussed in detail, including discrete wavelet transforms with lifting schemes, displacement-compensated cross-image filtering, wavelet decomposition coefficient quantization and context-based arithmetic entropy coding. Also, spatial subregion random access and selective decoding for cubic-panorama image dataset compression are investigated and the structure of the scalable lifted wavelet-based scheme with displacement compensation is presented. In addition, experimental results of this developed compression scheme applied to coding testing cubic-panorama image datasets are demonstrated, and the development of this specially designed compression scheme is summarized at the end of this chapter.

Chapter 6 covers global displacement estimation and compensation used for cubic-panorama image dataset compression. It starts out with a section of introduction to global displacement estimation and displacement-compensated image prediction, followed by a review of the relevant research developments. After a discussion about global displacement analysis, a displacement-compensated image prediction algorithm with block-based scaled depth estimation is presented. The features of global displacements in cubic-panorama image datasets are investigated, and a block-based hybrid coding scheme with global displacement estimation and compensation is presented. The experimental results of the presented coding scheme are demonstrated. This chapter concludes with a summary section.

Finally, the contributions made in this thesis work are summarized and the future research work on compressing image datasets used for image-based virtual environment navigation is anticipated in Chapter 7.

Chapter 2

Navigation in Image-Based Virtual Environments

Preceding the discussions on image dataset compression, some necessary background knowledge is introduced in this chapter. The first part of this chapter presents the image-based virtual environment navigation system, out of which this thesis topic arises. In the second part of this chapter, image-based rendering techniques, which build the basis of image-based virtual environment navigation systems and are so closely related to the design of image dataset compression schemes, are reviewed with the emphasis on the image-based rendering techniques without geometric data, such as the cubic-panorama rendering technique.

2.1 Image-Based Virtual Environment Navigation System

Image-based virtual environment navigation allows users to walk (or generally move) through the virtual representations of real-world environments, provides them with appropriate spatial parallax, lighting and illumination changes in the virtual environments, and gives them realistic, immersive and interactive exploration experiences by making use of synthesized photo-realistic images. A practical system of image-based virtual environment navigation has been developed in the context of Navire (Virtual Navigation in Image-Based Representations of Real-World Environments) [62], which was an NSERC research project carried out in VIVA Lab at the University of Ottawa. With the objective of achieving effective and natural virtual navigation in image-based representations of real-world environments, the Navire project's goal was to address and solve the technical challenges in developing a practical system making use of image datasets mostly composed of cubic-panorama basis images. This research project covered all aspects of the image-based virtual environment navigation system. This thesis reflects the work done for solving the challenging problem of compressing the cubic-panorama image datasets in cooperation with VIVA Lab colleagues working in other aspects in the context of the Navire project [7, 21, 30, 47, 80, 104]. In this section, an image-based virtual environment navigation system involving image dataset compression as one of its fundamental components is presented first, followed by brief descriptions of the main components contained in the system. This section serves to provide background knowledge for image dataset compression and illustrate the connections between image dataset compression and other components of the system.

2.1.1 System Overview

An image-based virtual environment navigation system mainly consists of raw image acquisition, image sequence preprocessing, basis image analysis, image dataset compression, image sequence rendering, image sequence transmission and virtual environment navigation. A block diagram depicting the structure of an image-based virtual environment navigation system is shown in Fig. 2.1. Some practical image-based virtual environment navigation systems providing different exploration capabilities have been developed [7, 10, 88, 100].

An image-based virtual environment navigation system starts work with the raw

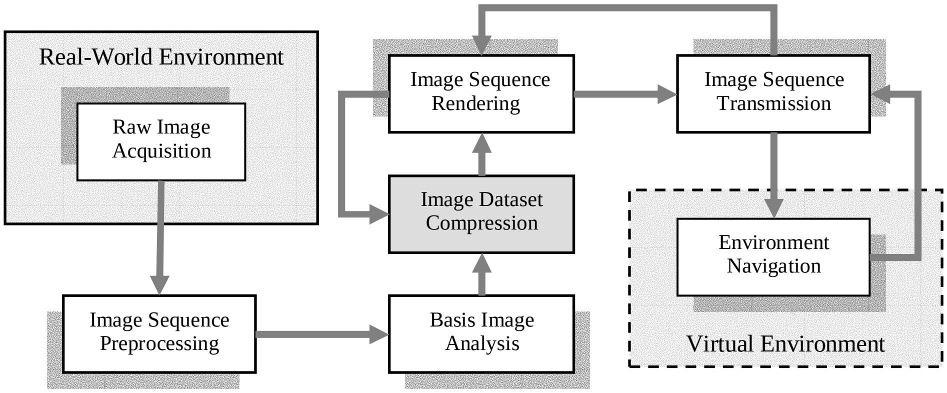


Fig. 2.1 Block diagram of an image-based virtual environment navigation system with its main components including image dataset compression.

image acquisition. This component captures the discrete representation of the sampled real-world environment and obtains an image sequence consisting of a collection of raw images. Following the raw image acquisition, the image preprocessing is performed with required preliminary operations like image signal correction, noise filtering, image format conversion and image reorganization. Then, the resulting basis images are analyzed for necessary camera calibration, image registration, image correspondence, depth information extraction and so on to obtain some auxiliary information that will assist the following processes.

The image dataset compression component puts the basis images and the auxiliary information into an efficient bit-stream representation to optimize the storage space allocation. Taking the compressed image bit-stream as primitive input, the image sequence rendering component synthesizes novel-view images and generates rendered image sequences according to users' instructions. The image sequence transmission component performs combined source and channel coding to efficiently encode the rendered image sequences, multiplex them with supplementary multimedia contents, adapt the multiplexed payloads to the features of the communication channels and deliver the resultant bit-streams via networks. Then, remote users are able to interactively extract and observe the synthesized image sequences that imitate video sequences captured by a virtual camera as it moves with the desired orientations along the navigation trajectories controlled by users according to their exploration intents.

2.1.2 Raw Image Acquisition

The raw image acquisition process generates the source input for the image-based virtual environment navigation system. With a camera imaging system, it captures the image representation for the environment of interest at physical locations in the real world. The procedure of image acquisition is designed and controlled to get an efficient and complete image representation for the real-world environment. The camera orientations and the camera motion trajectories for capturing the raw images are well planned to completely cover the environment of interest. On one hand, over-sampling the environment increases the burden of image acquisition and results in unnecessary image overlapping [9], making the size of the raw image sequence extremely large. On the other hand, under-sampling the environment creates the aliasing problem, causing serious image distortion. Spectral analysis [101] and plenoptic sampling theories [8, 54] provide the required guideline to avoid over-sampling and under-sampling in the raw image acquisition process.

An imaging system is usually built with one single camera [74], multiple cameras [7] or even a camera array combined with a data recording device for storing the captured raw images, a portable computer for controlling the image acquisition process and an on-board power source for providing steady power supply. In some cases, supplementary information, such as the scene depth distribution, camera poses and geo-reference data, is also required to be obtained together with raw images to reduce the image sampling density, support model building, improve the image compression performance or assist image rendering. In this scenario, some apparatus like a global positioning system (GPS) for capturing geo-reference data [7] or a laser rangefinder for obtaining scene depth information [94] are incorporated in the imaging system.

2.1.3 Image Sequence Preprocessing

Following the raw image acquisition, image sequence preprocessing performs necessary operations on raw images for image signal correction, noise filtering, image format conversion, image reorganization, etc. [5]. Image preprocessing is required in the system to prepare images for the following image processing processes.

Basic image preprocessing includes gamma correction, white balance and noise filtering. Gamma correction is performed with regard to the optoelectronic transfer characteristics of sensors. The power law relationship can be applied for this gamma correction purpose. White objects should generate equal values of R, G and B components. To correct the differences of R, G, and B values of a white object resulting from different illumination types and color temperatures, a white balance approach is applied by, for example, keeping the green pixel value G unchanged and multiplying the red and blue pixel values R and B by their respective gains. In addition, noise filtering for bad pixel replacement and noise reduction could also be applied upon request.

Image format conversion changes the format of raw image signals to the standard input image format required by image dataset compression. At this stage, basis images of the image dataset in the desired format are derived from raw images by a transformation, image projection or image mapping operation. Generally, the directly-output images from sensors are Bayer color images with only one color component at each sample position. In this scenario, the pixel values of the other two color components at each sample position are interpolated to obtain the full RGB image format. In some cases, a reorganization of images is required for easing image analysis, image dataset compression or image-based rendering. For example, rearranging a set of plane images into a collection of manifold mosaic images [55], putting groups of captured images into a group of panoramic images [74], and aligning or re-binning multi-perspective panoramas to obtain a set of new multi-perspective panoramas with increased inter image redundancy [55, 96] can be performed in the image preprocessing process.

2.1.4 Basis Image Analysis

Image analysis is applied to basis images of the image dataset for image correspondence, camera calibration, image registration, depth-information extraction and so on [89] to obtain some auxiliary data or side information in assistance to the following image dataset compression, image sequence rendering and virtual environment navigation.

Image correspondence across two or more images with different viewpoints plays an important role in the image analysis process. Most practical image correspondence techniques belong to either region-based solutions or feature-based solutions. Region-based solutions can be applied to all image samples and directly generate a dense disparity map. However, they tend to break down where there is lack of texture or depth discontinuity occurs. Feature-based solutions provide more precise and reliable matching, but establish the correspondence only between sparse sets of image features and increase the computational cost due to the extra computations for feature definition and detection. Hybrid image correspondence techniques combining region-based and feature-based techniques can also be applied for image analysis.

Some auxiliary information can be obtained through image analysis. The scene depth distribution can be derived from the disparity map resulting from image correspondence [30]. With camera calibration, the intrinsic and extrinsic camera parameters, such as the focal length, camera distortion parameters, effective pixel sizes on the image plane, camera positions and orientations, can be estimated based on image information and projective geometry. Some geo-reference information can be obtained either in the image acquisition process or alternatively by image registration.

2.1.5 Image Dataset Compression

In the image dataset compression process, a specially designed coding scheme is utilized to compress the image dataset into an efficient bit-stream representation and accommodate it to the allocated storage space. Some specific requirements for compressing image datasets need to be satisfied in the design of the compression scheme. A mechanism enabling proper random access and selective decoding should be provided in the scheme for image dataset compression. Low-complexity compression schemes or asymmetric compression schemes with low decoding complexity are needed for fast image extraction and decoding. Besides the requirement for compression efficiency, high quality of coded basis images is necessary because the basis images will be used as input for synthesizing novel-view images. Scalable coding is another desirable feature for image dataset compression. In particular, spatial-resolution scalability is needed for synthesizing uniform-quality zooming views.

Some compression techniques can be utilized for compressing image datasets. Vector quantization (VQ) techniques [53, 86] are capable to provide the required random access and selective decoding due to their asymmetric structure and constant-length index-table coding features. Modified standardized coding techniques [42, 75] can be applied to make use of the state-of-the-art standard coding approaches while adapting them to the features of image dataset compression. Also, wavelet-based compression techniques [23, 55] can be used to take advantage of the high-performance scalable coding provided by discrete wavelet transforms. Moreover, advanced video coding techniques [41, 93] and lifted motion-compensated wavelet transform techniques [22, 64] are potentially applicable with further research to be conducted.

2.1.6 Image Sequence Rendering

Image sequence rendering synthesizes novel-view images by taking primitive inputs from compressed image datasets and generates rendered image sequences based on IBR techniques. IBR uses basis images of image datasets by means of re-sampling, mosaicking, segmentation, or more generally mapping, re-projection, interpolation, or more specifically warping, morphing, transformation and so on to generate rendered image sequences with virtual viewpoints [79, 80, 104].

Some IBR techniques need some environmental geometric data to synthesize novel-view images. Different amounts of geometric data are required for different image rendering techniques. The geometric data could be utilized in an implicit or explicit way, and either accurate or approximate geometric data can be required depending upon the IBR technique in use. There is a tendency to apply IBR without using geometric data to avoid the difficulties and troubles with geometric data acquisition and processing. In such cases, more basis images with greater cross-image coherence are needed in the image dataset to avoid aliasing.

IBR follows users' instructions to produce the rendered image sequences. It is required that IBR provides users with rendered images at a certain quality level, generates smooth view transitions and synthesizes novel views at virtual viewpoints in the authorized walk-through regions for virtual environment navigation. Practical IBR techniques are developed by restraining the viewing space in order to simplify the image dataset representation [2]. Promising IBR techniques need to be developed to reduce the aliasing and artifacts in the rendered image sequences and fully satisfy the requirements for providing effective and natural navigation experiences.

2.1.7 Image Sequence Transmission

In the image sequence transmission process, combined source and channel coding is performed to efficiently compress the rendered image sequences, multiplex them with augmented multimedia contents, adapt the multiplexed source codes to the communication channels and deliver the output bit-streams via networks to remote users.

Standard video coding schemes can well satisfy the requirements for compressing the rendered image sequences. ITU-T Rec. H.263 [40], ISO/IEC MPEG [36, 37] and H.264/AVC [41] are available video coding standards for this purpose. These standards are capable of providing high-performance video coding with different compression efficiency. Motion-compensated prediction is a key technique used in video coding standards to achieve high coding performance. In the rendered image sequences, the image motion is determined by the motion of the virtual camera in the case of static environment representations. This image motion reveals some special features related to the pan, tilt and zoom motions of the virtual camera. These motion features can be utilized to facilitate efficient and fast image motion estimation and compensation and effectively improve the compression efficiency. Adopting a standard video coding scheme can also benefit from the compatibility the standard provides with other third-party developed software or hardware.

A transmission bit-stream is produced to adapt to the features of the communication channels. The streaming outputs of rendered image sequences, the accompanying auxiliary information and the augmented multimedia contents are combined in the transmission multiplexer. Packet techniques are utilized to map the multiplexed source data to the transmission layer. Error resilience and robustness tools, e.g., resynchronization, data partitioning and header extension codes, can be applied for error-prone communications like streaming over Internet and wireless mobile communications. A transmission buffer is designed to regulate the output bit-stream rate to fit the bandwidth limitation of the communication channels.

2.1.8 Virtual Environment Navigation

Navigating in the virtual environments, users are able to interactively extract and observe the synthesized image sequences that imitate video sequences captured by a virtual camera as it moves with the desired orientations along the navigation trajectories controlled by users according to their exploration intents. Users are allowed to freely wander in the virtual replica of the real-world environment by choosing their own paths and focusing on the views they are interested in.

Navigation in image-based virtual environments is provided to users by the system through user-friendly interfaces. Users indicate their exploration intents by using, e.g., a mouse, a joystick or a keyboard. Their instructions are sent back to the remote central computer, and the image rendering component synthesizes image sequences according to these instructions. After that, the synthesized streaming image sequences are transmitted to users, decoded and displayed at the user end. Users are aware of their locations and poses in the virtual environments with the assistance of the position and orientation instruction maps, and then give instructions about where they would like to move next to explore the virtual environments. This interactive working mode is maintained in the whole process of virtual environment navigation. A screen shot of the Navire viewer [6] with the orientation-location instruction map used in a developed image-based virtual environment navigation system is shown in Fig. 2.2.

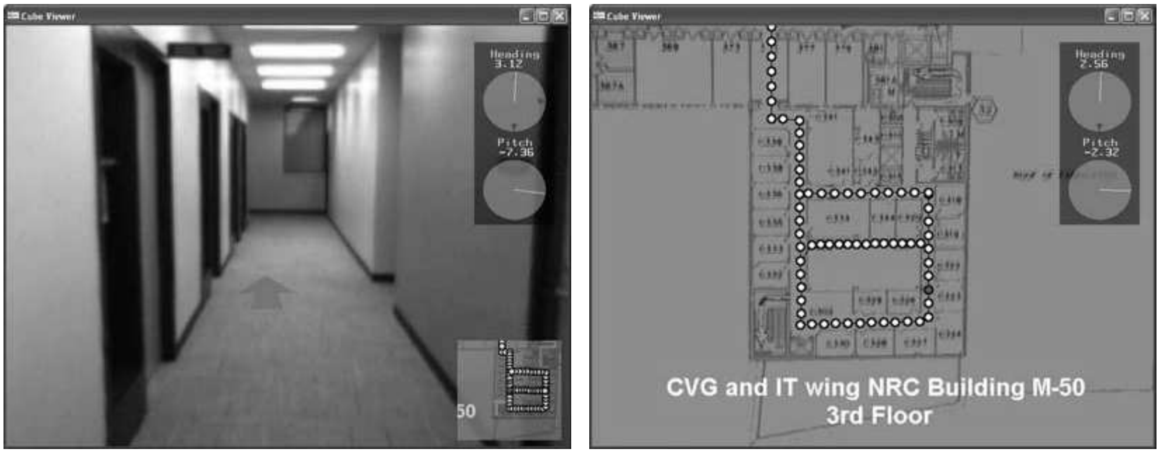


Fig. 2.2 A screen shot of the Navire viewer: (left) a user's view of the virtual environment and (right) the associated orientation-location map.

Besides the necessary position and orientation instruction maps, some other informative auxiliary multimedia contents can be added to augment scene understanding. Moreover, synthesized stereoscopic image pairs could be provided and displayed to enrich the visual information. All these measures are helpful to enhancing the realistic and immersive exploration experiences in the virtual environments.

2.2 Image-Based Rendering Techniques

As a primary component in the image-based virtual environment navigation system, image sequence rendering based on IBR is very closely related to image dataset compression, because not only does IBR generate synthesized images relying on the compressed image datasets, but also the design of image dataset compression schemes needs to meet the requirements of IBR. Moreover, the decoder of image datasets is located within the image sequence rendering component instead of the image dataset compression component. Therefore, this section is devoted to a review of IBR techniques: IBR and its mathematical representation are introduced; the relationships between IBR and image dataset compression are clarified; the classification for IBR techniques is presented, followed by brief descriptions of IBR techniques with and without geometric data.

2.2.1 Image-Based Rendering

IBR has significant advantages over the conventional geometry-based image rendering, which had dominated in image rendering for many years. Conventional image rendering relies on geometric model building to synthesize novel views. Building the models to represent the environments is a complicated and computationally-expensive process based on the geometric information of the environments. Depending on the complexity of the environments, the model building process could involve the construction of as many as millions of geometric models. In order to increase the sense of reality, some other features, such as the photometric properties of lighting and shading, of the environments should be incorporated during the model building process. This makes the geometry-based image rendering more complicated and expensive. Synthesizing a novel view from numerous geometric models is so time-consuming that a tradeoff has to be made to balance the rendered image quality and the rendering speed. It is difficult to achieve photo-realistic quality of rendered images even with the state-of-the-art geometry-based image rendering techniques. In contrast, IBR directly takes image inputs from the image datasets to synthesize novel views. Instead of complicated model-based representations of the environments, IBR just relies on the basis images in the image datasets to synthesize high quality, photo-realistic novel images. The rendering speed of IBR is high and independent of the complexity of the environments. IBR proposes an effective way to represent the real-word environments and provides a promising approach to generate photo-realistic novel-view images.

Mathematically, IBR techniques can be investigated by making use of the plenoptic function [2] representing the intensity distribution of light rays in a three dimensional space. In its most general form, the plenoptic function is a seven-dimensional function which can be expressed as

$$\mathcal{L}_P = f_P(P_x, P_y, P_z, \theta, \phi, \lambda, t) \tag{2.1}$$

representing the light ray intensity $\mathcal{L}_P$ of any wavelength λ, at any position (P_x, P_y, P_z), towards any direction (θ, ϕ) and at any time t in a three-dimensional environment space (Fig. 2.3). An IBR technique can be considered as a method for determining the values of the plenoptic function corresponding to required orientations and viewpoints.

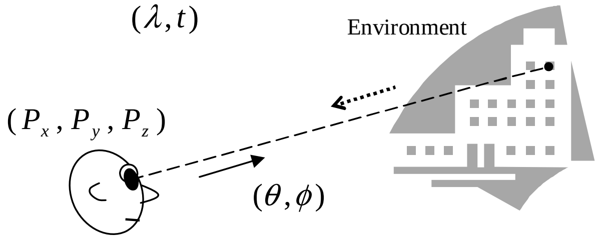


Fig. 2.3 Generalized seven-dimensional plenoptic function representation of light ray distribution in a three-dimensional space.

2.2.2 IBR and Image Dataset Compression

Image dataset compression and image sequence rendering based on IBR are closely related to each other. The interaction between these two components in an image-based virtual environment navigation system is illustrated in Fig. 2.4. The streaming output of compressed basis images from image dataset compression provides the primitive input for IBR to synthesize image sequences; the scheme of image dataset compression is designed to satisfy the requirements of IBR; and the compressed image dataset is decoded at the front end of the IBR process.

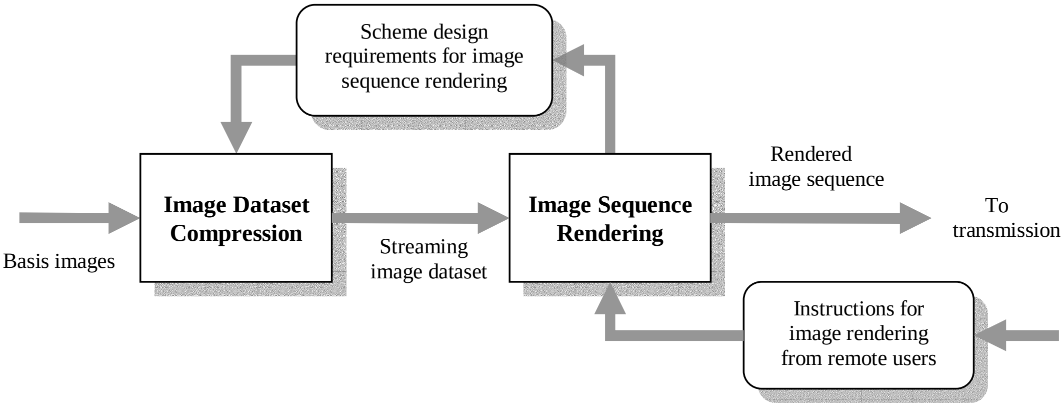

Scheme design requirements for image sequence rendering
Basis images
Image Dataset Compression
Streaming image dataset
Image Sequence Rendering
Rendered image sequence
To transmission
Instructions for image rendering from remote users

Fig. 2.4 Interaction between image dataset compression and image sequence rendering based on IBR in an image-based virtual environment navigation system.

Generally, an IBR technique is developed with respect to a special basic format of image dataset. Other formats of image datasets are useable with the IBR technique only if they are convertible to this basic image format. For image-based virtual environment navigation, an IBR technique with its associated basic image dataset format is chosen to be applied based on whether it can facilitate the navigation in the authorized virtual environment regions with the desired degrees-of-freedom of navigation. Also, the image acquisition process is performed for sampling the environment to obtain image datasets conforming to this basic format.

IBR synthesizes novel-view images by taking basis images as the primitive input directly from the compressed streaming image dataset. Different formats of image datasets lead to different compression scheme designs, as image dataset compression needs to adapt to the format of image datasets to achieve the target coding performance. Therefore, an IBR technique is associated with a basic image dataset format, which is closely related to the specially designed image dataset compression scheme. That is why there always is a discussion on the special issue of image dataset compression accompanying a newly developed IBR technique.

Image dataset compression needs to meet the requirements of IBR in order to obtain the desired system performance. IBR requires efficient access to the compressed streaming image datasets and fast decoding. The complexity of the compression scheme depends on how IBR requirements are satisfied. This is crucial especially in interactive applications like image-based virtual environment navigation. How the compression scheme is designed to efficiently encode the image dataset and accommodate a high system performance depends, to a great extent, on the IBR requirements. From this point of view, a specially designed image dataset compression scheme corresponds to a specific IBR technique.

2.2.3 Classification of IBR Techniques

IBR techniques can be grouped into different categories to facilitate convenient descriptions and discussions. Different strategies can be applied for classifying IBR techniques. Primarily according to the nature of the scheme for pixel indexing or transfer, Kang [46] classifies IBR techniques into four categories: non-physically based image mapping, mosaicking, interpolation from dense samples and geometrically-valid pixel re-projection. Based on how much geometric information is involved for image rendering, Shum et al. [76] put IBR techniques into three categories. They are IBR with implicit geometry, IBR with explicit geometry and IBR without geometry. Zhang and Chen [102] categorize IBR techniques into IBR restraining the viewing space and IBR introducing source description, according to the ways the plenoptic function representations for IBR are simplified from the original 7D format representation. Although different classifications for IBR techniques are available, sometimes it is still found difficult to make a clear decision on whether to put an IBR technique into one category or another. There exist some IBR techniques that do not belong to a single category but fall somewhere between two categories.

In this thesis, a simplified classification of IBR techniques is made from the image dataset compression point of view. According to whether or not the environment geometric data is used in the process of synthesizing novel-view images, IBR techniques are divided into two categories: IBR without geometric data and IBR with geometric data. The former synthesizes novel-view images just relying on basis images in image datasets. The latter, on the other hand, synthesizes novel-view images with the help of certain geometric data besides taking basis images as its primitive inputs. For image dataset compression, extra coding techniques are needed to compress the associated geometric data in the case of IBR with geometric data.

2.2.4 IBR with Geometric Data

Some IBR techniques are developed with the requirement for using environmental geometric data to assist image rendering. Either accurate or approximate geometric data could be requested, and the geometric data could be utilized in an implicit way or an explicit way, depending on the particular IBR technique in use. Some IBR techniques synthesize novel views with the assistance of disparity maps obtained from image matching or image correspondence. This is considered as rendering with implicit geometric data because of the relation between the disparity map and the depth distribution of the environment. Some representative IBR techniques that are classified as IBR with geometric data are Lumigraph [24], view morphing [71], layered depth images (*LDI*) [72], surface light field [94], image-based visual hulls [60], etc.

In the image dataset compression process, additional considerations need to be given to properly compress the geometric data coming with basis images. It can be coded separately by some other compression methods such as sequential data compression [105], or it can be integrated into the image dataset bit-stream for efficiently compressing the geometric data to achieve satisfied overall compression performance.

2.2.5 IBR without Geometric Data

Obtaining environmental geometric data during the image acquisition process makes the data acquisition system and procedure more complicated and expensive. Although some developed techniques are available to get the geometric data directly from image datasets instead of obtaining it in the image acquisition process, it is difficult to get all required geometric data with desired accuracy even with the state-of-the-art image analysis techniques. In order to avoid the difficulties and troubles with geometric data acquisition and processing, there is a tendency to make more use of IBR without geometric data in place of IBR with geometric data at the cost of more intensive image acquisition. Under this circumstance, more basis images with greater cross-image coherence are obtained in the image datasets to avoid aliasing. Some representative IBR techniques that are classified as IBR without geometric data are listed in Table 2.1.

A panorama can be regarded as an orientation-independent map of the environment represented by a simplified 2D plenoptic function. It corresponds to a projection of the environment onto a geometrical omni-directional surface. Depending on the shape of the surfaces, panoramas are divided into three different main categories of formats (Fig. 2.5). They are cylindrical panoramas [10], spherical panoramas [82] and cubic panoramas [25]. By using a panoramic image, the reconstruction of a new image viewed at the fixed center point in a desired direction refers to a process of image warping or

IBR	Dim.[a]	Observe Space	Rendering Method	Ref.[b]
Panoramas	2D	fixed point	image warping and re-projection	[10] [25] [82]
Concentric mosaics	3D	circular area	interpolation of image slits based on basis image columns	[74]
Light field rendering	4D	rectangular area	re-sampling and assembling the basis images	[53]
Plenoptic stitching	4D	unobstructed area	stitching image columns derived from recoded omni-directional images	[3]

[a] Dimensionality
[b] Reference

Table 2.1 Representative IBR techniques without geometric data

re-projection. With a single panoramic image, the viewing space is limited to one fixed point that is the geometrical center of the panorama. The viewing direction can be changed with a pan or tilt motion at this fixed viewpoint, and zooming-in and zooming-out can be facilitated by making use of the panoramic image. If the viewing space needs to be enlarged, a collection of panoramas along pre-designed trajectories can be generated through the use of special omni-directional cameras or constructed from a captured regular-format image collection. In this case, users are able to switch between the panorama sample nodes for navigating in an extended space [10].

In this thesis, besides the discussions on the generic issues of image dataset compression, cubic-panorama image datasets are chosen as a concrete form of image dataset for the purposes of intensive investigation, implementation and experiments of image dataset compression schemes due to the unique advantages of cubic panoramas over

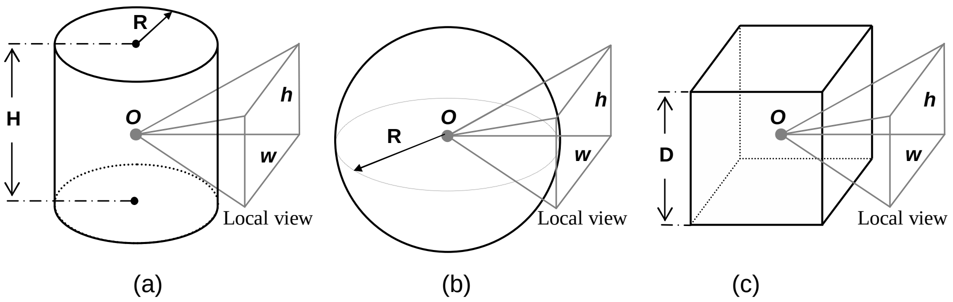


Fig. 2.5 Three main categories of panoramas: (a) the cylindrical panorama, (b) the spherical panorama and (c) the cubic panorama.

other environmental mapping methods [25].

Chapter 3

Image Dataset Compression for Image-Based Virtual Environment Navigation

Compression of image datasets used for image-based virtual environment navigation is indispensable to reduce the required storage space and adapt to the limited transmission bandwidth. It constitutes a fundamental component in an image-based virtual environment navigation system. This chapter focuses on the discussions about image dataset compression for image-based virtual environment navigation.

First, the problem of compressing image datasets used for image-based virtual environment navigation is presented in Section 3.1. In Section 3.2, the characteristics of image dataset compression are investigated, including the features of the image datasets, the requirements for image dataset compression, the key issues in compressing image datasets and the strategies for the design of image dataset compression schemes. In Section 3.3, research progresses made so far in the image and video coding research community in developing various coding techniques and schemes for compressing image datasets are summarized in a comprehensive literature review. Different categories of current solutions for image dataset compression are compared in Section 3.4. Finally, the discussion on image dataset compression used for image-based virtual environment navigation is summarized in Section 3.5.

3.1 Problem Description

Previously in Section 2.2.1, a plenoptic function $\mathcal{L}_P$ was introduced to represent the intensity distribution of light rays in a real-world environment. In order to obtain the image dataset input for image-based virtual environment navigation, the camera imaging system $\mathcal{H}_{Cam}$ samples this light ray intensity signal $\mathcal{L}_P$ on the real-world environment domain $\bar{\mathbf{D}}_{\mathcal{E}}$, on which $\mathcal{L}_P$ is defined as shown in Fig. 2.3, and maps the sampled signal into a sequence of raw images $\mathcal{I}_{\mathcal{R},\,l}$ $(l\in\{1,2,...,L\},\ L\in\mathbb{Z})$ on a defined raw image domain $\bar{\mathbf{D}}_{\mathcal{R}}$. Thus, we have

$$\mathcal{H}_{Cam}:\ \mathcal{L}_P \longmapsto \mathcal{I}_{\mathcal{R},\,l},\ \forall\, l\in\{1,2,...,L\}. \tag{3.1}$$

Let $\mathcal{I}_{\mathcal{B},\,k}$ $(\,k\in\{1,2,...,K\},\ K\in\mathbb{Z})$ represent a basis image in a specific image format on a well-defined basis image domain $\bar{\mathbf{D}}_{\mathcal{B}}$, which could be, for example, a cubic surface as shown in Fig. 1.2. Following the raw image acquisition, the preprocessing component $\mathcal{H}_{Pre}$ of the navigation system derives the basis images from the captured raw images by, for example, an image projection, image mapping or image transformation operation

$$\mathcal{H}_{Pre}:\ \mathcal{I}_{\mathcal{R},l},\longmapsto \mathcal{I}_{\mathcal{B},\,k},\ \forall\, l\in\{1,2,...,L\},\ \forall\, k\in\{1,2,...,K\}. \tag{3.2}$$

Thus, an image dataset is obtained as a collection of a set of K basis images

$$\tilde{\mathbb{S}}_I=\big\{\big(\mathcal{I}_{\mathcal{B},1}(\bar{\mathbf{x}}),\ \mathcal{I}_{\mathcal{B},2}(\bar{\mathbf{x}}),\ ...,\ \mathcal{I}_{\mathcal{B},\,K}(\bar{\mathbf{x}})\big)\big|\ {}_{\bar{\mathbf{x}}\in\bar{\mathbf{D}}_{\mathcal{B}},\,K\in\mathbb{Z}}\big\}. \tag{3.3}$$

The image dataset $\tilde{\mathbb{S}}_I$ used for image-based virtual environment navigation typically consists of densely generated basis images $\mathcal{I}_{\mathcal{B},\,k}(\bar{\mathbf{x}})$ and involves a large amount of data. It occupies huge storage space if directly kept in storage media and requires high transmission bandwidths if directly sent through communication networks. Usually high-resolution basis images corresponding to a dense $\bar{\mathbf{D}}_{\mathcal{B}}$ are needed for novel-view image rendering. $\tilde{\mathbb{S}}_I$ for large environment representations with large authorized navigation spaces is needed in more and more applications. This requires a large number K of $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$. Also, there is a tendency of developing IBR techniques for using more basis images in place of geometric information to take full advantages of image-based novel-view rendering over model-based image rendering and avoid the troubles with geometry data acquisition and processing. All these result in even larger sizes of $\tilde{\mathbb{S}}_I$ with larger number K of $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ defined on a denser $\bar{\mathbf{D}}_{\mathcal{B}}$. This poses a critical problem for storage and transmission. Therefore, compression of the image datasets is crucial to fit them to the storage allocation, adapt them to the bandwidth limitation and make image-based virtual environment navigation practical.

A lot of intensive research work on image and video sequence compression has been conducted for many years. Various compression techniques and schemes are well developed and widely applied. Some of the state-of-the-art techniques and schemes have found their way to a variety of international image and video sequence coding standards, ranging from ISO/IEC JPEG [33] and JPEG2000 [34] for still image coding, ITU-T H.261 [39] and H.263 [40] as well as ISO/IEC MPEG-1 [35], MPEG-2 [36] and MPEG-4 [37] for video sequence coding to the recent H.264 (also named MPEG-4 AVC) [41] jointly developed by ITU-T and ISO/IEC for advanced video coding. However, these compression techniques and schemes are designed for generic image and video sequence compression, and are sub-optimal if directly applied for compressing the image datasets used for image-based virtual environment navigation. Compression of the image datasets is significantly different from the conventional problem of generic image and video sequence compression. Making the compression schemes adaptive to the specific features of the image datasets is very important to obtain the desired coding performance. Meeting the special requirements for compressing the image datasets is the key to achieving expected interactive exploration experiences in image-based virtual environment navigation. Compression techniques and schemes specially designed for compressing image datasets need to be developed.

3.2 Characteristics of Image Dataset Compression

In this section, the features of the image datasets used for image-based virtual environment navigation are described especially from the viewpoint of image dataset compression. The requirements for compressing image datasets are discussed with the emphasis on providing fast interactive image sequence rendering. Also, the challenges confronted in compressing image datasets to achieve the desired compression performance are presented, and the principal strategies of designing the coding schemes for effective and efficient image dataset compression are investigated.

3.2.1 Features of Image Datasets

Generally, image datasets used for image-base virtual environment navigation can mathematically be represented by samples of simplified plenoptic functions. Image dataset compression techniques can be developed by studying the mathematical representations of these plenoptic functions. Besides, the dominant cross-image displacement in basis images demonstrates a certain regular pattern for each type of image dataset. This cross-image displacement feature, depending on the motion pattern of the camera or cameras capturing the raw image sequences, could be utilized to improve the compression performance. Moreover, the statistical features of image datasets can be estimated and employed for compression purposes. Some coding parameters can be chosen adaptively for image datasets according to their unique statistical features.

Typically, an image dataset consists of numerous high-resolution basis images. There are tremendous local and cross-image redundancies among these basis images. High compression ratios could potentially be achieved in compressing image datasets. In addition, some types of image datasets are accompanied by certain kinds of side information, which can be applied for efficiently exploiting coherency in the image datasets. Some detailed discussions about the features of image datasets used for image-based virtual environment navigation are given in this section.

(1) Image datasets can be represented mathematically by samples of simplified plenoptic functions.

In its most general form, the seven-dimensional plenoptic function $\mathcal{L}_P$ in (2.1) represents the intensity distribution of light rays in a three dimensional environment space. This corresponds to a complicated high-dimensional multi-variable data structure, which makes image dataset compression more difficult to deal with. In order to simplify the expression of $\mathcal{L}_P$, the wavelength variable λ can be removed from the plenoptic function by assuming that each wavelength component or a combination of light rays at a certain number of wavelengths (e.g. a color component) is separately investigated

$$\mathcal{L}_P \rightarrow \mathcal{L}_{P6} = f_{P6}(P_x, P_y, P_z, \theta, \phi, t). \tag{3.4}$$

Moreover, the time variable t can be removed from the expression in the case of static environments under discussion. As a result, only five independent variables remain in the plenoptic function expression

$$\mathcal{L}_P \rightarrow \mathcal{L}_{P5} = f_{P5}(P_x, P_y, P_z, \theta, \phi). \tag{3.5}$$

The dimensionality $\mathbf{L}_d$ of the plenoptic function $\mathcal{L}_P$ can be further reduced by constraining the viewing space to certain bounded areas just as in the previously mentioned IBR techniques in Table 2.1. Under these circumstances, the basis image $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ in $\tilde{\mathbb{S}}_I$ can be regarded as observed discrete samples of the simplified $\mathcal{L}_P$. The IBR process becomes a procedure of determining the values of the simplified $\mathcal{L}_P$ associated with the required viewpoints, based on the image samples $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$. Some compression techniques for $\tilde{\mathbb{S}}_I$ can be investigated by studying the simplified representations of $\mathcal{L}_P$. The data amount of the image datasets can be significantly reduced by lowering the dimensionality $\mathbf{L}_d$ of the plenoptic function representations. Generally, this is the first consideration taken as an effort to reduce the image dataset sizes. However, the authorized viewing space will accordingly be limited. Therefore, it is required to simplify the plenoptic function while keeping the desired viewing space.

(2) Image datasets typically consist of a large number of high-resolution basis images.

By sampling the environments to obtain the image datasets, certain high sampling rates are required to avoid aliasing. A large number K of $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ are obtained in the image datasets to completely cover the represented environments and to be able to synthesize any novel views at arbitrary viewpoints in the authorized navigation spaces. The image dataset size is further increased in the case of representing a large environment with a large navigation space and enabling good-quality zoom-in image rendering based on high-resolution basis images. Representations of large environments result in large number K and high-resolution basis images correspond to dense $\bar{\mathbf{D}}_{\mathcal{B}}$. The increased data amount of image datasets can be regarded as the expense incurred to avoid the troubles of model building and provide photo-realistic rendering. To some extent, the number K of $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ in an image dataset $\tilde{\mathbb{S}}_I$ can be controlled by properly applying the plenoptic sampling theory [8, 54] and spectral analysis [101], but even with the minimum number of basis images based on sampling theories, the typical image datasets are still large in number K of the basis images and involve a huge amount of data. Compression of the image datasets is unavoidable to reduce the image dataset size and make image-based virtual environment navigation available in practical applications.

(3) Image datasets involve high local and cross-image redundancies.

Taking basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ of $\tilde{\mathbb{S}}_I$ as the samples of simplified plenoptic functions, there exist enormous redundancies in all $\mathbf{L}_d$ dimensions of the plenoptic functions. For example, the image datasets of plenoptic stitching [3] and light field rendering [53] exhibit redundancies in four dimensions; the image datasets of concentric mosaics [74] exhibit redundancies in three dimensions; and the image datasets of cylindrical, spherical and cubic panoramas [10, 25, 82] in two dimensions.

From another point of view, all the basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ in an image dataset $\tilde{\mathbb{S}}_I$ for image-based virtual environment navigation are image samples of the same environment obtained from different perspectives, and a large number of raw images $\mathcal{I}_{\mathcal{R},l}$ $(\forall\, l \in \{1, 2, ..., L\})$ are taken to avoid aliasing. This leads to tremendous cross-image redundancy existing in the image datasets. Due to these features, high compression ratios can be expected in compressing the image datasets. The higher the dimensionality $\mathbf{L}_d$ of the plenoptic function representation $\mathcal{L}_P$ is, the more redundancy exists in the corresponding image dataset $\tilde{\mathbb{S}}_I$ and the higher compression efficiency the compression schemes can potentially achieve.

(4) The dominant cross-image displacement of the image datasets demonstrate a certain regular pattern for each type of image dataset.

The image motion in a generic video sequence can be caused by the movements of the objects captured in the images, by moving the camera for capturing the video sequence and even by the background motion, leading to a complicated motion vector distribution in the motion fields. Suppose that static environments are under discussion. Then, in the image dataset $\tilde{\mathbb{S}}_I$ used for image-based virtual environment navigation, the cross-image displacement is caused only by the motion of the camera producing the basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$. Although there are some factors, such as the depth distribution $\mathcal{D}_k(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ of the environment, generating some local differences in the displacement vector distribution map, the dominant cross-image displacement of an image dataset $\tilde{\mathbb{S}}_I$ typically follows a certain pattern for the given *IBR* technique, corresponding to the motion pattern of the camera producing $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$. In order to cover the whole environment, the trajectory of the camera movement is well planned based on the associated plenoptic sampling theory for producing the image dataset. The orientation of the camera and the trajectory of the camera movement together decide the camera motion pattern and in turn determine the cross-image displacement pattern. This cross-image displacement feature of the image dataset $\tilde{\mathbb{S}}_I$ could be utilized to significantly improve the compression performance.

(5) The statistical features of image datasets can be obtained and employed for compression purpose.

For generic video sequence coding, the content of the next image to be encoded cannot be precisely predicted. In other words, the exact image contents are randomly generated, and generally, their statistical features are not available before encoding the video sequence. In contrast, for the image dataset $\tilde{\mathbb{S}}_I$ used for image-based virtual environment navigation, the image contents of the basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ are known and unchanged once they are produced. One part or even all of an image dataset $\tilde{\mathbb{S}}_I$ can be taken as a training image set. The histogram and other statistical information of the image dataset can be obtained. These statistical features of the image dataset could be employed to enhance the compression efficiency. Some coding parameters can be chosen adaptively for each image dataset $\tilde{\mathbb{S}}_I$ or even each basis image $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(k \in \{1, 2, ..., K\})$ according to its unique statistical features. The image dataset $\tilde{\mathbb{S}}_I$ can even be repeatedly encoded or a recursive encoding process can be taken to finally obtain an optimized bit-stream output of the encoded image dataset.

(6) Some image datasets are accompanied by certain kinds of side information.

Some image datasets are accompanied by certain kinds of side information, such as the environmental depth distributions $\mathcal{D}_k(\bar{\mathbf{x}})$, camera poses (θ_k, ϕ_k) and camera locations $(P_{x,k}, P_{y,k}, P_{z,k})$ with $\forall\, k \in \{1, 2, ..., K\}$. IBR techniques with geometric data are associated with image datasets with side information. Such side information is obtained in the image acquisition process or results from the image analysis process, and is useful for simplifying the image acquisition and the image-based representation of the environment, or in assistance of environmental model building, novel-view image rendering or image dataset compression. The sampling density in capturing the raw image sequences can be reduced due to the use of such side information. For the purpose of image dataset compression, side information can be applied for exploiting image coherency and efficiently compressing the image datasets. However, extra considerations and coding techniques are needed for compressing the side information.

3.2.2 Requirements for Image Dataset Compression

Compression of image datasets used for image-based virtual environment navigation needs to meet the requirements of the navigation system. Most of these requirements come from the IBR-based image sequence rendering component, which constitutes a primary part of the system. Based on the decoded basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(\forall\, k \in \{1, 2, ..., K\})$ in an image dataset $\tilde{\mathbb{S}}_I$, IBR synthesizes, at the desired viewpoints, a sequence of novel-view images $\mathcal{I}_{\mathcal{S},m}(\hat{\mathbf{x}})$, $\forall\, m \in \{1, 2, ..., M\}$, $M \in \mathbb{Z}$. Here $\mathcal{I}_{\mathcal{S},m}(\hat{\mathbf{x}})$ denotes a synthesized planar image with a generic perspective view defined on a two-dimensional image domain $\bar{\mathbf{D}}_{\mathcal{S}}$ and $\hat{\mathbf{x}}$ is a vector in $\bar{\mathbf{D}}_{\mathcal{S}}$: $\hat{\mathbf{x}} \in \bar{\mathbf{D}}_{\mathcal{S}}$. Some special requirements for compressing image datasets need to be satisfied for efficient interactive image rendering.

Random access and selective decoding are unique requirements for image dataset compression. A mechanism enabling flexible random access to, and selective decoding of, small relevant parts necessary to reconstruct the required image segments in the coded image dataset should be provided in the design of the image dataset compression scheme. Low-complexity compression schemes with low computational expenses or asymmetric compression schemes with relatively high encoding complexity but low decoding complexity are needed for fast image decoding. Besides the requirement for compression efficiency, the quality of coded basis images needs to be guaranteed as the basis images will be used as input to synthesize novel-view images at the required quality level. In certain cases, even near lossless coding could be used to keep the coded basis images in high quality. Scalable coding is another extremely desirable feature for image dataset compression. In particular, spatial-resolution scalability is required for uniform high-quality zooming. Even the design of a baseline image dataset compression system should incorporate spatially-scalable coding.

(A) Random access to the coded image datasets:

A basis image $\mathcal{I}_{\mathcal{B},\,k}(\bar{\mathbf{x}})\ \ (k \in \{1, 2, ..., K\})$ defined on the domain $\bar{\mathbf{D}}_{\mathcal{B}}$ can be divided into image segments $\mathcal{I}_{\mathcal{B},\,k}(\bar{\mathbf{x}}) = \{\mathcal{I}_{\mathcal{B},\,k,\,j}(\bar{\mathbf{x}})|\ _{\forall j \in \{1,2,...,J\},\ J \in \mathbb{Z}}\}$. Each image segment $\mathcal{I}_{\mathcal{B},\,k,\,j}(\bar{\mathbf{x}})$ is defined on a sub-domain $\bar{\mathbf{D}}_{\mathcal{B},\,j}$ with $\bar{\mathbf{D}}_{\mathcal{B},\,j} \subset \bar{\mathbf{D}}_{\mathcal{B}}$ and $\bar{\mathbf{D}}_{\mathcal{B}} = \cup_{\forall\, j \in \{1,2,...,J\}} \bar{\mathbf{D}}_{\mathcal{B},\,j}$. IBR synthesizes a novel-view image at a desired viewpoint by taking image segments $\mathcal{I}_{\mathcal{B},\,k,\,j}(\bar{\mathbf{x}})$ from a set of basis images in $\tilde{\mathbb{S}}_I$. The choice of the viewpoints and thus the corresponding synthesized images are interactively controlled by the users of an image-based virtual environment navigation system. This requires a mechanism enabled in the design of the image dataset compression scheme to provide efficient random access to the required image segments $\mathcal{I}_{\mathcal{B},\,k,\,j}(\bar{\mathbf{x}})$, $\bar{\mathbf{x}} \in \bar{\mathbf{D}}_{\mathcal{B},\,j}$ in the coded image dataset $\tilde{\mathbb{S}}_I$. This random access mechanism is very important in facilitating an efficient image rendering process. The conventional still image coding method limits the random access to the beginning of a whole coded image, and the conventional video sequence coding limits it to intra-coded frames or the beginning of a group of images at the image frame level. In compressing the image datasets used for image-based virtual environment navigation, a much more flexible random access mechanism is required to save memory and speed up the image decoding and rendering processes. The random access to the coded image datasets could be required at image column ($\bar{\mathbf{D}}_{\mathcal{B},\,j}$ occupies a strip region in $\bar{\mathbf{D}}_{\mathcal{B}}$), image pixel ($\bar{\mathbf{D}}_{\mathcal{B},\,j}$ dwindles to a point)or other image segment levels, depending on the requirement of the IBR that follows the image dataset compression.

(B) Selective decoding of the coded image datasets:

For conventional video sequence decoding, if an image encoded as a predicted image needs to be reconstructed, an intra reference image preceding the predicted image should be decoded first before the predicted image becomes decodable. Therefore, at least two images have to be decoded although only one predicted image is required to be reconstructed. If the reconstructed image is encoded as a bi-directionally predicted image, altogether at least three images (one preceding and one following reference images and the bidirectional predicted image itself) have to be decoded. This slows down the decoding process and makes it much less efficient. Following the same coding strategy, even greater numbers of basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(k \in \{1, 2, ..., K\})$ in the image dataset $\tilde{\mathbb{S}}_I$ need to be decoded for rendering one synthesized image $\mathcal{I}_{\mathcal{S},m}(\hat{\mathbf{x}})$ $(m \in \{1, 2, ..., M\})$, as usually much more than a couple of $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ are used to synthesize a novel-view image $\mathcal{I}_{\mathcal{S},m}(\hat{\mathbf{x}})$ for most IBR techniques. For this reason, it is required for the image dataset compression that only the relevant image part on a domain $\bar{\mathbf{D}}'_{\mathcal{B},j}$ $(\bar{\mathbf{D}}_{\mathcal{B},j} \subset \bar{\mathbf{D}}'_{\mathcal{B},j})$ as small as possible needs to be selectively decoded in order to reconstruct a required image segment $\mathcal{I}_{\mathcal{B},k,j}(\bar{\mathbf{x}})$, $\bar{\mathbf{x}} \in \bar{\mathbf{D}}_{\mathcal{B},j}$ of a coded basis image $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$, $\bar{\mathbf{x}} \in \bar{\mathbf{D}}_{\mathcal{B}}$ in the image dataset $\tilde{\mathbb{S}}_I$ without having to decode other parts of $\tilde{\mathbb{S}}_I$. The ideal case is to make $\bar{\mathbf{D}}'_{\mathcal{B},j} = \bar{\mathbf{D}}_{\mathcal{B},j}$. Selective decoding is necessary for efficient and fast image decoding and image rendering.

(C) Low complexity with low computational expense:

The image dataset compression scheme prescribes the structure of the image dataset codec. The structural complexity of the codec determines the computational expense, the encoding and decoding speeds and the system delay. Fast basis image retrieval and interactive novel-view image rendering need to benefit from low structural complexity of the codec. Low-complexity with low computation expense is required for the design of the image dataset compression schemes for image-based virtual environment navigation, especially for some applications targeting real-time IBR without hardware assistance.

Compressing an image dataset $\tilde{\mathbb{S}}_I$ characterized by a plenoptic function representation $\mathcal{L}_P$ of high dimensionality $\mathbf{L}_d$ usually corresponds to a high-complexity codec structure. In this case, it is difficult to design a symmetric compression scheme with low complexity on both the encoding and decoding sides. However, compressing image datasets can be performed offline in advance. Therefore, an asymmetric compression scheme with relatively high encoding complexity but low decoding complexity can be employed for the purpose of fast image decoding and interactive image rendering.

(D) Fast decoding for fast image sequence rendering:

The interactive mode of image-based virtual environment navigation requires fast novel-view image synthesis, enabled through fast basis image decoding accommodated by the image dataset compression scheme. The decoding speed is affected by the random-access mechanism, the selective-decoding ability as well as the decoding complexity. For conventional video sequence compression, the decoding speed of a system is measured by the number of images the system decodes within a unit time (a second) or the average time the system spends to decode one image, and fast decoding requires that the average decoding time should not exceed a prescribed value, e.g., 0.03 or 0.04 second for real-time video sequence display. In comparison, if real-time image sequence display is required for the synthesized image sequences $\mathcal{I}_{\mathcal{S},m}(\hat{\mathbf{x}})$, $(\forall\, m \in \{1, 2, ..., M\},\ M \in \mathbb{Z})$, it means that the basis image decoding time and the interactive image sequence rendering time put together should not exceed the prescribed time limit. Thus, less time is left for the decoding process itself compared with conventional video decoding. So, fast decoding becomes a more challenging requirement in the design of image dataset compression schemes.

(E) High image quality for interactive virtual environment navigation:

The coded basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(k \in \{1, 2, ..., K\})$ in the image dataset $\tilde{\mathbb{S}}_I$ should be kept in uniform and high image quality as these images not only will be used as input images to synthesize novel-view images $\mathcal{I}_{\mathcal{S},m}(\hat{\mathbf{x}})$ $(m \in \{1, 2, ..., M\})$ but also may be displayed as still images for interactive virtual environment navigation. In conventional video sequence coding, bitrate control techniques are applied to adjust the coding parameters, generating coded images with different image qualities related to the image display order, and optimal video sequence display can be obtained despite the image quality differences in the video sequences. In image-based virtual environment navigation, the access to basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(k \in \{1, 2, ..., K\})$ and the generation of synthesized images $\mathcal{I}_{\mathcal{S},m}(\hat{\mathbf{x}})$ $(m \in \{1, 2, ..., M\})$ are randomly controlled by users. There is no fixed order for displaying them. Also, both the basis images and the synthesized images can be required to display as still images at the navigation stage. As the human vision system *(HVS)* generally has a smaller distortion tolerance to still images than time-variant image sequences, basis images and synthesized images should be of high quality. Because the synthesized images are rendered from the basis images, the quality of the coded basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(k \in \{1, 2, ..., K\})$ should be high enough to generate good-quality synthesized images. Sometimes, even near lossless coding is applied to keep the coded basis images in high image quality.

(F) Spatially scalable coding for zooming image rendering:

Spatially scalable coding is especially desirable for compressing image datasets used for image-based virtual environment navigation. Scalable coding with spatial, temporal and quality scalabilities is an effective solution to adapting the image and video sequence to different storage capacity, to distributing them via networks with different bandwidths and replaying them on different display devices. Thus, it is recommended in a variety of international image and video coding standards. However, scalable coding is not involved in commonly applied profiles of international coding standards, especially not in the baseline coding schemes, as currently the majority of the applications have no need to implement scalable coding. In contrast, when exploring an image-based virtual environment, the commonest user behavior next to panning and tilting with the virtual camera is to take a close shot or a long shot while keeping shot direction unchanged. In these cases, basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ $(k \in \{1, 2, ..., K\})$ defined on basis image sub-domains with different densities corresponding to different resolutions of basis images in the image dataset $\tilde{\mathbb{S}}_I$ are required for obtaining rendered zooming images in uniform good quality, and spatially scalable coding is needed to satisfy this requirement in the compression scheme design. Spatially scalable coding is more important for compressing image datasets used in image-based virtual environment navigation than in other applications. Even the design of a baseline image dataset compression system should incorporate spatially scalable coding.

(G) High efficiency of image dataset compression:

This is the essential objective and fundamental requirement for compressing image dataset $\tilde{\mathbb{S}}_I$ to make image-based virtual environment navigation available in practical applications. High-resolution basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}})$ with dense $\bar{\mathbf{D}}_{\mathcal{B}}$ and high-dimensional plenoptic function representation $\mathcal{L}_P$ with large number $\mathbf{L}_d$ of the image dataset $\tilde{\mathbb{S}}_I$ have the potential for achieving high compression efficiencies. The challenge is to obtain high compression efficiency and at the same time to satisfy other requirements for compressing image datasets.

3.2.3 Key Issues in Compressing Image Datasets

For image dataset compression used in image-based virtual environment navigation, we need to efficiently represent the basis images $\mathcal{I}_{\mathcal{B},k}(\bar{\mathbf{x}}) \quad (\forall\, k \in \{1, 2, ..., K\})$ in the streaming output by making use of the features of the image datasets, support random access to and selective decoding of the coded image segments $\mathcal{I}_{\mathcal{B},k,j}(\bar{\mathbf{x}}) \quad (\bar{\mathbf{x}} \in \bar{\mathbf{D}}_{\mathcal{B},j}, j \in \{1, 2, ..., J\})$ and satisfy other requirements for interactive image sequence rendering. In order to fulfill these needs, we confront a number of issues in developing image dataset compression techniques and schemes. Some of these issues pose considerable challenges to the work on an efficient scheme for compressing the image datasets.

Accommodating random access and selective decoding in an efficient coding scheme is a key and unique issue in compressing image datasets. Block-based hybrid coding of spatial transforms combined with cross-image prediction is widely applied in conventional video sequence compression schemes. However, with the spatial block transform, the whole block has to be decoded even though only one pixel in the block is needed for image rendering, and the cross-image prediction limits the flexibility of selective decoding to the beginning of an intra-coded reference image. Moreover, variable-length entropy coding and some other techniques, such as multiple-reference prediction and predicted images as reference images, prove to be effective to improve the coding efficiency, but they complicate the procedure for locating the required image segments in the bit-streams. The alternative wavelet-based coding expands the operation of removing spatial redundancy over the full frame, but makes random access and selective decoding more difficult. On the other hand, the compression techniques able to support random access and selective decoding usually provide only limited compression efficiency or add large overhead and thus sacrifice the compression performance. It is a challenge to achieve high compression efficiency while meeting the random access and selective decoding requirements.

Supporting spatial-resolution scalable coding is another key issue in compressing image datasets. It is extremely desirable for image dataset compression used for image-based virtual environment navigation, but not practical to be implemented in conventional block-based hybrid coding schemes due to the high cost of coding efficiency reduction. It is more practical to incorporate scalable coding into image dataset compression schemes by making use of discrete wavelet transforms based on the developments in the research on lifted discrete wavelet transforms with motion-compensated temporal filtering for video sequence compression. However, it still presents some difficulties to adapt this approach from generic video sequence coding to image dataset compression because a more complicated tradeoff among the spatial-resolution scalability, the coding efficiency and the random access and selective decoding flexibility needs to be made. In addition, high-dimension image dataset compression is more difficult to deal with than the conventional three-dimensional video sequence compression as the coherence in the extra dimensions needs to be exploited by proper techniques.

The interactive working mode of image-based virtual environment navigation requires fast image synthesis enabled through fast basis image decoding. This applies a more rigorous decoding speed constraint on the scheme design for image dataset compression compared to the conventional video sequence compression. The decoding speed of image datasets is related to the compression efficiency, the random-access flexibility, the spatial-resolution scalability and the decoding structure complexity as well. All these factors must be managed to facilitate just-in-time image decoding and rendering.

These issues present some critical problems in compressing the image datasets. Specially designed compression schemes should be developed to overcome these problems and achieve the coding objectives.

3.2.4 Strategies for Compressing Image Datasets

Generally, the first consideration for reducing the size of the image dataset $\tilde{\mathbb{S}}_I$ is to lower the dimensionality $\mathbf{L}_d$ of the plenoptic function $\mathcal{L}_P$. It was taken in almost all the image dataset compression schemes accompanying the existing practical IBR techniques, such as light field rendering [53] and concentric mosaics rendering [74]. Lowering the dimensionality $\mathbf{L}_d$ not only largely reduces the data amount of the image dataset $\tilde{\mathbb{S}}_I$, but also makes the raw image sequence acquisition, image dataset compression and image sequence rendering more manageable. However, the authorized viewing space $\mathbf{S}_V$ will accordingly be reduced together with lowering $\mathbf{L}_d$. Therefore, simplification of the plenoptic function $\mathcal{L}_P$ should be done while maintaining the desired viewing space. For this reason, we can divide the whole viewing space into several smaller subspaces $\mathbf{S}_V = \cup_{\forall n \in \{1,2,\ldots,N\}} \mathbf{S}_{V,n}$, $N \in \mathbb{Z}$. Then, the plenoptic function can be simplified separately for each smaller viewing sub-space $\mathbf{S}_{V,n}$.

Compression schemes for coding image datasets can be developed by making use of some existing efficient image and video sequence coding schemes. However, as the compression techniques in these coding schemes were originally designed for compressing generic images and video sequences, some modifications are necessary to adapt them to the special characteristics of the image datasets and obtain the required coding performance. Adapting compression techniques to the data structure of the image dataset results in more efficient exploitation of the coherent redundancy in the image dataset [75, 99]. In conventional image and video sequence coding schemes, block-based techniques such as *DCT* have simple implementation structures and are utilized to exploit redundancy among adjacent image samples; Motion-compensated frame prediction is effective to exploit the cross-image redundancy and results in higher compression efficiency. These techniques are applicable to image dataset compression while adapting them to the features of image datasets. In addition, some adjustments can be made in the data structures of the image datasets to improve the coding performance of image dataset compression [55, 96]. This could be done by, for example, explicitly rearranging basis images in the image dataset $\tilde{\mathbb{S}}_I$ to get a new image sequence with similar image contents to increase the cross-image correlation. Then, the following compression techniques can efficiently exploit the redundancy across images.

Random access to coded image datasets at a certain image segment level and selective decoding of small relevant parts of the bit-streams to obtain the required image segments $\mathcal{I}_{\mathcal{B},k,j}(\bar{\mathbf{x}})$ $(\bar{\mathbf{x}} \in \bar{\mathbf{D}}_{\mathcal{B},j},\ k \in \{1, 2, ..., K\},\ j \in \{1, 2, ..., J\})$ are unique and substantial requirements for image dataset compression. An image dataset compression scheme combines a well-designed random access and selective decoding mechanism with some efficient image sequence coding techniques. A specific random access and selective decoding approach is derived by investigating the reverse procedure of the adopted encoding operations. Alternatively, random access and selective decoding can be facilitated by, for example, creating some index tables or sets of pointers accompanying the coded image dataset. Increasing the flexibility of random access to coded image datasets adds a bigger overhead to the compressed bit-stream and thus sacrifices more compression efficiency. A tradeoff needs to be reached to balance the compression efficiency and random access flexibility.

3.3 Solution Methods of Image Dataset Compression

Image dataset compression used for image-based virtual environment navigation has drawn a lot of research interest in the image and video processing research community in recent years. Some progress in compressing image datasets has been reported in a variety of publications. Different kinds of compression techniques and schemes are developed for coding image datasets. Most of the initial work on image dataset compression employed VQ-based schemes for the benefit of the convenient implementation of random access to the bit-stream and fast decoding provided by VQ. Quite a few of compression schemes are designed by modifying existing image and video sequence coding techniques, many of which can be found in a variety of international image and video coding standards, by taking advantage of high coding performance of these state-of-the-art techniques and adapting them to the features of image dataset compression. Some other schemes are based on wavelet transform coding to take advantage of the high coding efficiency and the attractive spatial, temporal and quality scalabilities provided by discrete wavelet transforms. Besides all these schemes, there are still some other schemes specially designed to efficiently compress some particular forms of image datasets and meet their specific requirements.

3.3.1 Vector Quantization Compression Schemes

VQ needs a training image set for constructing the codebook. This results in a complicated VQ encoding structure. However, the VQ decoding process is fast, referring to a simple index look-up procedure, and the index table can be encoded with fixed-size codewords for quick flexible random access to the compressed basis images. The training image set required by VQ can be easily generated from the image datasets. The asymmetric structure and constant-length index table coding features make VQ a natural choice for coding image datasets. Most of the original work on compressing image datasets used VQ-based compression schemes [53, 74] to reduce the data size and provide the required random access and selective decoding.

Levoy and Hanrahan [53] utilized a VQ-based compression scheme when they first proposed the popularly applied light field rendering *(LFR)*. They partition the *LFR* image dataset into 2D or 4D tiles of 12D or 48D vectors. A subset of the *LFR* image dataset is chosen as a training set to construct a codebook best approximating the sample vectors. The compression ratio for 48D VQ to exploit coherence in all four dimensions is 24:1. Also, they implemented Lempel-Ziv entropy coding, which is less complex but less efficient than Huffman coding, to compress the codebook and the index array for further increasing the coding efficiency. A total compression ratio of 120:1 was reported after the two-stage pipeline operation.

A VQ-based compression scheme was also applied when concentric mosaic rendering *(CMR)* was initially proposed [74]. A compression ratio of about 12:1 is obtained for compressing *CMR* image datasets and 26:1 if VQ is followed by entropy coding for additional compression. Entropy coding can further enhance the efficiency by representing the coded VQ indices with variable length codes according to their probability, but the whole compressed image dataset should be loaded into memory and decoded first before the coded image dataset is randomly accessible. This extra step slows the decoding process. The compression efficiency of *CMR* is lower than that of *LFR* due to the lower redundancy existing in the 3D *CMR* image dataset compared to the 4D *LFR* image dataset.

In order to further exploit the cross-image redundancy and improve the coding efficiency of VQ-based compression schemes, development work for image dataset compression was undertaken in [86] and [85]. Based on disparity compensated prediction *(DCP)*, a hierarchical compression scheme for *LFR* was proposed by Tong and Gray [85]. In the hierarchical *DCP (HDCP)* scheme, basis images are put into a certain number of layers based on viewing resolution. The lowest layer itself corresponds to the low resolution version of the image dataset. Each enhancement layer plus all its lower layers constitutes a higher version of the image dataset. The lowest layer is compressed by VQ with constant-length coded indices for fast random access. The enhancement layers are coded by disparity compensated prediction with reference to the basis images in lower layers. The prediction residues are quantized by VQ. Also based on *DCP*, a scheme named hierarchical disparity-compensated VQ *(HDVQ)* is presented in

[86]. An optional single-frame prediction or multiple-frame prediction could be utilized as a tradeoff among prediction quality, implementation complexity and access speed. Multiple-prediction makes the prediction chain longer and thus increases the coding efficiency but complicates the random access. *HDVQ* can increase the compression efficiency as much as ten times over that of generic tree-structured VQ at the cost of a slower rendering speed by a factor smaller than 2.

VQ reduces the data sizes of image datasets while enabling random access and selective decoding required by IBR. However, VQ is a pixel-based technique. It does not fully exploit the high redundancy between basis images. Thus, its compression efficiency is limited and can not meet the requirement of higher compression ratios in many applications.

3.3.2 Modified Standardized Compression Schemes

Some compression schemes for coding image datasets have been designed as a result of modifications to standardized video coding schemes recommended in a variety of international video coding standards. Following the first VQ-based compression scheme for compressing *CMR* image dataset together with the introduction of *CMR*, Shum et al. [75] proposed an MPEG-like compression scheme developed by modifying a standard MPEG scheme for *CMR* used in virtual reality. By taking each normal planar image or panorama image as a video frame, independently coded intra **I** frames and bi-directionally predicted **B** frames are employed. The **B** frames exploit the inter frame redundancy and enhance the coding efficiency. Predictive **P** frames are not employed as they increase the length of the dependency chain in the decoding process. All the reference frames are loaded into the memory first for later use in the case of panorama *CMR* image datasets. For planar *CMR* image datasets, only some of the reference frames instead of all are loaded first due to the much large number of reference frames than that of panorama *CMR* image datasets. Macroblocks (MB) in each frame are scanned and processed in a vertical order so that all the image pixels in an image column are involved in a group of consecutive blocks. Global motion estimation is performed for each whole block group. A set of pointers to groups of blocks is embedded in the compressed bit-stream for supporting random access.

Zhang and Li [99] proposed a reference block codec *(RBC)* approach and further improved the coding performance of the MPEG-like scheme for coding *CMR* image datasets. It classifies input images into anchor **A** frames and predicted **P** frames. Anchor **A** frames are almost the same as intra **I** frames in standard schemes and the predicted **P** frames are coded with reference to anchor **A** frames by motion compensation. All the MBs at the same horizontal location are grouped as a basic access and decoding unit called MB group *(MBG)*. After a rate-distortion comparison by separately taking each of the two neighbor anchor **A** frames as the reference frame, the reference of each predicted *MBG* is limited to just one anchor **A** frame to control the decoding delay. A two-level hierarchical index table is stored in the bit-stream for random access decoding. The first level of index table stored in the bit-stream header registers the length of each coded frame, and the second level of the index table stored in frame headers registers the length of each coded *MBG*. Elaborately designed caches are employed to avoid unnecessary repeat of decoding the same *MBG* and speed up the image rendering. *RBC* provides high coding performance and enables just-in-time image decoding and rendering.

The multiple reference frame *(MRF)* scheme in [98] applied the same strategy for the compression of Lumigraph image datasets. *MRF* also divides the basis images into anchor **A** frames and predicted **P** frames. But each predicted **P** frame has four reference **A** frames instead of two due to the 2D structure of Lumigraph image datasets in place of the 1D structure of *CMR* image datasets. For fast image decoding and rendering purpose, only one of the four anchor **A** frames is chosen as the actual reference frame after the rate-distortion comparison. Also, random access and selective decoding are provided by a two-level hierarchical index table inserted in the bit-stream. However, the second level of the index table stored in the frame headers registers the length of each coded MB instead of the *MBG* length. Decoder caches are implemented in assistance to just-in-time decoding.

In order to provide high compression ratios, Magnor and Girod [59] proposed the **V**-coder based on the conventional video sequence compression scheme modified for coding the 4D *LFR* image datasets. It extends the video coding scheme to compressing *LFR* image datasets by first converting the image format to *YUV* space and down-sampling the chrominance components in both directions by a factor of two. Then, a number of basis images evenly distributed in the image dataset are coded as **I** frames. All other basis images are coded as predicted frames by taking reference to **I** frames. Due to the 4D structure of the *LFR* image datasets, multiple reference images are employed to improve the prediction accuracy. The overall performance relies on the number of **I** frames in the whole image dataset. All the **I** frames are decoded and kept in local memory before the rendering process starts. They provide a low resolution image dataset. The **V**-coder features fast decoding of the image dataset and only has modest memory requirement.

At the same period of time, a **D**-coder with a hierarchical compression scheme for coding *LFR* image datasets by making use of *DCP* was suggested in [58]. First, basis images on the four corners of the 2D image dataset are independently coded by standard scheme using block-based *DCT*. Then, the center image in the image array is encoded as predicted image with reference to four corner images by using its disparity map. After that, the four middle images on each side of the 2D image dataset array are encoded also as predicted images with reference to the center image and the two nearest corner images. So far, the nine coded basis images divide the 2D array of *LFR* image datasets into four quadrants and the four corner images in each quadrant have been coded. So the above encoding procedure can be recursively applied to each quadrant until all the basis images are encoded. The **D**-coder efficiently exploits the cross-image coherence in the *LFR* image dataset by hierarchical multi-level predictive coding. It increases the rendered image quality by incrementally refining the image dataset with more decoded intermediate basis images. The full *LFR* image dataset is obtained by recursively reconstruct all the predicted basis images beginning from a small number of intra coded basis images.

The **V**-coder and **D**-coder largely improve the compression efficiency. They can provide compression ratio in the order of 10^2 to 10^3 at medium and high image quality. But both encoders do not support the random access and selective decoding. A motion-oriented coding scheme for compressing *CMR* image datasets is proposed in [42]. This scheme, also based on the standardized video coding techniques, has a different coded data structure and different bit-stream syntax adapted for the requirements of *CMR*. The specific motion features of the camera in capturing the *CMR* image datasets are utilized to enhance the coding efficiency and improve the decoding flexibility. Simple one-level side information is embedded into the bit-stream to support the pixel-column random access. While the compression efficiency is kept at a high level, only a small portion of each reference frame needs to be accessed and decoded to get a target pixel column for synthesizing a novel view. This speeds up the image data retrieval and benefits the subsequent fast *CMR* process.

3.3.3 Wavelet-Transform-Based Compression Schemes

Wavelet transforms prove to be an effective and efficient technique for image and video sequence compression. The potential ability to provide spatial, temporal and image quality scalabilities makes wavelet transforms more attractive for some applications with scalable coding requirements. Some compression schemes for coding image datasets are developed based on wavelet transforms.

Luo et al. [55] presented a wavelet-based compression scheme for coding *CMR* image datasets. The *CMR* image dataset they used consists of a stack of mosaic basis images along different radii. These basis images are first aligned by circularly shifting to maximize the similarity and increase the redundancy across all the basis images in the *CMR* image dataset. Then, a 3D separable wavelet transform is applied to the aligned basis images to exploit the redundancy within and across images. A lifting scheme is employed in the decomposition and later in the reconstruction processes for saving memory and speeding up the forward and inverse transforms. Although this proposed scheme provides a compression performance comparable to that of MPEG-2, the required random access and selective decoding are not supported. The whole *CMR* image dataset need to be decoded before the rendering process begins. This results in an initial delay and requires a large memory to hold the *CMR* image dataset.

To overcome the above-mentioned problem, Wu et al. [95] proposed the progressive inverse wavelet synthesis *(PIWS)* scheme to support the random access to the coded image dataset and minimize the relevant computational expense. This scheme is designed based on the analysis of the inverse lifting operation for wavelet synthesis. *PIWS* employ a mixed cache. Due to the in-place computation of the lifting scheme, each memory unit in the mixed cache holds a recovered wavelet coefficient, an intermediate lifting value or a reconstructed image pixel at different stages. A finite state machine is established to indicate the type to data stored in each memory unit. The mixed cache guarantees a low computation expense by properly keeping previous intermediate results. *PIWS* provides more efficient random access and decoding. It only accesses and decodes partial image data necessary for synthesizing the required novel view in just-in-time rendering. Another wavelet-based compression scheme providing the required random access for image-based virtual environment navigation is proposed in [43]. This compression scheme is based on the approach of a lifted wavelet transform combined with embedded block entropy coding. It is adapted to compressing image datasets and facilitates a column random access mechanism. Although some coding efficiency is sacrificed because no inter-image redundancy is exploited with independent intra-image coding, it reduces the decoding delay, simplifies the implementation of a random access mechanism and easily facilitates real-time image decoding.

As an effort to further exploit the cross-image redundancy and improve the coding efficiency of the 3D wavelet-based compression technique, a smart rebinning scheme is presented in [96] for coding *CMR* image datasets. This scheme aligns the basis images along the horizontal direction according to the calculated displacements. Then, it cuts the images into stripes and pastes these stripes into multi-perspective panoramas, which are smoother and more natural looking. In this way, the cross-image correlation is largely enhanced while the high correlation within the images is retained. After the image rearrangement, some support regions of the multi-perspective panoramas are not rectangular any more. Either a padding scheme is employed to make the support regions rectangular for applying a rectangular 3D wavelet codec, or an arbitrary shape 3D wavelet codec is applied directly to the multi-perspective panoramas. If a padding scheme is used, the 3D wavelet-based compression scheme in [55] or the scheme in [95] for an efficient decoding can be applied. The image rebinning largely improves the efficiency of cross-image filtering. The smart rebinning scheme improves the performance of a direct 3D wavelet scheme by 4.3 dB and outperforms the MPEG-2 scheme by 3.7 dB on the tested image datasets.

Girod et al. [23] applied wavelet transforms for coding *LFR* image datasets. They developed a novel scheme based on 4D wavelet transforms with a disparity compensated lifting scheme. In their wavelet-based scheme, a 2D inter-view wavelet transform and a 2D intra-view wavelet transform are performed on four dimensions of the 2D *LFR* image array. The inter-view transform across the 2D image array is performed by disparity-compensated lifting with the Haar or the bi-orthogonal Cohen-Daubechies-Feauveau 5/3 wavelet kernels. The intra-view transform is performed with the bi-orthogonal Cohen-Daubechies-Feauveau 5/3 wavelet kernel. A modified block-wise set partitioning in hierarchical trees *(SPIHT)* algorithm [69] is utilized to coding the transform coefficients in each subband. The bit-streams in all blocks are truncated and assembled by the rate-distortion optimization criteria. This 4D wavelet transform scheme outperforms the previous wavelet-based compression schemes up to 6 dB in overall image quality.

3.4 Comparison of Image Dataset Compression Schemes

VQ-based compression schemes are characterized by their asymmetric structures with simple fast decoding and easy access to compressed basis images enabled by the constant-length coded index tables. Because of this, they become a natural choice for compressing image datasets. Compared to modified standardized compression schemes and wavelet-based compression schemes, VQ-based coding schemes for image dataset compression provide easy random access and selective decoding with less computational expenses. However, the compression efficiency of VQ-based compression schemes is lower than that of other compression schemes, as VQ is a pixel-based technique in nature and can not fully exploit the heavy redundancy across basis images.

For some applications with the requirement for high compression efficiency, other compression schemes than VQ-based compression schemes should be applied. Modified standardized compression schemes take advantage of the well-developed standardized image and video coding schemes which are modified to support random access and selective decoding with high compression performance. Although they have more complex implementation structure than VQ-based schemes and sacrifice some compression efficiency for facilitating more flexible random access and selective decoding, they still can provide quite high compression efficiency with moderate structure complexity. They are popularly applied for compressing image datasets due to their overall-performance advantage over other schemes.

Wavelet-based compression schemes are characterized by their high compression efficiency and the unique potential for facilitating scalable coding. Image dataset preprocessing or displacement estimation and compensation are required to be incorporated into wavelet-based compression schemes to improve compression efficiency. Wavelet-based compression schemes with displacement-compensated wavelet transforms and embedded progressive bit-plane coding have more complex implementation structures than those of the other schemes. They usually replace other schemes in the case that scalable coding is required in the target applications. The comparison of the coding schemes for image dataset compression is summarized in Table 3.1.

Compression Scheme	Compression Efficiency	Structure Complexity	Scheme Characteristics	References
VQ-based scheme	Low	Low	Simple fast decoding, easy random access and selective decoding	[53],[74], [85],[86]
Modified standardized scheme	High	Moderate	well-developed, widely-applied coding techniques	[42],[58], [59],[75], [98],[99]
Wavelet-transform-based scheme	High	High	temporal, spatial and quality scalabilities	[23],[55], [95],[96]

Table 3.1 Comparison of the coding schemes for image dataset compression used for image-based virtual environment navigation

3.5 Summary

Image dataset compression used for image-based virtual environment navigation is indispensable to reduce the required storage space and adapt to the limited transmission bandwidth. It is significantly different from the conventional image and video sequence compression. Image datasets can mathematically be represented by samples of simplified plenoptic functions. They consist of numerous high-resolution basis images, and tremendous local and cross-image redundancies exist among these basis images. The dominant cross-image displacement in image datasets demonstrates a certain regular pattern for each type of image dataset. In addition, the statistical features of the image datasets can be estimated and employed for compression purpose. Some types of image datasets are accompanied by certain kinds of side information. Adapting coding schemes to the special features of image datasets is very important in the scheme design for effective and efficient image dataset compression.

Some specific requirements need to be satisfied for compressing image datasets. Random access and selective decoding are unique requirements of image dataset compression. Low-complexity compression schemes with low computational expenses or asymmetric compression schemes with low decoding complexity are needed for efficient interactive image rendering. The quality of coded basis images needs to be kept at high levels. Spatial-resolution scalable coding is extremely desirable for image dataset compression. A proper tradeoff among the coding efficiency, the spatial-resolution scalability and the random access and selective decoding flexibility needs to be made. Specifically designed compression schemes should be developed to adapt to the features of image datasets and meet the specific requirements of image dataset compression used for image-based virtual environment navigation.

Progress in image dataset compression has been reported in a variety of publications. Different compression techniques and schemes are developed, ranging from VQ-based schemes to modified standardized schemes and wavelet-based schemes. Although some progress has been made for compressing image datasets, many problems still remain. Currently, the overwhelming majority of the developed compression schemes are limited to dealing with either the *CMR* image datasets [55, 42, 74, 75, 95, 99] or the *LFR* and Lumigragh image datasets [23, 53, 59, 85, 98]. Until now, no publicly available work on compressing cubic-panorama image datasets, which are becoming more popular and expected to be more widely applied with their unique features, has been demonstrated. This thesis work make an effort to fill in this blank in the research on image dataset compression.

The key issues of cubic-panorama image dataset compression will be investigated in the remainder of this thesis. Adapting coding schemes to the special features of image datasets is very important in the design of a scheme for efficient image dataset compression. In Chapter 4, a spatially consistent representation of cubic-panorama image datasets is proposed by taking advantage of this specific form of image dataset. Then, the compression schemes are adapted to this spatially consistent representation of cubic-panorama image datasets, resulting in superior compression performance. Moreover, providing spatially scalable coding is extremely desirable for image dataset compression in order to synthesize the frequently required zooming views with steady image quality. In order to satisfy this fundamental requirement, a scalable lifted wavelet-based coding scheme with displacement compensation is specially developed for cubic-panorama image dataset compression in Chapter 5. In addition, with regard to the unique requirement of random access and selective decoding for image dataset compression, a specific rectangular sub-region access mechanism is designed for cubic-panorama image dataset compression in the developed scalable lifted wavelet-based coding scheme. A suitable tradeoff among coding efficiency, spatial resolution scalability and random access flexibility is reached. Also, making full use of the features of image datasets is a principle strategy of designing efficient coding schemes for image dataset compression. As a specific feature of cubic-panorama image datasets, the dominant image displacement across basis images demonstrates a regular pattern determined by the motion of the cameras producing cubic-panorama image datasets. A novel global displacement estimation and compensation approach is developed based on the dominant image displacement feature of cubic-panorama image datasets in Chapter 6.

Chapter 4

Cubic-Panorama Image Dataset Compression with the Spatially Consistent Representation

The best general-purpose representation of environment mapping is the projection onto a cube [25]. Providing omni-directional image information, cubic panoramas can be easily generated and manipulated. Novel-view images can be synthesized conveniently from cubic-panorama image datasets. They are becoming more popular and are expected to be more widely applied with their unique features in image-based virtual environment navigation [7, 21, 90]. However, because of the huge sizes of the cubic-panorama image datasets, an efficient coding scheme for compressing cubic-panorama image datasets is crucial to make their applications practical.

Although some compression schemes can be utilized to encode cubic-panorama image datasets, these schemes are designed for compressing planar images with a rectangular support. It is inevitable to sacrifice some coding efficiency with the planar image representation of cubic-panorama image datasets because of the inconsistency on the boundaries of connected side images. In order to overcome this problem, a spatially consistent representation for cubic-panorama image datasets is proposed and applied to compressing cubic-panorama image datasets. A special block padding algorithm is designed to construct the reference blocks for displacement estimation and compensation.

Two different compression schemes are utilized and adapted to the spatially consistent representation of cubic-panorama image datsets. Improved coding performance is achieved with the spatially consistent representation of cubic-panorama image datasets due to the consistent, unconstrained displacement estimation and compensation.

4.1 Introduction

Cubic panoramas can be easily generated and manipulated. Novel-view images can be synthesized conveniently with the assistance of interpolated intermediate cubic panoramas derived from the basis cubic panoramas or basis images in cubic-panorama image datasets. Compared with the cylindrical and spherical formats of panoramic images [10, 82], cubic panoramas are more suitable for being used as the image database in some applications, like navigation in image-based virtual environments [25]. Cubic-panorama image datasets are more widely applied and are becoming more popular [7, 21]. However, originally generated cubic-panorama image datasets involve a large amount of data. Without proper compression, they occupy huge amounts of space to keep on storage media and require high bandwidths to transmit over networks. Efficient compression of the cubic-panorama image datasets is crucial to reducing the required storage space and transmission bandwidth and facilitating the applications of cubic-panorama image datasets in image-based virtual environment navigation.

In conventional compression schemes, cubic-panorama image datasets are treated as planar image sequences with a rectangular support. It is inevitable to sacrifice some coding efficiency because of the inconsistency on the boundaries of connected side images. In order to deal with this problem, a spatially consistent representation for cubic-panorama image datasets is proposed based on the previous work in [44]. The image boundary constraints on displacement vectors are eliminated. The search for displacement vectors as well as matching blocks in the reference frames can be naturally extended across side-image boundaries into neighbor side images in all directions. A specific algorithm for constructing the reference blocks around the corners of the cube is designed for displacement estimation and compensation. Optimized matching reference blocks can be obtained to further reduce the prediction errors and improve the compression efficiency. Two compression schemes respectively based on the baseline framework of block-based hybrid coding and displacement-compensated wavelet transform coding are adopted, and adapted to the features of cubic-panorama image datasets. The experimental results of applying the proposed compression schemes to coding sample cubic-panorama image datasets are presented. It is shown that superior coding performance is achieved with the spatially consistent representation compared with the generic planar representation of cubic-panorama image datasets.

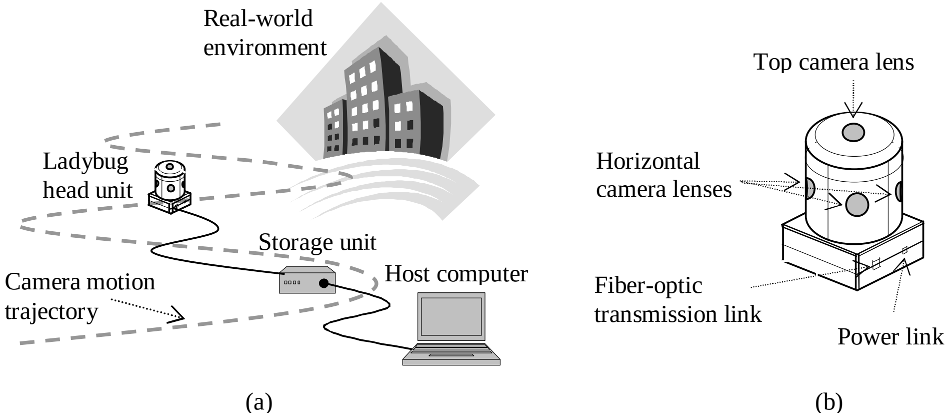
Following this section of introduction to cubic-panorama image dataset compression, Section 2 presents the process of cubic-panorama image dataset generation. In Section 3, the spatially consistent representation for cubic-panorama image datasets is proposed together with the investigation on reference block padding around the corners of the cube and the discussions on unrestricted displacement estimation and compensation. Section 4 describes the block-based hybrid compression scheme and the displacement-compensated wavelet transform compression scheme adopted and adapted for coding cubic-panorama image datasets with the spatially consistent representation. The experimental results of applying the proposed compression schemes to coding sample cubic-panorama image datasets with and without the spatially consistent representation are presented and compared in Section 5. Finally the discussions on cubic-panorama image dataset compression in this chapter are summarized in Section 6.

4.2 Cubic-Panorama Image Dataset Generation

In order to obtain the required cubic-panorama image dataset, we need to undergo the raw image sequence acquisition process, construct cubic-panorama side images from the captured raw images, and perform the necessary image format conversion.

4.2.1 Raw Image Sequence Acquisition

The imaging system we used for acquiring raw image sequences is a multi-sensor omni-directional camera system shown in Fig. 4.1(a). It consists of a camera head unit, a storage unit and a portable host computer. The camera head unit is a Ladybug camera (Fig. 4.1(b)) from Point Grey Research. It incorporates six *ICX204AQ* color CCD image sensors and has six high quality micro lenses with the focal length of 2.5mm. One lens pointing straight up is configured on the top of the camera head unit and five lenses pointing horizontally are assembled in a horizontal circle. The camera head is pre-calibrated by the manufacturer to enable high-quality image processing. The lens settings, e.g. the iris and the focus, are fixed to keep the cameras calibrated during the acquisition process. The CCD sensors and the lenses are packed tightly to minimize the parallax effects. This camera head allows easy acquisition of panoramic images and has been utilized in several applications [7, 31, 88]. The storage unit involves a number of large-capacity hard drives to record the uncompressed raw image sequences with huge sizes, and the host computer is used to control the acquisition procedure and to process and retrieve the recorded image sequences.



(a) (b)

Fig. 4.1 Cubic-panorama raw-image sequence acquisition: (a) the imaging system and (b) the multi-sensor omni-directional Ladybug camera head.

The multi-sensor camera head outputs 6 simultaneously sampled color filter array (CFA) raw images $\mathcal{I}^{Br}_{\mathcal{R},\,ki}$ $(\forall\, i\!\in\!\{1,2,...,6\})$ of the Bayer raw image signal space $\tilde{\mathcal{S}}^{Br}_{\mathcal{R}}$ at the k^{th} shot instant of a sequential acquisition process. Here the superscript Br denotes Bayer, the subscript $\mathcal{R}$ represents raw images, and $k \in \{1, 2, ...,K\}$ where $K \in \mathbb{Z}$ is the number of shots taken in the whole acquisition process. The mathematic structure most useful in describing sampling of images is the lattice [16]. The Bayer *CFA* sampling structure with the raw images sampled on a rectangular lattice Λ is shown in Fig. 4.2 [15]. The *CFA* sampling lattice $\Lambda = LAT(\mathbf{V})$, the sampling matrix $\mathbf{V} = [\bar{\mathbf{v}}_1|\bar{\mathbf{v}}_2]$ and the basis vectors $\bar{\mathbf{v}}_1 = [\mathbf{X}, 0]^T; \bar{\mathbf{v}}_2 = [0, \mathbf{Y}]^T$ where $\mathbf{X}$ and $\mathbf{Y}$ are the spatial sampling periods in horizontal and vertical directions respectively. The raw Bayer *CFA* images $\mathcal{I}^{Br}_{\mathcal{R},\,ki}$ consist of *R, G* and *B* components $\mathcal{I}^{Br}_{\mathcal{R},\,ki} = \big\{\mathcal{I}^{Br}_{\mathcal{R},\,ki,\,p}|\ \forall p \in\!\{r, g, b\}\big\}$. Correspondingly, the sampling structures $\Lambda_p\big(p\!\in\!\{r, g, b\}\big)$ of the *R, G* and *B* components are represented as following:

$$\begin{aligned}\Lambda_r &= [\mathbf{X}, 0]^T + LAT[\bar{\mathbf{v}}_{1r}|\bar{\mathbf{v}}_{2r}],\ \bar{\mathbf{v}}_{1r} = [2\mathbf{X}, 0]^T, \bar{\mathbf{v}}_{2r} = [0, 2\mathbf{Y}]^T\ ,\\ \Lambda_g &= LAT(\mathbf{V}_g) = LAT[\bar{\mathbf{v}}_{1g}|\bar{\mathbf{v}}_{2g}],\ \bar{\mathbf{v}}_{1g} = [2\mathbf{X}, 0]^T, \bar{\mathbf{v}}_{2g} = [\mathbf{X}, \mathbf{Y}]^T\ ,\\ \Lambda_b &= [0, \mathbf{Y}]^T + LAT[\bar{\mathbf{v}}_{1b}|\bar{\mathbf{v}}_{2b}],\ \bar{\mathbf{v}}_{1b} = [2\mathbf{X}, 0]^T, \bar{\mathbf{v}}_{2b} = [0, 2\mathbf{Y}]^T\ .\end{aligned} \tag{4.1}$$

The unit-cell hyper-volume $d(\Lambda_g) = |\det \mathbf{V}_g| = 2\mathbf{XY}$ for Λ_g is half of that for Λ_r or Λ_b. Therefore, the sampling density of the G component as the reciprocal of $d(\Lambda_g)$ is twice that of R or B. The union of the lattice Λ_g and the shifted lattices Λ_r and Λ_b composes Λ: $\cup_{\forall\, p\in\{r,g,b\}}\, \Lambda_p = \Lambda$.

In order to obtain larger vertical field of view, the CCD sensors capture the five horizontal views $\mathcal{I}^{Br}_{\mathcal{R},\,ki}(\forall\, i\!\in\!\{1, 2, ..., 5\})$ in portrait orientation with resolution 768x1024 per view. The view overlap of adjacent lenses is controlled within a small amount of 5 to 10 pixels. Therefore, the horizontal circumference is covered by approximately 3800 pixels. Totally the six CCD sensors capture approximately 4.7M effective pixels at each shot. The uncompressed streaming raw Bayer *CFA* image data is transmitted to the

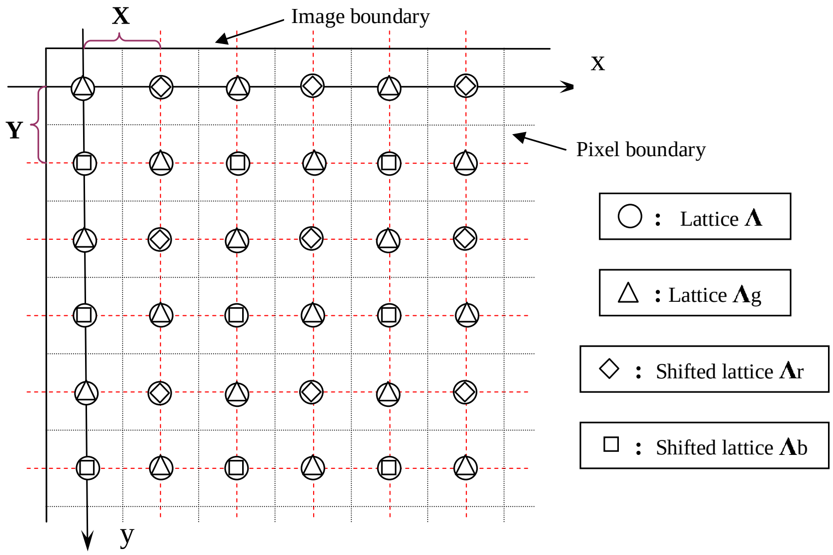


Fig. 4.2 Bayer *CFA* sampling structure with the sampling lattice Λ as the union of the lattice Λ_g and the shifted lattices Λ_r and Λ_b [15].

storage unit through a fiber-optic cable at an adjustable rate of 3.75, 7.5, 15 or 30 fps.

4.2.2 Panoramic Side-Image Formation

One basis cubic panorama consists of six side images $\mathcal{I}^{Rgb}_{\mathcal{B},k} = \{\mathcal{I}^{Rgb}_{\mathcal{B},kj} | \forall j \in \{1, 2, ..., 6\}\}$ derived from the k^{th} set of six raw images and projected on the six faces of the cube. Here the superscript Rgb denotes the RGB color format and the subscript $\mathcal{B}$ represents basis images. As a preliminary process to obtain these six side images of a basis cubic panorama, full RGB images $\mathcal{I}^{Rgb}_{\mathcal{R},ki,s}$ $(\forall s \in \{R, G, B\})$ in the RGB raw image signal space $\tilde{\mathcal{S}}^{Rgb}_{\mathcal{R}}$ need to be reconstructed from the acquired raw Bayer *CFA* tiled images $\mathcal{I}^{Br}_{\mathcal{R},ki,p}$ $(\forall p \in \{r, g, b\})$: $\tilde{\mathcal{S}}^{Br}_{\mathcal{R}} \rightarrow \tilde{\mathcal{S}}^{Rgb}_{\mathcal{R}}$, and then obtain $\mathcal{I}^{Rgb}_{\mathcal{B},kj,s}$ $(\forall j \in \{1, 2, ..., 6\})$ from $\mathcal{I}^{Rgb}_{\mathcal{R},ki,s}$ $(\forall i \in \{1, 2, ..., 6\})$. A demosaicking process is required to interpolate other two components of R, G and B at each spatial location where there is only one component captured in $\mathcal{I}^{Br}_{\mathcal{R},ki,p}$ on the sampling lattice Λ.

Assuming the spatial sampling periods **X** and **Y** are equal **X**=**Y** and this same sampling period is taken as the unit of length, the raw Bayer *CFA* color images $\mathcal{I}^{Br}_{\mathcal{R},\,ki}$ can be represented by using the luma component $\mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,L}$ and the chroma components $\mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C1}, \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C2}$ in $\tilde{\mathcal{S}}^{Lcc}_{\mathcal{R}}$ – the *LCC* raw image space, $\tilde{\mathcal{S}}^{Lcc}_{\mathcal{R}} \rightarrow \tilde{\mathcal{S}}^{Br}_{\mathcal{R}}$:

$$\begin{aligned}\mathcal{I}^{Br}_{\mathcal{R},\,ki}[n_1,n_2] = \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,L}[n_1,n_2] + \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C1}[n_1,n_2]\exp\big(j\pi(n_1+n_2)\big) +\\ \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C2}[n_1,n_2]\exp(j\pi n_1) - \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C2}[n_1,n_2]\exp(j\pi n_2),\end{aligned} \tag{4.2}$$

In order to reduce the reconstruction error, a least-squares design methodology [17] can be utilized in the filter design for extracting the estimated values of the chroma components $\mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C1}$ and $\mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C2}$, and then the luma component $\mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,L}$ can be estimated by using (4.2). After that, $\mathcal{I}^{Rgb}_{\mathcal{R},\,ki,\,s}$ $(\forall s \in \{R,G,B\})$ can be determined by the following equations [15]

$$\begin{aligned}\mathcal{I}^{Rgb}_{\mathcal{R},\,ki,\,R} &= \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,L} - \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C1} - 2\,\mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C2}\ ,\\ \mathcal{I}^{Rgb}_{\mathcal{R},\,ki,\,G} &= \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,L} + \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C1}\ ,\\ \mathcal{I}^{Rgb}_{\mathcal{R},\,ki,\,B} &= \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,L} - \mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C1} + 2\,\mathcal{I}^{Lcc}_{\mathcal{R},\,ki,\,C2}\ .\end{aligned} \tag{4.3}$$

After *R, G* and *B* components are interpolated on Λ, a group of six raw full RGB images $\mathcal{I}^{Rgb}_{\mathcal{R},\,ki}$ $(\forall i \in \{1,2,...,6\})$ can be used to create a basis cubic panorama $\mathcal{I}^{Rgb}_{\mathcal{B},\,k}$ which consists of six side images $\mathcal{I}^{Rgb}_{\mathcal{B},\,kj}$ $(\forall j \in \{u,b,l,f,r,d\})$ in the full RGB space $\tilde{\mathcal{S}}^{Rgb}_{\mathcal{B}}$. The subscripts *u, b, l, f, r* and *d* represent the up, back, left, front, right and down side images respectively. The camera head is pre-calibrated to facilitate high quality image analysis and processing, and this calibration is retained during the whole process of acquisition. Now the camera calibration parameters are retrieved for image projection. The original overlap of 5 to 10 pixels between adjacent $\mathcal{I}^{Rgb}_{\mathcal{R},\,ki}$ is fused by using a standard blending technique in OpenGL [7] or the more sophisticated multiperspective plane sweep (*MPPS*) technique [88]. Then, by locating the geometric center of a cube **C** at the same common projection center of the group of the six raw images, the six faces of **C** are projected into the surface **P** of the stitched raw images (There is no raw image corresponding to the down side face of **P**), and texture mapping is performed from raw images $\mathcal{I}^{Rgb}_{\mathcal{R},\,ki}$ $(\forall i\in\{1,2,...,6\})$ of resolution 768x1024 onto the surface of **C** (Fig. 4.3). Six side images $\mathcal{I}^{Rgb}_{\mathcal{B},\,kj}$ $(\forall j\in\{u,b,l,f,r,d\})$ of resolution 512x512 or 1024x1024 viewed from the inside of the cube at the projection center are rendered on the six faces of **C** by texture mapping. Each of these side views has 90 degrees of FOV in both the horizontal and vertical directions. There is a blind area in the bottom view of the cubic panorama due to the lack of a lens facing down in the camera head unit. An example of a set of six resulting side images composing a cubic panorama is shown in Fig. 4.4 in a flattened out pattern.

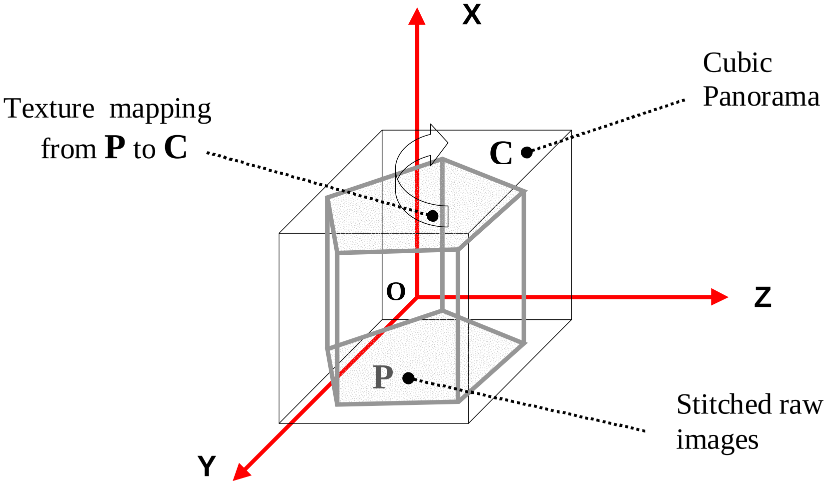


Fig. 4.3 Cubic panorama generation: texture mapping from six stitched raw images to a cubic panorama consisting of six side images.

4.2.3 Format Conversion with Component Transformation

In order to reduce the color data correlation of the tristimulus values $\mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,R}$, $\mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,G}$, and $\mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,B}$ and more efficiently represent cubic-panorama image datasets, an inter-

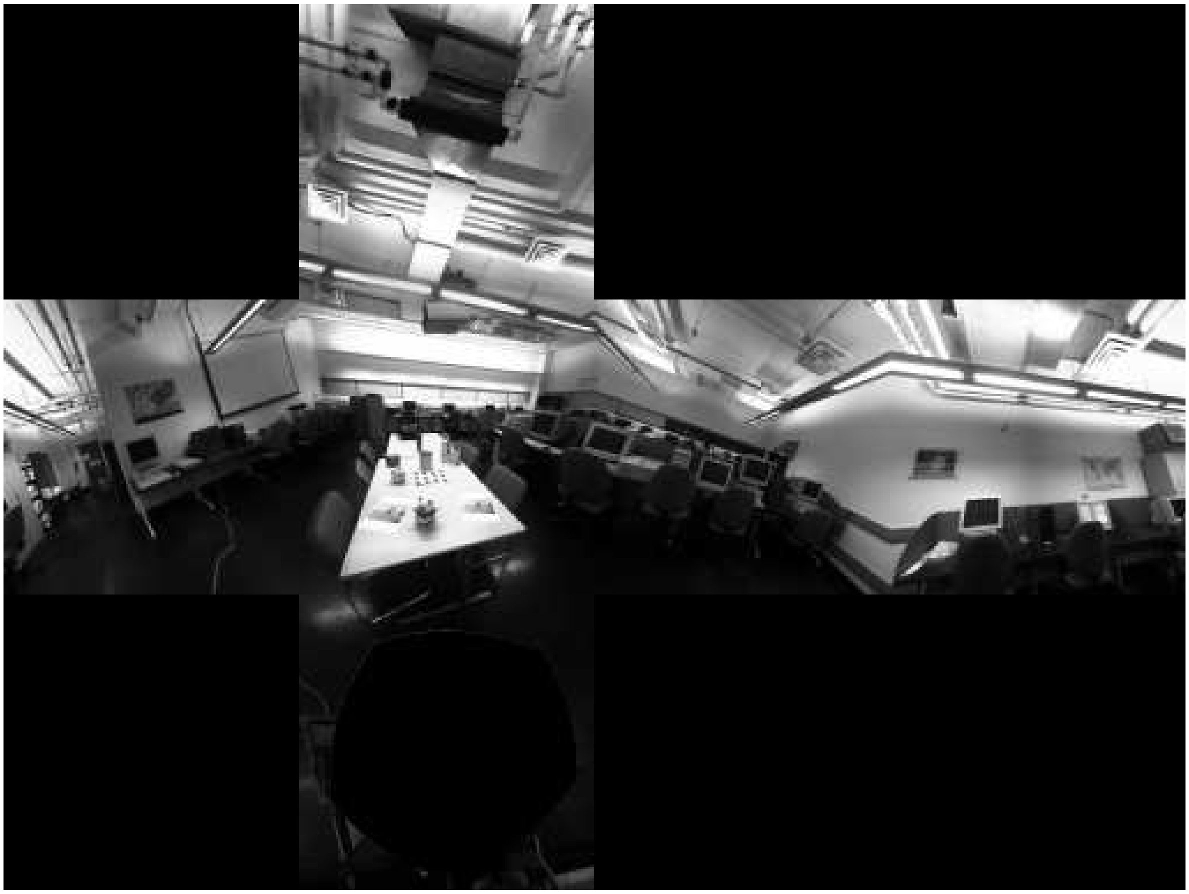


Fig. 4.4 An example of a set of six resulting side images composing a cubic panorama unfolded in a flattened out pattern.

component image transformation from the full RGB image signal space $\tilde{\mathcal{S}}_{\mathcal{B}}^{Rgb}$ to the YUV image signal space $\tilde{\mathcal{S}}_{\mathcal{B}}^{Yuv}$ with one luminance component $\mathcal{I}_{\mathcal{B},\,kj,\,Y}^{Yuv}$ and two color differences or chrominance components $\mathcal{I}_{\mathcal{B},\,kj,\,U}^{Yuv}$ and $\mathcal{I}_{\mathcal{B},\,kj,\,V}^{Yuv}$ is performed for each RGB side image $\mathcal{I}_{\mathcal{B},\,kj}^{Rgb}$ of the cubic panoramas. The forward pixel-wise component transformation is carried out by using the following equation

$$\begin{pmatrix} \mathcal{I}_{\mathcal{B},\,kj,\,Y}^{Yuv} \\ \mathcal{I}_{\mathcal{B},\,kj,\,U}^{Yuv} \\ \mathcal{I}_{\mathcal{B},\,kj,\,V}^{Yuv} \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{pmatrix} \begin{pmatrix} \mathcal{I}_{\mathcal{B},\,kj,\,R}^{Rgb} \\ \mathcal{I}_{\mathcal{B},\,kj,\,G}^{Rgb} \\ \mathcal{I}_{\mathcal{B},\,kj,\,B}^{Rgb} \end{pmatrix}. \tag{4.4}$$

This forward component transform can be implemented more efficiently as

$$\begin{aligned} \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,Y} &= 0.299(\mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,R} - \mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,G}) + \mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,G} + 0.114(\mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,B} - \mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,G}), \\ \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,U} &= 0.564(\mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,B} - \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,Y}),\ \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,V} = 0.713(\mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,R} - \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,Y}). \end{aligned} \tag{4.5}$$

The corresponding reverse component transform is

$$\begin{pmatrix} \mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,R} \\ \mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,G} \\ \mathcal{I}^{Rgb}_{\mathcal{B},\,kj,\,B} \end{pmatrix} = \begin{pmatrix} 1.000 & 0 & 1.402 \\ 1.000 & -0.344 & 0.714 \\ 1.000 & 1.772 & 0 \end{pmatrix} \begin{pmatrix} \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,Y} \\ \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,U} \\ \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,V} \end{pmatrix}. \tag{4.6}$$

As the spatial bandwidths of the chrominance signals are typically lower than that of the luminance signal, and the human visual system (HVS) is less sensitive to high-frequency chrominance signals than luminance signals, the chrominance components $\mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,U}$ and $\mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,V}$ are down-sampled by a factor of 2 in both the vertical and the horizontal directions to further reduce the data sizes. This corresponds to the optimized chrominance format YUV 4:2:0. The sampling structure of the YUV 4:2:0 format of the basis side images is shown in Fig. 4.5. While the sampling lattice of the luminance component $\mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,Y}$ remains the same as $\Lambda_L = \Lambda$, the chrominance components $\mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,U}, \mathcal{I}^{Yuv}_{\mathcal{B},\,kj,\,V}$ are sampled on the shifted rectangular lattice

$$\Lambda_C = [\mathbf{X/2}, \mathbf{Y/2}]^T + LAT[\bar{\mathbf{v}}_{1C}|\bar{\mathbf{v}}_{2C}],\ \bar{\mathbf{v}}_{1C} = [2\mathbf{X}, 0]^T, \bar{\mathbf{v}}_{2C} = [0, 2\mathbf{Y}]^T. \tag{4.7}$$

In the following discussions on image dataset compression, YUV 4:2:0 format of the image datasets will be used as the default image format unless certain other format is explicitly specified. Therefore, the superscript Yuv will be removed for simplification $\mathcal{I}_{\mathcal{B},\,kj} \Leftrightarrow \mathcal{I}^{Yuv}_{\mathcal{B},\,kj}$ and $\mathcal{I}_{\mathcal{B},\,k} \Leftrightarrow \mathcal{I}^{Yuv}_{\mathcal{B},\,k}$.

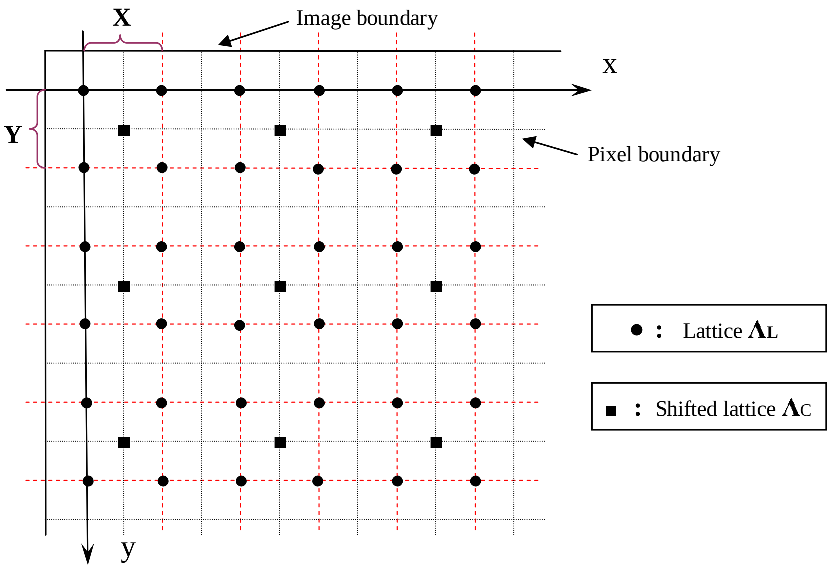


Fig. 4.5 YUV4:2:0 sampling structure consisting of the luminance sampling lattice Λ_L and the shifted chrominance sampling lattice Λ_C.

4.3 Spatially Consistent Cubic-Panorama Representation

A spatially consistent representation of cubic-panorama image datasets is introduced to more efficiently process this specific form of image dataset. A discussion on the reference block padding with a generic smooth filtering algorithm is made in order to apply the spatially consistent representation of the cubic-panorama image datasets to image dataset compression. As a result, unlimited displacement estimation and compensation can be facilitated to improve the performance of cubic-panorama image dataset compression.

4.3.1 Consistent Representation for Cubic-Panoramas

For image dataset compression, the straightforward way to deal with a cubic panorama is to represent it by a planar image, which is obtained by combining the six side images of the cubic panorama to form a single planar image with a rectangular support.

With this generic planar image representation for cubic-panoramas, it is not possible to form seamless consistent connections between the six side images while maintaining a rectangular support. In this case, the inconsistent transition of pixel intensity across side images sacrifices some coding efficiency. Also, the outer boundaries of the connected planar cubic-panorama images constrain the search of displacement vectors and the corresponding displacement-compensated prediction. In order to overcome these problems, a spatially consistent representation of cubic panorama image datasets is introduced (Fig. 4.6), based on the unique features of cubic panoramas.

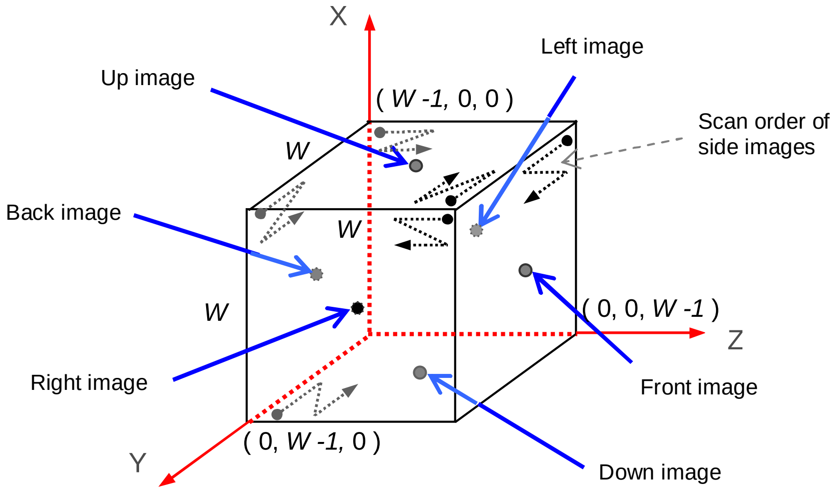


Fig. 4.6 Spatially consistent representation of a cubic panorama with six side images on the six faces of the cube.

In the spatially consistent representation for cubic panoramas, the six side images of a cubic panorama are stitched together corresponding to their spatial locations. Suppose the dominant camera motion direction is along a straight line or the cubic-panorama basis images are aligned though a well-designed projective rectification process [47]. Let this dominant direction of the camera motion coincide with the positive Z-axis direction of the Cartesian coordinate system. Taking the spatial sampling period $\mathbf{X}$ as the unit of length, the back side image $\mathcal{I}_{\mathcal{B},kb}$ is positioned in the XOY plane and the front side image $\mathcal{I}_{\mathcal{B},\,kf}$ is positioned in the plane $z\!=\!W\!-\!1$, where W is the width of the cube side. The left side image $\mathcal{I}_{\mathcal{B},\,kl}$ is in XOZ plane and the right side image $\mathcal{I}_{\mathcal{B},\,kr}$ is in the plane $y\!=\!W\!-\!1$. The down side image $\mathcal{I}_{\mathcal{B},\,kd}$ and the up side image $\mathcal{I}_{\mathcal{B},\,ku}$ are in YOZ plane and the plane $x\!=\!W\!-\!1$ respectively. A cubic panorama consisting of these six side images can be represented in a spatially consistent form as

$$\mathcal{I}_{\mathcal{B},\,k}\,[\,n_1,\ n_2,\ n_3\,] =$$

$$\begin{cases} \mathcal{I}_{\mathcal{B},\,ku}\,[\,W\!-\!1\!-n_2,\ W\!-\!1\!-n_3\,], & n_1 = W\!-\!1, 0 \le n_2 \le W\!-\!1, 0 \le n_3 \le W\!-\!1; \\ \mathcal{I}_{\mathcal{B},\,kb}\,[\,W\!-\!1\!-n_2,\ W\!-\!1\!-n_1\,], & 0 \le n_1 \le W\!-\!1, 0 \le n_2 \le W\!-\!1, n_3 = 0; \\ \mathcal{I}_{\mathcal{B},\,kl}\,[\,n_3,\ W\!-\!1\!-n_1\,], & 0 \le n_1 \le W\!-\!1, n_2 = 0, 0 \le n_3 \le W\!-\!1; \\ \mathcal{I}_{\mathcal{B},\,kf}\,[\,n_2,\ W\!-\!1\!-n_1\,], & 0 \le n_1 \le W\!-\!1, 0 \le n_2 \le W\!-\!1, n_3 = W\!-\!1; \\ \mathcal{I}_{\mathcal{B},\,kr}\,[\,W\!-\!1\!-n_3,\ W\!-\!1\!-n_1\,], & 0 \le n_1 \le W\!-\!1, n_2 = W\!-\!1, 0 \le n_3 \le W\!-\!1; \\ \mathcal{I}_{\mathcal{B},\,kd}\,[\,W\!-\!1\!-n_2,\ n_3\,], & n_1 = 0, 0 \le n_2 \le W\!-\!1, 0 \le n_3 \le W\!-\!1; \\ 0, & \text{otherwise.} \end{cases} \tag{4.8}$$

Every side image in a cubic panorama has four neighbor side images. The transitions of pixel intensity from one side image to any of its neighbor side images are spatially consistent across side image boundaries. This spatially consistent representation embodies the actual spatial connections between the side images constituting a cubic panorama. A seamless consistent combination of the six side images is achieved in this spatially consistent representation.

4.3.2 Reference Block Padding

Displacement estimation and compensation are needed for displacement-compensated cross-image redundancy removal in compressing the spatially consistent representation of cubic-panorama image datasets. With block-based displacement estimation, reference blocks are required to be constructed across side image boundaries in the reference frames to obtain the matching displacement vectors in the displacement search window.

If the search for the displacement vectors comes to the corners of the cube, the reference block need to be constructed with image samples from three adjacent side images. In this case, no full-size reference blocks can be obtained directly from the side images in the reference frame. Reference blocks need to be padded for block-based displacement estimation and compensation. The cubic panorama is laid out to show the block padding in Fig. 4.7. According to whether the displacement estimation and compensation are performed for the horizontal side images, the up side image or the down side image, different methods are used for padding the corner reference blocks depending on the direction of the dominant image displacement in the current side image. As a result, reference blocks on the same cubic corner may be padded in different ways due to different current side images.

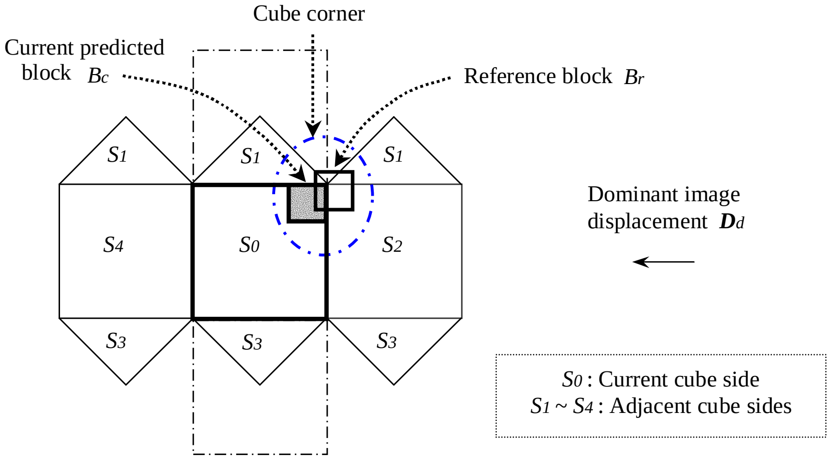

Cube corner
Current predicted block B_c
Reference block B_r
S_1
S_1
S_1
S_4
S_0
S_2
S_3
S_3
S_3
Dominant image displacement $\boldsymbol{D}_d$
S_0 : Current cube side
S_1 ~ S_4 : Adjacent cube sides

Fig. 4.7 Cubic panorama laid out to show block padding.

The construction of reference blocks on the corners of the cube is illustrated in Fig. 4.8(a). According to the direction of the dominant image displacement, the matching reference block B_r of the current predicted block B_c on Side S_0 is more likely located further into Side S_2 rather than Side S_1 across side image boundaries. Therefore, the reference blocks on the corner are padded by using samples of Side S_1 rather than

Side S_2. As a result, the samples in a smaller region R_{GOC} enclosed by GOC need to be padded instead of a bigger region R_{ABOG} enclosed by $ABOG$: $R_{GOC} < R_{ABOG}$. Accordingly, the inconsistent transition boundary L_{FO} in the padding region is shorter than L_{EO} (the inconsistent transition boundary in case the padding is performed by using samples from Side S_2): $L_{FO} < L_{EO}$.

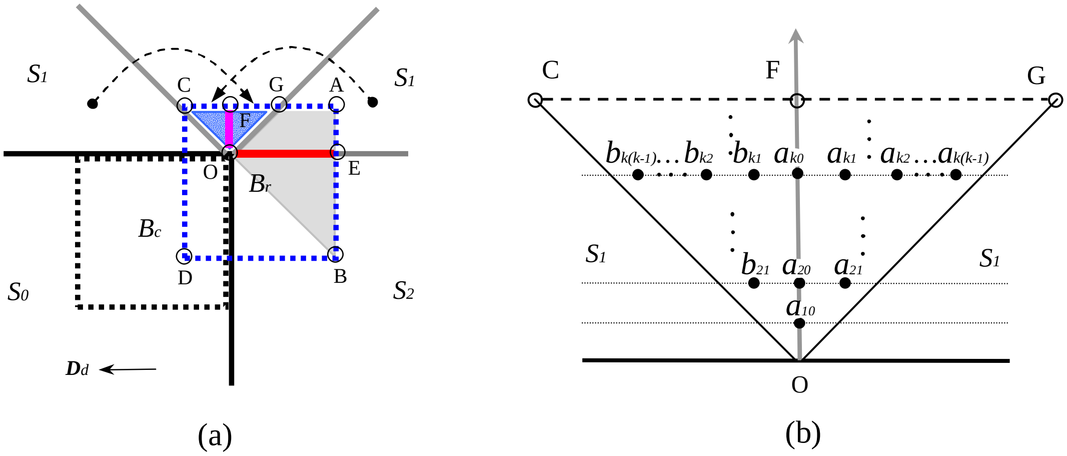


Fig. 4.8 Block padding on the corner of the cube. (a)Reference block construction on the cube corner. (b) Filtering in the padding region.

In the process of reference block padding, firstly the padded image samples $\mathcal{I}_P(a_{ki})$ ($\forall k \in \{1, 2, ..., K_P\}$, $\forall i \in \{0, 1, ..., k-1\}$, K_P is determined by the current search range) enclosed by GOF are produced by simply copying image samples from S_1 on the left side, and the padded image samples $\mathcal{I}_P(b_{kj})$ ($\forall k \in \{1, 2, ..., K_P\}$, $\forall j \in \{1, 2, ..., k-1\}$) enclosed by COF are produced by copying image samples from S_1 on the right side to preserve the consistent sample transition on the padding boundaries L_{GO} and L_{CO}. Secondly, encouraged by the offset-based filtering algorithm for deblocking abrupt changes appearing at the block boundaries of low bitrate video [83], a generalized smooth filtering algorithm with only shifting and addition operations to reduce the computational complexity is introduced to operate in the padding region R_{GOC} (Fig. 4.8(b)). Here the original algorithm is extended from operating on fixed even number of samples to varying odd number of samples in order to adapt it to the boundary condition in the block padding on the cube corners. For the k^{th} row starting from the side image boundary, the center sample at a_{k0} is modified as

$$\mathcal{I}'_P(a_{k0}) = [\mathcal{I}_P(a_{k0}) + \mathcal{I}_P(b_{k1})]/2. \tag{4.9}$$

Then the padded image samples in the region R_{GOF} are updated as

$$\begin{aligned}
&\mathcal{I}'_P(a_{k1}) = \mathcal{I}_P(a_{k1}) - [\mathcal{I}_P(a_{k0}) - \mathcal{I}_P(b_{k1})]/4,\\
&\mathcal{I}'_P(a_{k2}) = \mathcal{I}_P(a_{k2}) - [\mathcal{I}_P(a_{k0}) - \mathcal{I}_P(b_{k1})]/8,\\
&\cdots\cdots\\
&\mathcal{I}'_P(a_{k(k-1)}) = \mathcal{I}_P(a_{k(k-1)}) - [\mathcal{I}_P(a_{k0}) - \mathcal{I}_P(b_{k1})]/2^k.
\end{aligned} \tag{4.10}$$

The padded image samples in the region R_{COF} are modified as following

$$\begin{aligned}
&\mathcal{I}'_P(b_{k1}) = \mathcal{I}_P(b_{k1}) + [\mathcal{I}_P(a_{k0}) - \mathcal{I}_P(b_{k1})]/4,\\
&\mathcal{I}'_P(b_{k2}) = \mathcal{I}_P(b_{k2}) + [\mathcal{I}_P(a_{k0}) - \mathcal{I}_P(b_{k1})]/8,\\
&\cdots\cdots\\
&\mathcal{I}'_P(b_{k(k-1)})) = \mathcal{I}_P(b_{k(k-1)}) + [\mathcal{I}_P(a_{k0}) - \mathcal{I}_P(b_{k1})]/2^k.
\end{aligned} \tag{4.11}$$

An example of cubic-panorama image block padding is given in Fig. 4.9. Image samples in C_a and C_b are copied to S_b and S_a respectively and with a clockwise 90^0 rotation for block C_a and an anti-clockwise 90^0 rotation for block C_b. Then, image samples in S_b and S_a are filtered according to above introduced algorithm.

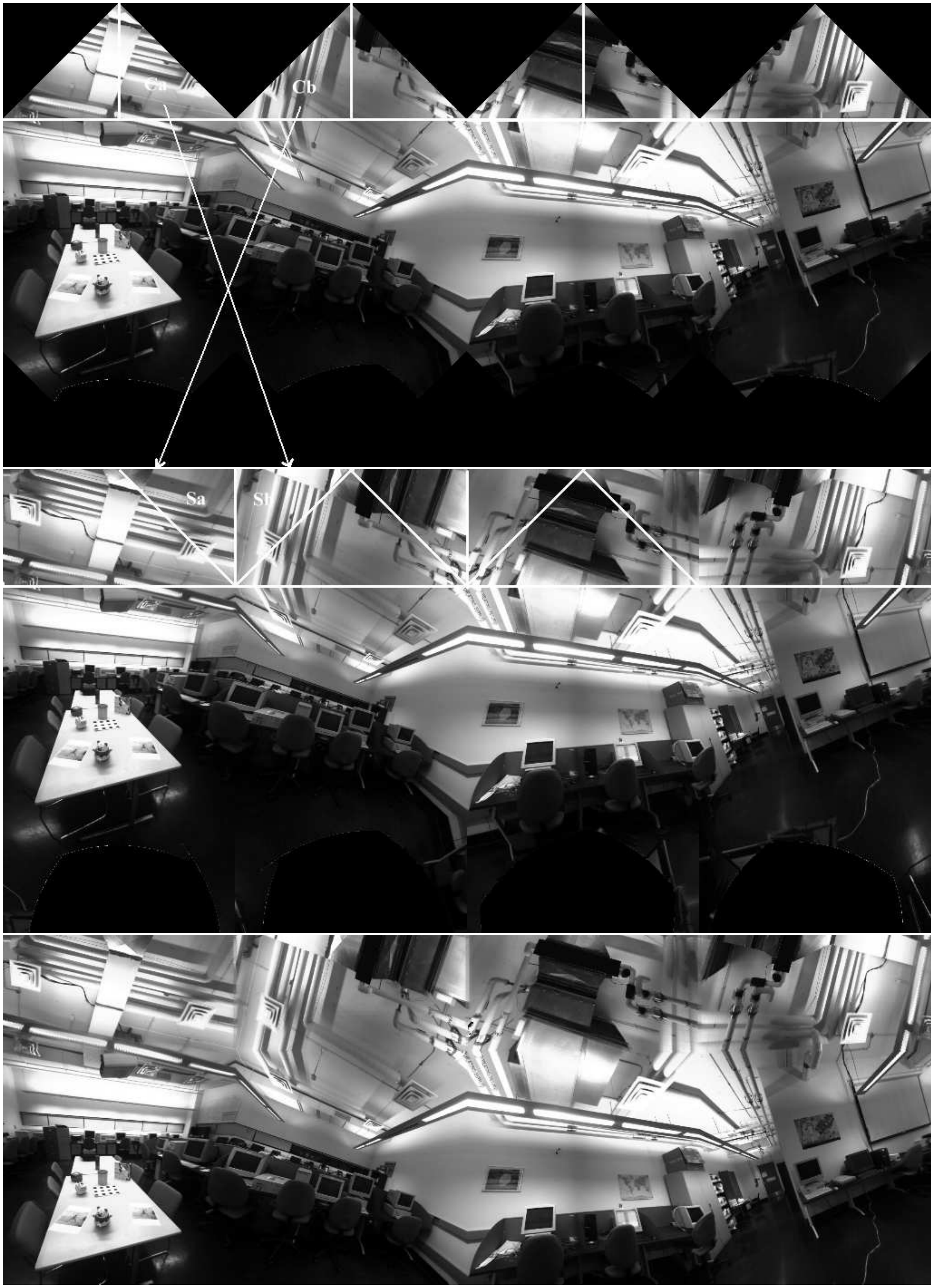


Fig. 4.9 An example of cubic-panorama image block padding: a cubic panorama flattened out with incomplete corner regions (top image), corner regions filled with samples from neighbor regions (middle image) and filled regions after filtering (bottom image).

4.3.3 Unrestricted Displacement Estimation and Compensation

In order to efficiently exploit cross-image redundancy, displacement estimation and compensation are performed on the spatially consistent representation of cubic-panorama image datasets. The search for matching displacement vectors in the reference images can be naturally extended across side image boundaries into neighbor side images. Because the six stitched sides of the cube form a closed surface, the spatially consistent representation eliminates the frame boundaries of the generic planar representation. There is no limitation for the values of the displacement vectors. The search for displacement vectors as well as matching reference blocks can be extended beyond the side image boundaries in all four directions (Fig. 4.10). This makes it possible to obtain an optimized matching reference block for a predicted block to further reduce the prediction error for displacement compensation and improve the compression efficiency.

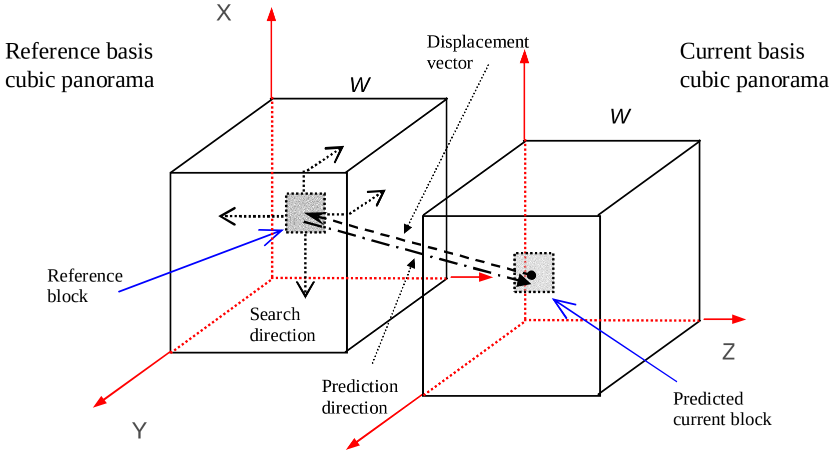


Fig. 4.10 Unrestricted displacement vector search with the spatially consistent representation of cubic-panorama image datasets.

4.4 Cubic-Panorama Image Dataset Compression

Two different coding schemes are adapted to compressing the spatially consistent representation of cubic-panorama image datasets in order to improve the compression performance of the coding schemes applied for image dataset compression. They are the block-based hybrid coding scheme and the displacement-compensated wavelet transform coding scheme. The baseline coding schemes are applied to just show the value of the spatially consistent representation for cubic-panorama image datasets. More detailed descriptions of the coding schemes will be given in the later chapters.

4.4.1 Block-Based Hybrid Coding

A block-based hybrid coding (*BBHC*) scheme is adapted to compressing the basis images of cubic-panorama image datasets with the spatially consistent representation. The structure of the baseline *BBHC* scheme is depicted in Fig. 4.11. The main components included in the scheme are displacement estimation, displacement-compensated prediction, forward and inverse spatial transformation, forward and inverse quantization and entropy coding. The *BBHC* scheme is primarily a combination of a spatial transformation with cross-image prediction enhanced by displacement compensation. With block-based coding, each basis image is partitioned into blocks of image samples. The spatial transform, the displacement estimation and compensation, the quantization and entropy coding are all performed on the block base.

Cross-image prediction is an effective and efficient technique to decorrelate image datasets across basis images. With the assistance of displacement estimation and compensation, the performance of cross-image prediction can be significantly enhanced. For displacement-compensated prediction, the current basis image $\mathcal{I}_{\mathcal{B},k}$ is predicted from an already encoded reference basis image $\mathcal{I}_{\mathcal{B},k'}$. Suppose the displacement vector is $\mathbf{D}[\bar{\mathbf{x}}]$

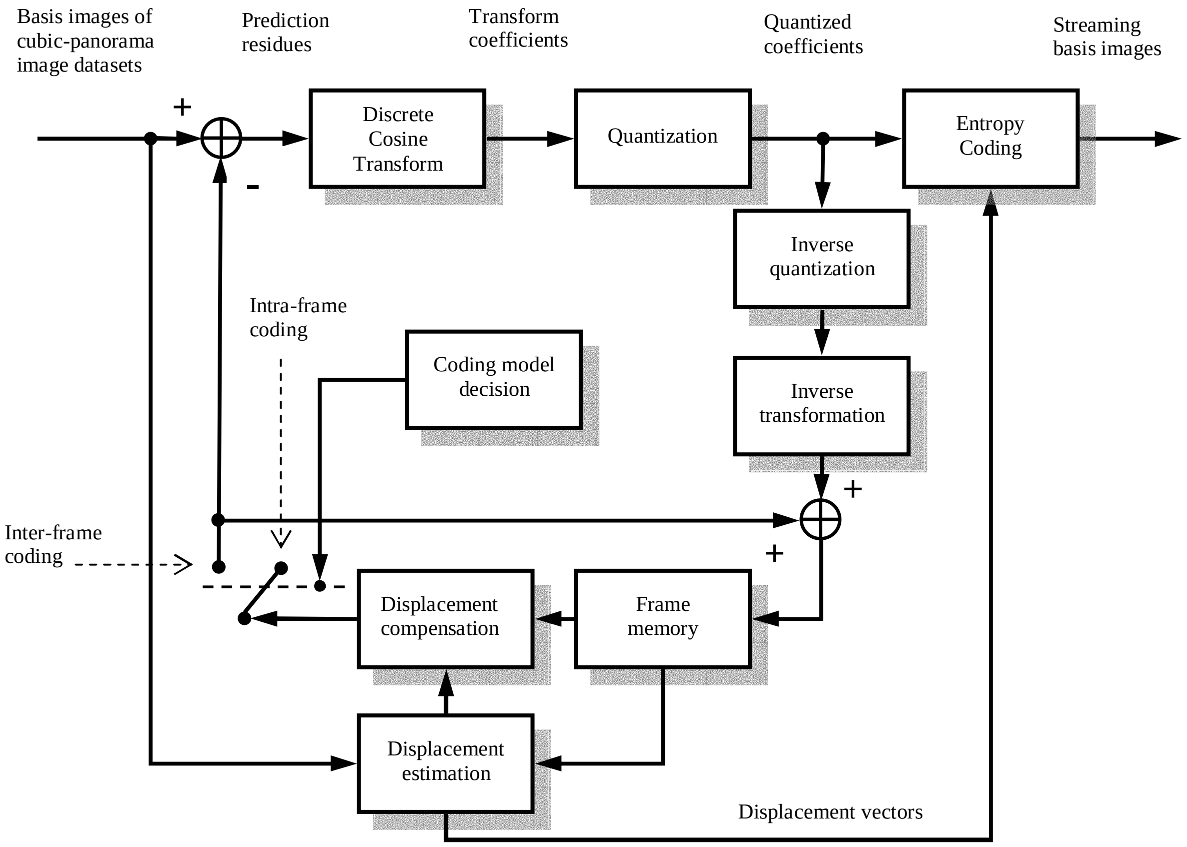


Fig. 4.11 Block-based hybrid coding (*BBHC*) scheme adapted to compressing the spatially consistent representation of cubic-panorama image datasets.

at the sample position $\bar{\mathbf{x}}$, the predicting basis image is given by

$$\tilde{\mathcal{I}}_{\mathcal{B},k}[\bar{\mathbf{x}}] = \mathcal{I}_{\mathcal{B},k'}\big[\bar{\mathbf{x}} + \mathbf{D}[\bar{\mathbf{x}}]\big]. \tag{4.12}$$

Then, instead of $\mathcal{I}_{\mathcal{B},k}$ the prediction error

$$\mathcal{E}_{\mathcal{B},k}[\bar{\mathbf{x}}] = \mathcal{I}_{\mathcal{B},k}[\bar{\mathbf{x}}] - \tilde{\mathcal{I}}_{\mathcal{B},k}[\bar{\mathbf{x}}] \tag{4.13}$$

is spatially transformed, quantized and encoded.

In order to exploit the spatial redundancy existing among image samples in the same basis image, a block-based spatial transformation is applied for each disjoint block of size 8×8. The discrete cosine transform (DCT) is the most popular spatial transform to serve this purpose. The separable, orthogonal, 2D forward DCT for an $M \times N$ block is given as

$$\mathcal{C}_{\mathcal{B},k}[i,j] = \sqrt{\frac{\beta_i \beta_j}{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \mathcal{E}_{\mathcal{B},k}[m,n] \cos(\frac{(2m+1)i\pi}{2M}) \cos(\frac{(2n+1)j\pi}{2M}), \qquad (4.14)$$

where $\beta_i = 1$ for $i = 0$; $\beta_i = 2$ for $i \in \{1, ..., M-1\}$, and $\beta_j = 1$ for $j = 0$; $\beta_j = 2$ for $j \in \{1, ..., N-1\}$. The corresponding 2D inverse DCT is

$$\mathcal{E}_{\mathcal{B},k}[m,n] = \frac{1}{\sqrt{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sqrt{\beta_i \beta_j} \mathcal{C}_{\mathcal{B},k}[i,j] \cos(\frac{(2m+1)i\pi}{2M}) \cos(\frac{(2n+1)j\pi}{2M}). \qquad (4.15)$$

4.4.2 Three-Dimensional Discrete Wavelet Transform Coding

Also, a three-dimensional displacement-compensated discrete wavelet transform coding (*DC-DWTC*) scheme is adapted to encoding the spatially consistent representation of cubic-panorama image datasets. The structure of the baseline *DC-DWTC* scheme is depicted in Fig. 4.12. The *DC-DWTC* process mainly consists of cross-image displacement estimation, the displacement-compensated cross-image discrete wavelet transform, spatial discrete wavelet transform, uniform decomposition coefficient quantization and embedded entropy coding.

Block-based full-search displacement estimation is performed for every block of the current predicted cubic-panorama basis image in the spatially consistent representation. A multi-layer displacement vector searching procedure is conducted in order to

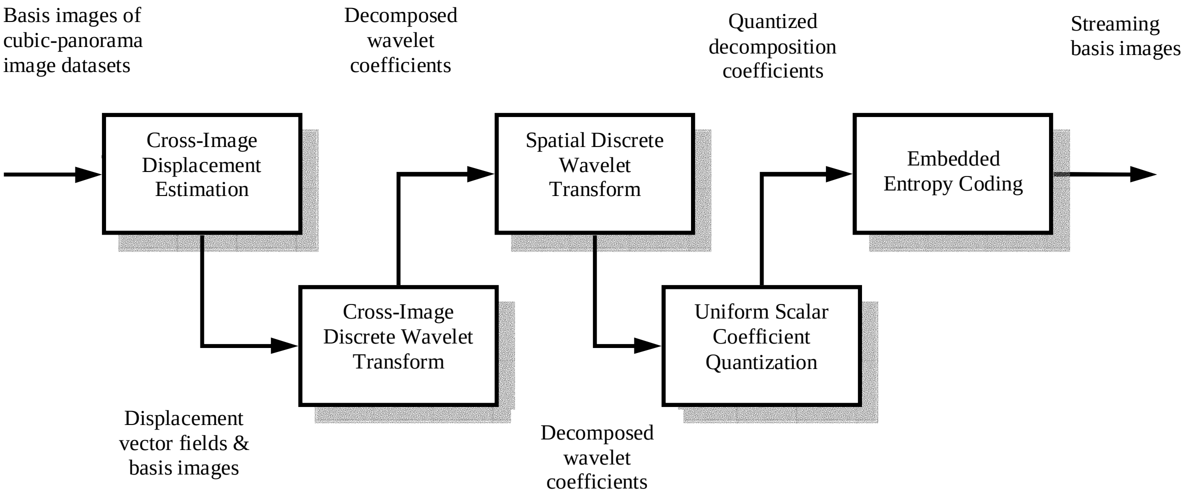

Basis images of cubic-panorama image datasets
Cross-Image Displacement Estimation
Displacement vector fields & basis images
Cross-Image Discrete Wavelet Transform
Decomposed wavelet coefficients
Spatial Discrete Wavelet Transform
Decomposed wavelet coefficients
Uniform Scalar Coefficient Quantization
Quantized decomposition coefficients
Embedded Entropy Coding
Streaming basis images

Fig. 4.12 Displacement-compensated discrete wavelet transform coding (*DC-DWTC*) scheme adapted to compressing the spatially consistent representation of cubic-panorama image datasets.

accelerate the displacement estimation and reduce the computational complexity. The resultant displacement vectors are utilized in the prediction and update stages of the cross-image wavelet transform. The Haar kernel is used in the cross-image wavelet analysis for obtaining the required low delay and facilitating flexible random access to and efficient selective decoding of the coded bit-stream of the cubic-panorama image dataset. A two-dimensional separable spatial wavelet transform is performed on the decomposed wavelet coefficients resulting from the cross-image wavelet analysis process. Decomposition wavelet coefficients in spatial and cross-image subbands are then quantized by uniform threshold quantization. Finally, all the resulting quantized coefficients are encoded in the embedded entropy coding process. More detailed investigation into this wavelet-based compression scheme will be conducted in the following chapter on the scalable lifted wavelet-based scheme with displacement compensation for cubic-panorama image dataset compression.

4.5 Experimental Results

In the experiments, the software implementations of the wavelet-based compression scheme *DC-DWTC* and the block-based hybrid compression scheme *BBHC* presented in the previous section were applied respectively to encoding the testing cubic-panorama image datasets with the spatially consistent representation. The experimental results are compared with those of both of these compression schemes applied to encoding the same testing cubic-panorama image datasets with the generic planar representation. The coding schemes which are compared in the experiments have the same implementation structure, using the same set of compression techniques presented in Fig. 4.11 and Fig. 4.12. However, when the schemes are used to code cubic-panorama image datasets with the spatially consistent representation, they are modified to be able to take this specific format of representation as input, and the required prediction block padding and filtering are integrated into the schemes to facilitate the unrestricted displacement estimation and compensation. When the coding schemes are used to compress cubic-panorama image datasets with generic planar representation, the six $N \times N$ side images of each cubic-panorama basis image are concatenated in the order of top, down and 4 horizontal side images to form a $6N \times N$ planar image with a rectangular support. Therefore, no modification must be made for the coding schemes to compress the cubic-panorama image datasets. In this case, they are treated as generic planar image sequences like the generic video sequences. The testing cubic-panorama image datasets used in the experiments are *Lab* (Fig. 4.13) and *Corridor* (Fig. 4.14). They were generated from the raw image sequences captured with the Ladybug imaging system. Some information about these two testing cubic-panorama image datasets is given in Table 4.1.

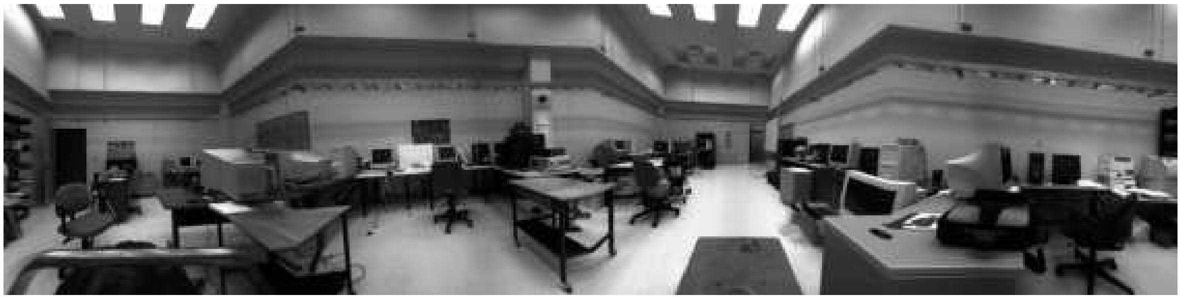


Fig. 4.13 Experimental cubic-panorama image dataset - *Lab* (extended horizontal view) obtained with the Ladybug imaging system.

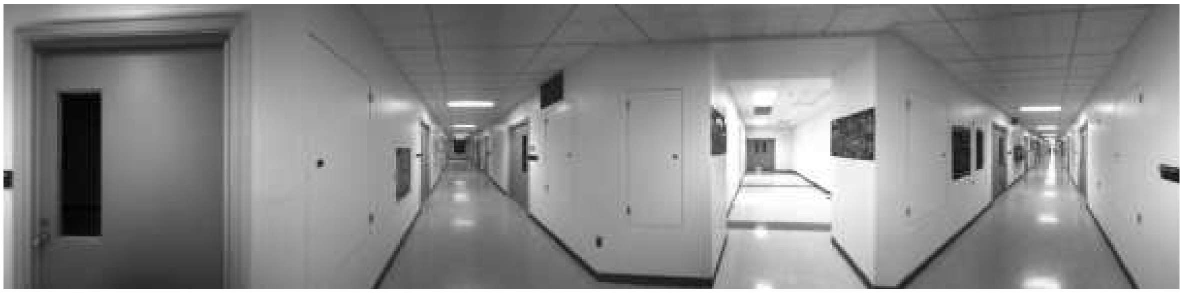


Fig. 4.14 Experimental cubic-panorama image dataset - *Corridor* (extended horizontal view) obtained with the Ladybug imaging system.

Image Dataset	Basis Images	Resolution	Environment	Data Size
Lab	112 frames	6x512x512	Research laboratory	504 MB
Corridor	120 frames	6x512x512	Corridor in the building	540 MB

Table 4.1 Information about the testing cubic-panorama image datasets used in the experiments

The objective metric used for evaluating the image quality of the compressed cubic-panorama image datasets and the performance of the compression schemes applied in the experiments is the peak signal-to-noise ratio (*PSNR*). Suppose the sampling structure of the cubic-panorama basis images $\mathcal{I}_{\mathcal{B},\,k}$ is represented by $\mathcal{L}_{\mathcal{C}}$ defined on the surface of the cube (Fig 4.6), and the total number of the samples on $\mathcal{L}_{\mathcal{C}}$ is $|\mathcal{L}_{\mathcal{C}}| = 6 \times N_1 \times N_2$, where $N_1 \times N_2$ is the resolution of each side image involved in a cubic-panorama basis image. The mean square error (*MSE*) is computed as

$$\mathcal{E}_{MS} = \frac{1}{|\mathcal{L}_{\mathcal{C}}|} \sum_{\forall\,[\,n_1, n_2, n_3\,] \in \mathcal{L}_{\mathcal{C}}} \left(\mathcal{I}_{\mathcal{B},\,k}\,[\,n_1, n_2, n_3\,] - \hat{\mathcal{I}}_{\mathcal{B},\,k}\,[\,n_1, n_2, n_3\,]\right)^2, \tag{4.16}$$

where $\hat{\mathcal{I}}_{\mathcal{B},\,k}\,[\,n_1, n_2, n_3\,]$ is the image sample value at the sample position $[\,n_1, n_2, n_3\,]$ reconstructed from the compressed cubic-panorama basis image $\mathcal{I}_{\mathcal{B},\,k}$. Then, PSNR is given by

$$\text{PSNR} = 10 \log_{10} \frac{\left(\mathcal{I}_{\mathcal{B},max} - \mathcal{I}_{\mathcal{B},min}\right)^2}{\mathcal{E}_{MS}}. \tag{4.17}$$

Firstly, the *DC-DWTC* scheme was applied to compress the cubic-panorama image dataset *Lab*. The experimental results are shown in Fig. 4.15 in terms of the PSNR versus the frame number of the 112-frame *Lab* image dataset. The PSNR of the luminance component **Y** of the *Lab* basis images encoded at the bitrate $R_b = 400\,kbpc$ (kilobits per cube) is presented for each basis image. The spatially consistent representation approach (*SCRA*) shows an improved performance over the generic planar representation approach (*GPRA*). The average PSNR of the entire dataset of the former is 0.48 dB higher than that of the latter.

The experimental results of the *Lab* image dataset encoded with *DC-DWTC* at different bitrates are shown in Fig 4.16. Varying improvements of average PSNR for *SCRA* are obtained with different bitrates compared to *GPRA*, ranging from 0.45 dB to 0.59 dB. Also, the experimental results of the *Corridor* image dataset encoded by using *DC-DWTC* with *SCRA* and *GPRA* at different bitrates are presented in Fig. 4.17. The improvements of average PSNR for encoding the Corridor image dataset with *SCRA* range from 0.38 dB to 0.48 dB. Greater average PSNR values and less improvements are

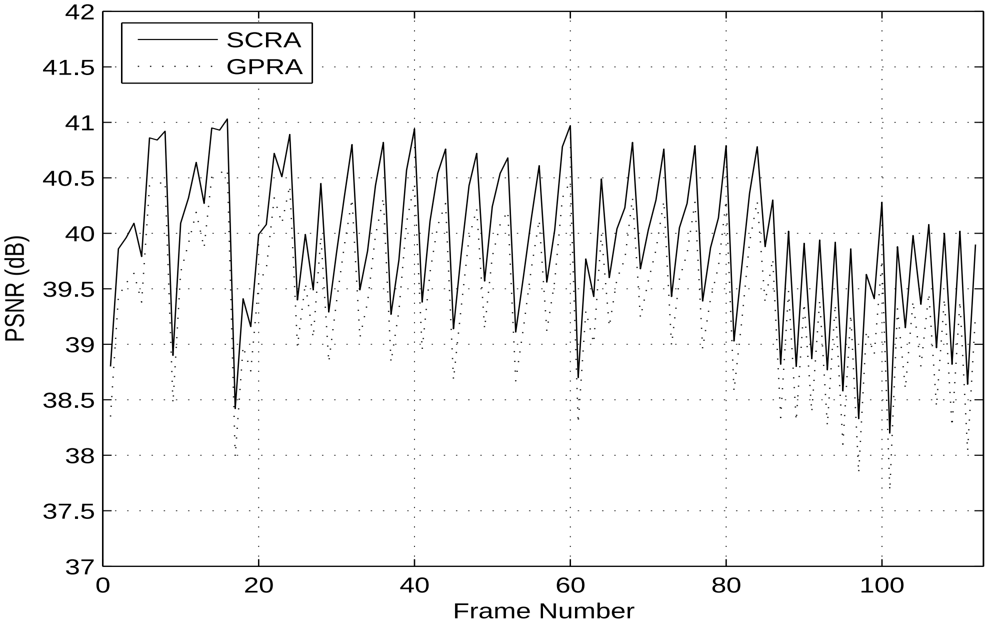


Fig. 4.15 Frame-by-frame PSNR of the luminance component **Y** for the112-frame *Lab* image dataset at $R_b = 400$ kbpc. Average PSNR: *GPRA* — 39.45 dB, *SCRA* — 39.93 dB.

observed for compressing the *Corridor* image dataset due to its less intensive texture feature compared with the *Lab* image dataset.

At several typical compression ratio levels, the compression performance of *DC-DWTC* with *SCRA* applied for the two testing cubic-panorama image datasets *Lab* and *Corridor* are presented in comparison with the performance of *DC-DWTC* with *GPRA* in Table 4.2 (for the compression ratio $\mathcal{R}_C = 30, 60, 100, 200$ respectively). It shows that at the same levels of compression ratios, the performance of *SCRA* compared with that of GPRA in terms of the PSNR of the luminance component is about 0.53 dB higher on average for the *Lab* image dataset and 0.44 dB higher on average for the *Corridor* image dataset over the compression ratio range $\mathcal{R}_C : 30 \sim 200$.

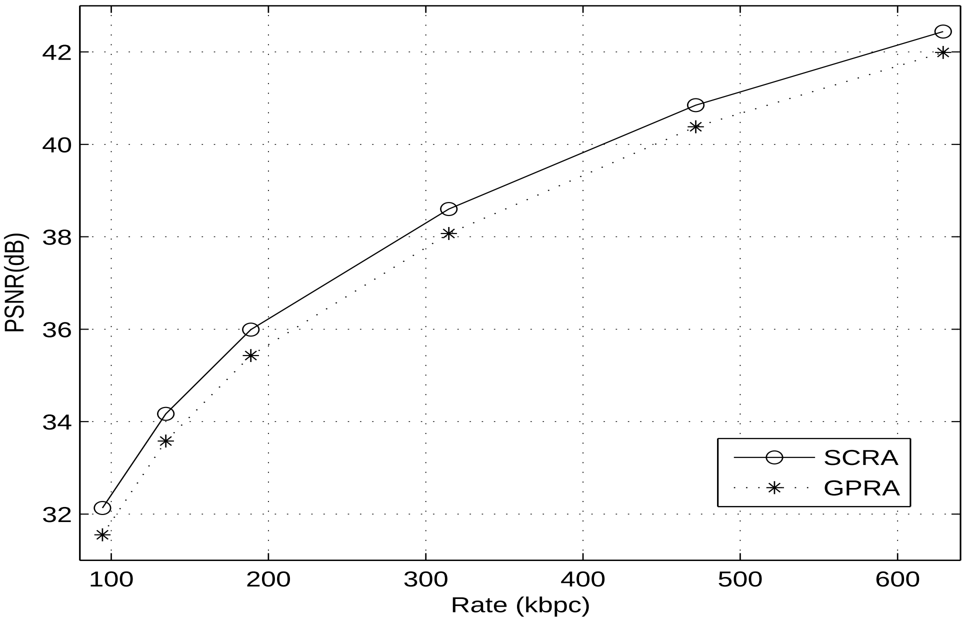


Fig. 4.16 Compression performance comparison of *SCRA* and *GPRA* with changing bitrates for the image dataset *Lab* encoded by *DC-DWTC*.

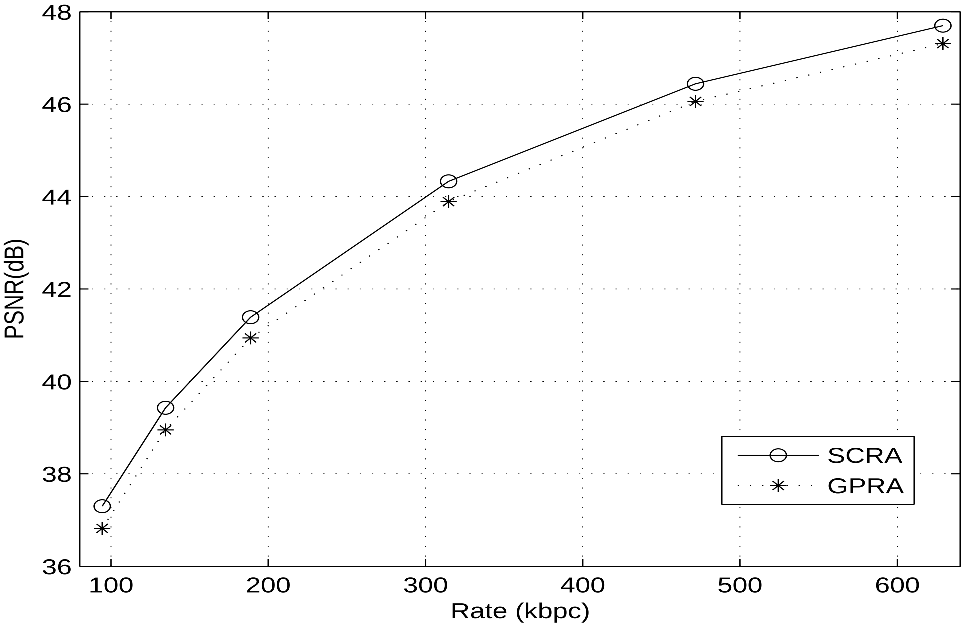


Fig. 4.17 Compression performance comparison of *SCRA* and *GPRA* with changing bitrates for the image dataset *Corridor* encoded by *DC-DWTC*.

Compression Ratio	Image Dataset	Comp.[a]	*PSNR* (dB) of *GPRA*	*PSNR* (dB) of *SCRA*	Gain (dB)
$\mathcal{R}_C = 200$	Lab	Y	31.55	32.13	+0.58
		U	40.57	40.99	+0.42
		V	39.01	39.41	+0.40
	Corridor	Y	36.82	37.30	+0.48
		U	42.62	42.97	+0.35
		V	41.80	42.20	+0.40
$\mathcal{R}_C = 100$	Lab	Y	35.43	35.99	+0.56
		U	42.85	43.26	+0.41
		V	41.48	41.92	+0.44
	Corridor	Y	40.94	41.39	+0.45
		U	45.55	45.88	+0.33
		V	44.91	45.32	+0.41
$\mathcal{R}_C = 60$	Lab	Y	38.07	38.60	+0.53
		U	44.30	44.68	+0.38
		V	43.33	43.74	+0.41
	Corridor	Y	43.89	44.33	+0.44
		U	47.54	47.91	+0.37
		V	47.11	47.53	+0.42
$\mathcal{R}_C = 30$	Lab	Y	41.99	42.44	+0.45
		U	46.98	47.30	+0.32
		V	46.31	46.66	+0.35
	Corridor	Y	47.31	47.70	+0.39
		U	50.67	50.96	+0.29
		V	50.48	50.84	+0.36

[a] Component

Table 4.2 Performance improvements of *DC-DWTC* with *SCRA* compared with *DC-DWTC* with *GPRA* applied to encoding cubic-panorama image datasets *Lab* and *Corridor* at $\mathcal{R}_C = 30, 60, 100, 200$.

The experimental results of the block-based hybrid compression scheme *BBHC* respectively with *SCRA* and *GPRA* applied to encoding the two testing cubic-panorama image datasets are presented in Fig. 4.18 and Fig. 4.19 in terms of the average PSNR of the luminance component **Y** versus the bitrate. In order to obtain compressed basis images with steady image quality, no bitrate control is used in this *BBHC* scheme. The frame quantization parameter Q_F is fixed for each basis image, and greater values of Q_F for the intra-coded basis images are chosen compared with the Q_F values of the predicted basis images to obtain compressed basis images with quality levels as similar as possible. Improved compression efficiency resulting from the use of *SCRA* compared with *GPRA* can be observed for both of the testing cubic panorama image datasets.

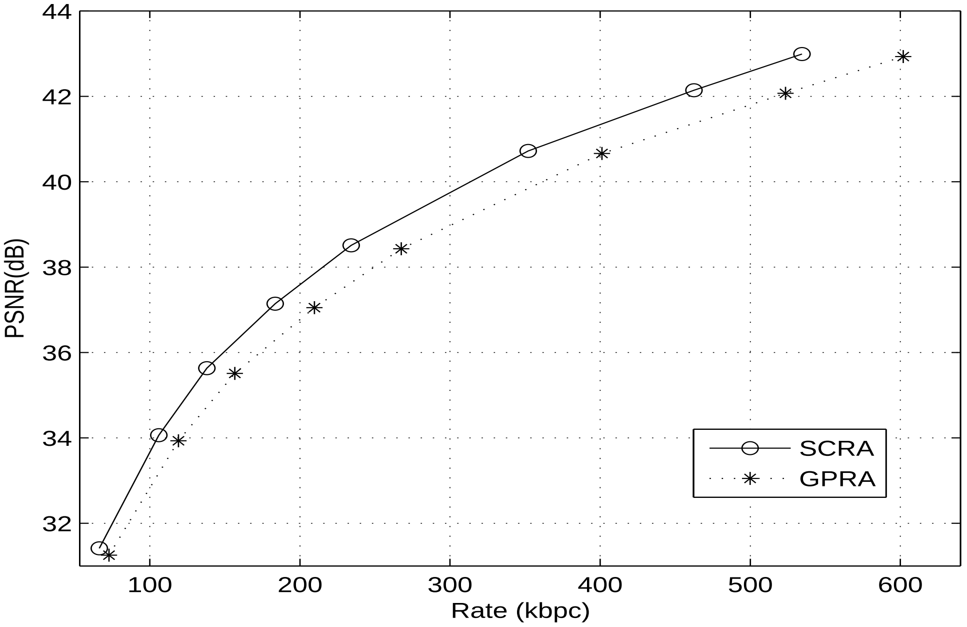


Fig. 4.18 Compression performance comparison of *SCRA* and *GPRA* with changing bitrates for the image dataset *Lab* encoded by *BBHC*.

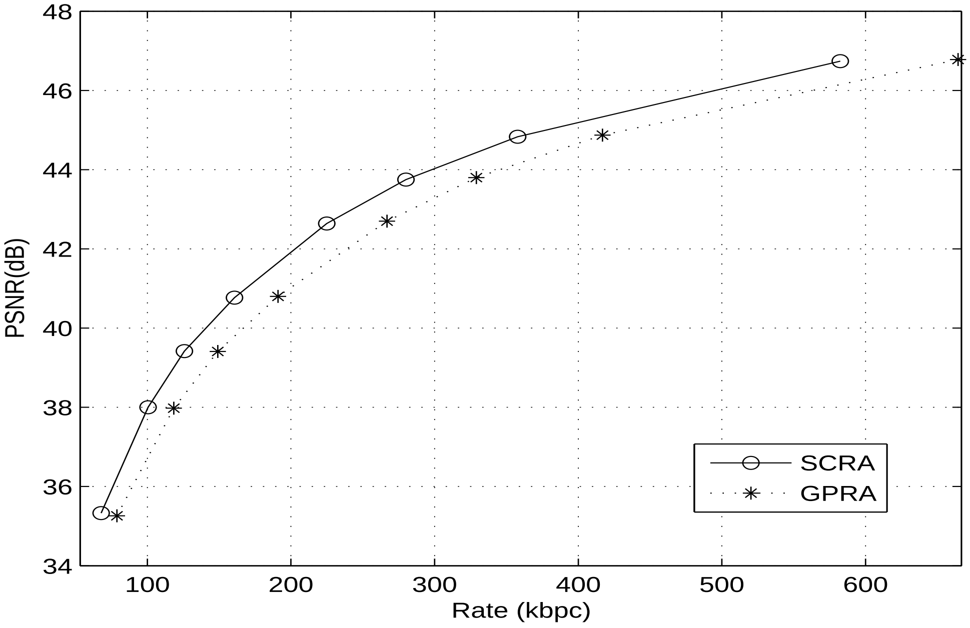


Fig. 4.19 Compression performance comparison of *SCRA* and *GPRA* with changing bitrates for the image dataset *Corridor* encoded by *BBHC*.

Tables 4.3 and 4.4 show the bitrate reductions of the *BBHC* scheme with *SCRA* over the *BBHC* scheme with *GPRA* at several different quality levels for the *Lab* image dataset and the *Corridor* image dataset respectively. It can be summarized that bitrate reductions of about 12% for the *Lab* image dataset and about 14% for the *Corridor* image dataset in the bitrate range R_b : 100 - 600 *kbpc* can be achieved with the use of *SCRA* compared with using *GPRA*.

Also, the coding performance of the *HHBC* scheme applied to the testing image datasets in terms of compression ratios is presented in Fig. 4.20 and Fig. 4.21. It shows that for the same levels of compression ratios, the average PSNR of the *HHBC* scheme with *SCRA* is obviously higher than that of the *HHBC* scheme with *GPRA*. A superior coding performance of *SCRA* for cubic-panorama image dataset compression is achieved over *GPRA*.

BBHC with **GPRA** **/SCRA**	PSNR **Y** (dB)	PSNR **U** (dB)	PSNR **V** (dB)	Bitrate (kbpc)	Bitrate Reduct.
GPRA	39.41	43.40	42.79	149.10	—
SCRA	39.42	43.45	42.87	125.76	15.66%
GPRA	42.70	45.73	45.30	266.90	—
SCRA	42.64	45.78	45.40	224.86	15.75%
GPRA	44.87	47.47	47.12	416.91	—
SCRA	44.83	47.47	47.18	359.63	13.74%
GPRA	46.78	49.63	49.42	664.43	—
SCRA	46.74	49.59	49.42	582.45	12.34%

Table 4.3 Bitrate reduction of the *BBHC* scheme with *SCRA* applied to encoding experimental cubic-panorama image dataset *Corridor*

BBHC with **GPRA** **/SCRA**	PSNR **Y** (dB)	PSNR **U** (dB)	PSNR **V** (dB)	Bitrate (kbpc)	Bitrate Reduct.
GPRA	37.05	42.62	41.31	209.66	—
SCRA	37.14	42.67	41.38	183.46	12.49%
GPRA	38.43	43.18	42.01	267.56	—
SCRA	38.51	43.24	42.08	234.11	12.50%
GPRA	40.66	44.22	43.22	401.22	—
SCRA	40.72	44.25	43.29	352.05	12.25%
GPRA	42.93	45.62	44.87	601.85	—
SCRA	42.99	45.67	44.92	534.48	11.19%

Table 4.4 Bitrate reduction of the *BBHC* scheme with *SCRA* applied to encoding experimental cubic-panorama image dataset *Lab*

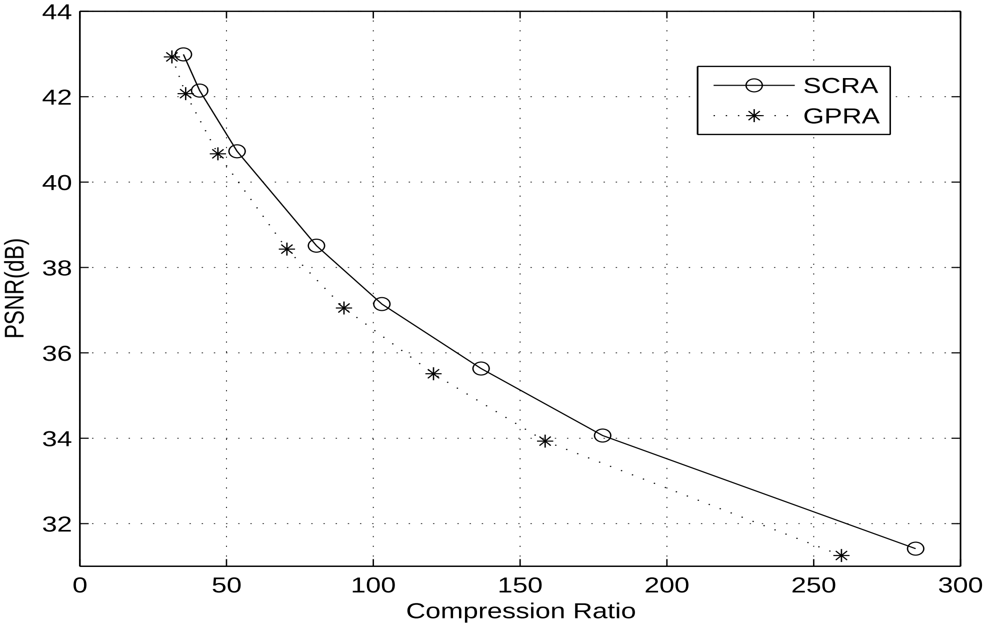


Fig. 4.20 PSNR versus compression ratio of *BBHC* with *SCRA* and *GPRA* applied for encoding the image dataset *Lab*.

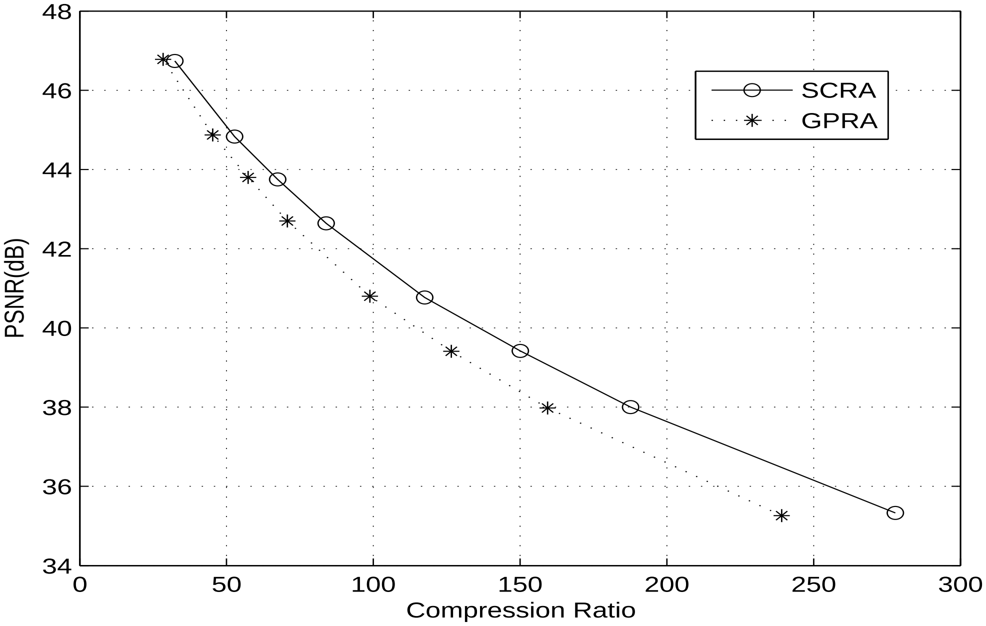


Fig. 4.21 PSNR versus compression ratio of *BBHC* with *SCRA* and *GPRA* applied for encoding the image dataset *Corridor*.

4.6 Summary

Efficient compression of cubic-panorama image datasets is crucial to reducing the required huge storage space and the high transmission bandwidth. In order to cope with the problem of coding efficiency sacrifice due to the inconsistent side-image connections in the conventional planar representation of cubic panoramas, a spatially consistent representation of cubic-panorama image datasets is proposed. With this spatially consistent representation applied to cubic-panorama image dataset compression, the constraints of side image boundaries and frame boundaries on the displacement vectors are eliminated. The search for displacement vectors as well as matching blocks in the reference basis images can be naturally extended across side image boundaries into neighbor side images in all directions. A specific block padding algorithm of reference block construction is desinged for displacement estimation and compensation. Optimized matching reference blocks can be obtained to reduce the prediction errors and improve the compression efficiency.

A displacement-compensated discrete wavelet transform coding (*DC-DWTC*) scheme and a block-based hybrid coding (*BBHC*) scheme are adapted to compressing cubic-panorama image datasets with the proposed spatially consistent representation. These two compression schemes are applied to encoding the testing cubic-panorama image datasets. Experimental results show that coding performance improvement of about 0.5 dB on average is obtained with the *DC-DWTC* scheme and more than 12% of bitrate reduction is obtained by utilizing the *BBHC* scheme. Superior compression performances are achieved with the spatially consistent representation of cubic-panorama image datasets compared with that of the generic planar representation of cubic-panorama image datasets.

Chapter 5

Scalable Lifted Wavelet-Based Scheme with Displacement Compensation for Cubic-Panorama Image Dataset Compression

Spatial-resolution scalable coding is extremely desirable for compressing image datasets used for image-based virtual environment navigation. As the synthesized zooming views, which are frequently required by users in exploring the image-based virtual environments, are generated by changing the field-of-view of the synthesized images and displaying them with the same spatial image resolution, basis images with different levels of spatial resolution are required to maintain stable quality of synthesized zooming images. In the standardized video coding schemes, scalable coding is recommended but is not practically applicable because of the high cost of reduced coding efficiency for incorporating scalable coding into the schemes. Discrete wavelet transform *(DWT)* coding techniques are characterized by their potential to provide temporal, spatial and image quality or *SNR* scalabilities in addition to their high compression performance, and are suitable for the purpose of image dataset compression with the spatially scalable requirement. However, it poses some challenges to develop a wavelet-based coding scheme for image dataset compression as a more complicated tradeoff among the coding efficiency, the spatial-resolution scalability and the random access and selective decoding flexibility should be made. More sophisticated compression techniques need to be applied in the design of a spatial-resolution scalable wavelet-based coding scheme while keeping high compression efficiency and facilitating required random access and selective decoding.

Motivated by the breakthrough and further developments in the research on motion-compensated three-dimensional wavelet transforms applied to video sequence compression [22, 56, 64, 66], a scalable lifted wavelet-based coding scheme with cross-image displacement-compensated filtering is developed for cubic-panorama image dataset compression with the spatial-resolution scalability requirement. In this wavelet-based compression scheme, emphasis is put on achieving the desired spatial-resolution scalability in a highly efficient coding scheme while supporting the required random access and selective decoding by adapting the coding scheme to cubic-panorama image dataset compression. Displacement estimation and compensation are incorporated into the cross-image filtering process to enhance the compression efficiency of the wavelet transform across basis images in the image dataset. Both the cross-image wavelet transform with displacement compensation and the spatial wavelet transform are factorized into lifting operations and thus fast, memory-saving and in-place wavelet analysis and synthesis are achieved.

A new embedded entropy coding approach named independent block with layered data partition *(IBLDP)* is proposed and combined with the displacement-compensated lifted wavelet transforms to form an efficient image dataset coding scheme. Unlike the widely-applied *EZW* [73] and *SPIHT* [69] wavelet entropy coding approaches, *IBLDP* does not rely on exploiting inter-subband redundancy to improve coding efficiency and consequently removes the restriction of inter-subband coding against resolution-scalable coding. Different from most wavelet entropy coding approaches like *SPECK* [32] and *EZBC* [27] which put all decomposition coefficients of a decomposed image in a subband into a coding dependency chain, *IBLDP* partitions the decomposition coefficients in a subband into independently coded coefficient blocks and makes spatial random access to image segments of basis images possible. Also, the independent coefficient block coding improves error resilience performance of the assembled output bit-stream, easily supports efficient and fast parallel processing of coded basis images and provides the flexibility of organizing code streams of coefficient blocks in different orders in the output bit-stream to satisfy the requirements of various applications. With the compact layered bitplane data representation and simplified coding structure, *IBLDP* features reduced computational and implemental complexity and hence is more suitable for the efficient and interactive image rendering application.

In this scalable displacement-compensated lifted wavelet-based compression scheme, a rectangular subregion side-image random access mechanism is provided for cubic-panorama image dataset compression used for image-based virtual environment navigation. A specific hierarchical data structure is designed to accommodate the spatial image random access and maintain the coding performance at high levels. The corresponding bit-stream syntax supporting this data structure is formed to provide efficient selective decoding for interactive image rendering. Multi-level index tables are embedded in the bit-stream to assist spatial image random access. This wavelet-based image dataset compression scheme is characterized by a suitable compromise between the coding efficiency and the random access flexibility. It reduces the decoding delay, simplifies the implementation of a random access mechanism and easily facilitates fast interactive image decoding. It can well meet the requirements for image dataset compression used in image-based virtual environment navigation.

5.1 Introduction

Spatial-resolution scalability is a key requirement for compressing image datasets used in image-based virtual environment navigation. In exploring the image-based virtual environments, the most common user behavior, next to panning and tilting with the virtual camera, is to take a close shot or a long shot with it. When, for example, a user wants to zoom into the scene and get a close-up image of it, higher spatial-resolution basis images should be available to synthesize the close-up images with stable image quality.

Scalable coding is recommended in block-based hybrid video coding schemes in several international video coding standards [36, 37, 40, 41]. However, the recommended scalable coding is seldom applied in practice because of the high expense of the sacrificed compression performance for implementing scalable coding in these standard compression schemes. Wavelet-based coding techniques provide an effective and efficient alternative approach for image dataset compression. The ability of providing temporal, spatial and image quality or *SNR* scalabilities in addition to the high compression efficiency makes wavelet-based coding techniques more attractive to some applications with scalable coding requirements. As the spatial-resolution scalability feature of wavelet-based coding techniques is particularly needed in image-based virtual environment navigation, a scalable wavelet-based coding scheme is adopted here for cubic-panorama image dataset compression with the spatial-resolution scalable requirement.

A conventional wavelet-based compression scheme comes with a complex implementation structure due to the use of filter banks, and the wavelet transform coefficient decomposition and reconstruction are not computationally efficient enough by the traditional convolution methods. The development of the lifting schemes [81] greatly alleviates this problem. It leads to faster, memory-saving, in-place wavelet decomposition and reconstruction and makes wavelet-based compression techniques more feasible for practical applications. For this reason, the lifted wavelet transform is adopted in JPEG 2000 [34, 50] as a standard coding technique for still image compression. On the other hand, conventional wavelet-based compression schemes are not able to efficiently exploit the redundancy in the cross-image direction if directly applied to inter-frame image sequence compression. This prevented wavelet-based compression schemes from achieving the desired high coding efficiency in video sequence compression for supporting scalable coding. The breakthrough and developments in the research on wavelet transforms with motion-compensated temporal filtering *(MCTF)* for video sequence compression [22, 64, 66] solve this problem and make wavelet-based coding schemes more attractive to image dataset compression with the required coding scalability.

MCTF combines motion compensation with 3D wavelet transforms applied to video sequence coding, facilitating the desired scalable coding and maintaining high compression efficiency. The original research on *MCTF* can be tracked in Ohm's work [63]. He presented his study on the concepts of temporal-axis decomposition along the motion trajectory and deeply investigated the issue of temporal subband signal encoding. Choi and Woods [14] applied a temporal Haar wavelet transform in a motion-compensated 3D subband coding *(MC-3DSBC)* scheme. They improved unconnected pixel processing and proposed a bit-stream optimization technique. Hsiang and Woods [27] also used a Haar wavelet transform in the temporal wavelet decomposition and incorporated the invertible motion compensation with half-pixel accuracy into *MC-3DSBC*. For further improving the coding efficiency, context-based modeling and embedded zero-block coding *(EZBC)* were also introduced into the coding scheme combined with the invertible motion compensation, and the coding scheme named motion-compensated *EZBC* (*MC-EZBC*) was developed in [28]. Later, backward block motion estimation and compensation were added to *MC-EZBC* by Chen and Woods [12], and the wavelet transform analysis and synthesis procedures were factorized into lifting steps to optimize the wavelet transform processing.

It presents some difficulties to adapt the wavelet-based coding scheme with motion-compensated temporal filtering for generic video sequence coding to image dataset compression and make a more complicated tradeoff among the spatial-resolution scalability, the coding efficiency and the random access and selective decoding flexibility. Some compression schemes for image datasets have been proposed based on wavelet transforms. Luo et al. [55] presented a wavelet-based compression scheme for coding concentric mosaic image datasets. To support random access to the coded image dataset and minimize the relevant computational expense, Wu et al. [95] proposed the progressive inverse wavelet synthesis *(PIWS)* scheme. As an effort to further exploit the cross-image redundancy and improve the coding efficiency of the wavelet-based compression scheme, a smart rebinning scheme is presented in [96]. Girod et al. [23] applied wavelet transforms for coding light field image datasets. They developed a novel approach with a disparity-compensated lifting scheme. However, none of these proposed schemes were designed for compressing cubic-panorama image datasets. Although some comments on spatially scalable coding for image dataset compression were made in some developed image dataset compression schemes, no effective spatially scalable coding scheme was demonstrated. It poses a challenge to conduct deep research into the relevant issues and adapt the wavelet-based coding scheme to a new application for compressing cubic-panorama image datasets.

All these motivate our work on a wavelet-based coding scheme based on the framework of motion-compensated temporal filtering with lifting operations to accommodate spatial-resolution scalable coding with high compression efficiency and effective random access and selective decoding for cubic-panorama image datasets used in image-based virtual environment navigation. With the assumption that static environments are under discussion, the temporal filtering in video sequence compression is replaced by cross-image filtering in the image dataset compression. The displacement compensation to enhance the cross-image coding efficiency of the wavelet-based coding scheme for image dataset compression takes the place of motion compensation in video sequence compression. Thus, this newly-developed scheme is called the scalable lifted wavelet-based coding scheme with cross-image displacement-compensated filtering.

5.2 Discrete Wavelet Transforms with Lifting Schemes

Discrete wavelet transforms are powerful tools for image and video sequence compression due to their capability of efficient image and video sequence representations with inherent scalabilities. A wavelet transform is a linear operation on image samples to represent images in a compact form, which can be described as

$$\mathbf{C_w} = \mathcal{T}_{\mathcal{W}}\{\mathbf{I_x}\} = \mathbf{W}\,\mathbf{I_x} \tag{5.1}$$

where $\mathcal{T}_{\mathcal{W}}$ denotes the wavelet transform operation, $\mathbf{I_x}$ is a column vector of image samples, $\mathbf{W}$ is a matrix representing the wavelet transform and $\mathbf{C_w}$ is a column vector of wavelet transform coefficients. Wavelet transforms exploit the image redundancy over a larger scale and eliminate the problem of block artifacts at low bitrates. They have proved to be effective and efficient for image and video sequence compression. Combined with embedded entropy coding, wavelet transform compression schemes are highly scalable in temporal rates, spatial resolution, bitrates and image quality or *SNR*.

Lifting schemes for wavelet analysis and synthesis developed in [81] significantly enhance the performance of wavelet-based compression schemes and extend their applications once limited by the complex implementation of the wavelet transform due to the use of filter banks. Wavelet transforms are factorized into lifting operation structures for fast and memory-saving computation.

5.2.1 Wavelet Transform Kernels

As shown in Fig. 5.1, a one-dimensional wavelet transform can be well described by a two-channel perfect reconstruction (PR) filter bank [87]. The first part of the structure representing the forward wavelet transform is an analysis filter bank consisting of a low-pass filter (LPF) $\mathbf{h}_0(n)$ and a high-pass filter (HPF) $\mathbf{h}_1(n)$ with a down-sampling operation by a factor of two following each of the two analysis filters. The second part of the structure includes a synthesis filter bank comprising a synthesis LPF $\mathbf{g}_o(n)$ and a synthesis HPF $\mathbf{g}_1(n)$ with an up-sampling operation by a factor of two before each of the two synthesis filters. This corresponds to an inverse wavelet transform.

For the perfect reconstruction of the input signal $X(n)$ by $\hat{X}(n)$, the Z-transforms

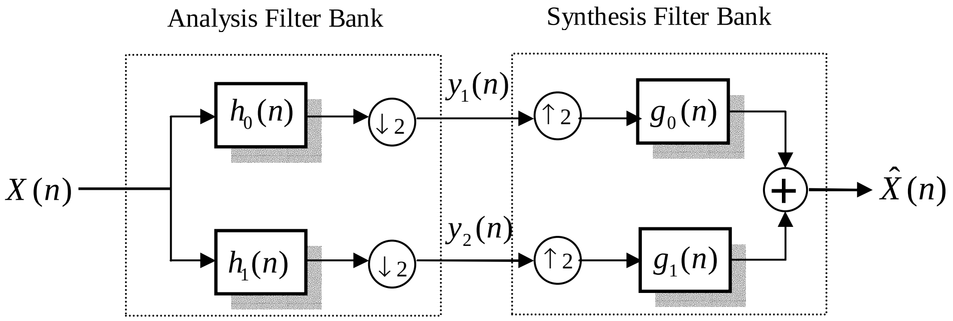


Fig. 5.1 One-dimensional two-channel wavelet perfect reconstruction filter bank consisting of an analysis filter bank and a synthesis filter bank.

$\mathbf{H}_0(z)$, $\mathbf{H}_1(z)$ of the analysis filters $\mathbf{h}_0(n)$, $\mathbf{h}_1(n)$ and the Z-transforms $\mathbf{G}_0(z)$, $\mathbf{G}_1(z)$ of the synthesis filters $\mathbf{g}_0(n)$, $\mathbf{g}_1(n)$ need to satisfy the following equations

$$\begin{aligned} \mathbf{H}_0(z)\mathbf{G}_0(z) + \mathbf{H}_1(z)\mathbf{G}_1(z) &= 2, \\ \mathbf{H}_0(-z)\mathbf{G}_0(z) + \mathbf{H}_1(-z)\mathbf{G}_1(z) &= 0. \end{aligned} \tag{5.2}$$

In order to remove the aliasing component, the synthesis filters are derived from the analysis filters based on the second equation in (5.2), resulting in

$$\mathbf{G}_0(z) = -Sz^{-D}\mathbf{H}_1(-z), \quad \mathbf{G}_1(z) = Sz^{-D}\mathbf{H}_0(-z), \tag{5.3}$$

where S is a scaling factor determining the normalization of the wavelet kernel, and D is an integer constant indicating the delay of the analysis filter bank and the synthesis filter bank.

For the cross-image wavelet transform, the length of the cross-image filter is a very important consideration. Shorter kernel wavelet transforms are more suitable for achieving the required low delay and facilitating flexible random access to and efficient selective decoding of the coding bit-stream. Due to this consideration, we apply the

Haar wavelet kernel with normalization **(1, 1)** for cross-image wavelet transform. The one-dimensional Haar wavelet kernel is specified as

$$\begin{aligned} \mathbf{H}_0(z) &= \tfrac{1}{2}(1+z^{-1}), \\ \mathbf{H}_1(z) &= \tfrac{1}{2}(1-z^{-1}). \end{aligned} \tag{5.4}$$

Alternatively, the bi-orthogonal LeGall 5/3 wavelet kernel [51] can be used as a relatively longer wavelet transform for cross-image filtering. Improved compression efficiency can be obtained by using longer wavelet kernels at the cost of sacrificed random access and selective decoding flexibility. The LeGall 5/3 wavelet kernel of normalization **(1, 2)** corresponds to

$$\begin{aligned} \mathbf{H}_0(z) &= \tfrac{1}{8}(-1+2z^{-1}+6z^{-2}+2z^{-3}-z^{-4}), \\ \mathbf{H}_1(z) &= \tfrac{1}{2}(1-2z^{-1}+z^{-2}). \end{aligned} \tag{5.5}$$

The wavelet transform kernel for the two-dimensional spatial discrete wavelet analysis and synthesis is the separable bi-orthogonal Daubechies 9/7 proposed in [4] with its high image compression performance. The Daubechies 9/7 wavelet analysis is applied to the cross-image decomposition coefficients following the displacement-compensated cross-image wavelet analysis. The coefficients of the one-dimensional analysis filters of the separable two-dimensional Daubechies 9/7 kernel with the **(1, 2)** normalization for the spatial wavelet analysis are presented in Table 5.1.

5.2.2 Cross-Image and Spatial Wavelet Analysis

With the Haar wavelet kernel used for the cross-image wavelet analysis, two consecutive basis images are combined to generate one high-pass subband image and one low-pass subband image (Fig. 5.2(a)). Here **BI** designates the basis images, and **HP** and **LP** denote the decomposition high-pass images and low-pass images respectively in the figure. To further remove the remaining redundancy, Mallat dyadic or pyramid

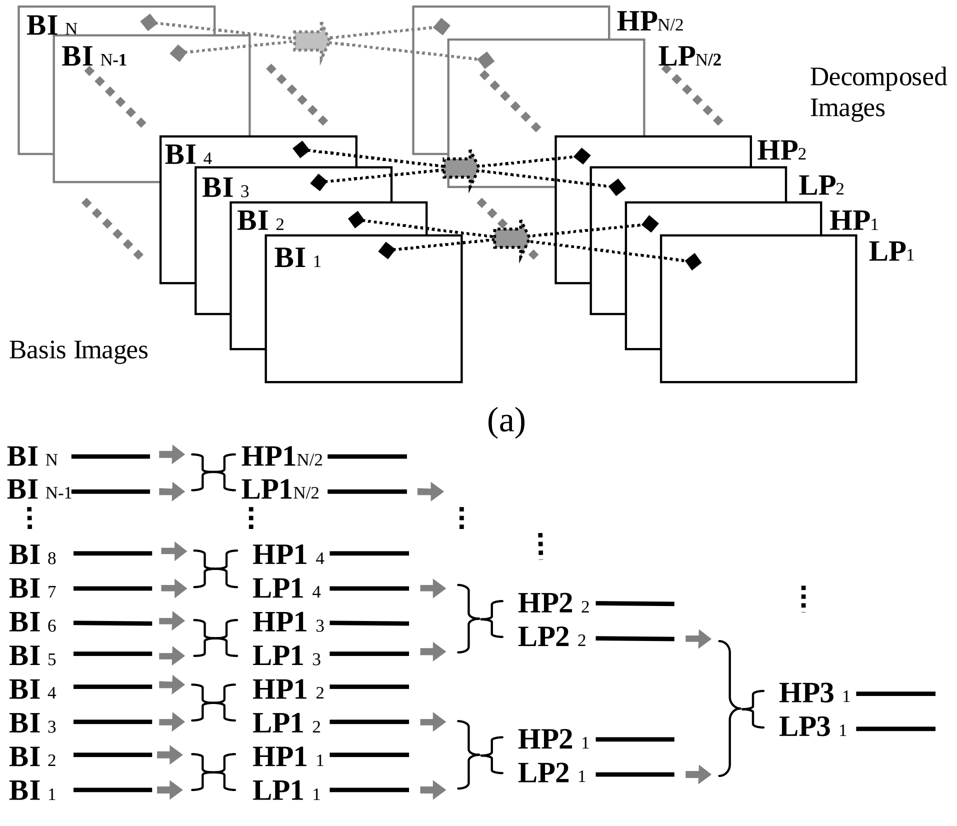
n	LPF $h_0(n)$	n	HPF $h_1(n)$
0	0.602949018236360	-1	1.115087052457000
-1, 1	0.266864118442875	-2, 0	-0.591271763114250
-2, 2	-0.078223266528990	-3, 1	-0.057543526228500
-3, 3	-0.016864118442875	-4, 2	0.091271763114250
-4, 4	0.026748757410810		

Table 5.1 Analysis filter coefficients of the Daubechies 9/7 wavelet transform kernel applied for the spatial wavelet transform

analysis is used by applying additional Haar wavelet analysis on the consecutive low-pass images resulting from the previous level of Haar wavelet analysis. A three-level cross-image Mallat dyadic Haar wavelet transform is depicted in Fig. 5.2(b) as an example of multilevel cross-image wavelet analysis. Mallat dyadic wavelet analysis is more efficient than other multilevel analysis structures as little improvement can be achieved if additional wavelet analysis is applied on the high-pass images.

For alternatively using the LeGall 5/3 wavelet kernel for cross-image wavelet analysis, the dependent relationship between the basis images and the resulting low-pass and high-pass images is more complex than that of using the Haar wavelet kernel. For example, a low-pass image at the first level of wavelet analysis is derived based on five basis images, and a high-pass image based on three basis images.

The 2D separable spatial Daubechies 9/7 wavelet analysis is performed in two steps on each decomposed image of spatial resolution $M \times N$ resulting from the cross-image analysis process (Fig. 5.3(a)). Firstly, a horizontal 1D Daubechies 9/7 wavelet analysis is applied on each row of the decomposed image, resulting in a low-pass image or subband **L** and a high-pass image or subband **H** of resolution $M/2 \times N$. Secondly, a vertical 1D Daubechies 9/7 wavelet analysis is applied on each column of the resultant low-pass and high-pass subbands **L** and **H**, resulting in four subbands **LL, LH, HL**



(a)

(b)

Fig. 5.2 One-dimensional two-band cross-image Haar wavelet analysis. (a) One-level cross-image Haar wavelet analysis. (b) Three-level cross-image Mallat dyadic wavelet analysis.

and **HH** of decomposition coefficients with halved resolution in both the horizontal and vertical directions: $M/2 \times N/2$. Also, Mallat dyadic Daubechies 9/7 is utilized for multilevel spatial wavelet analysis in the low subbands in order to further decorrelate the decomposition coefficients. As an example of multilevel spatial wavelet analysis, a three-level Mallat dyadic wavelet transform is depicted in Fig. 5.3(b). With $\mathbf{N_L}$ levels of spatial wavelet analysis, the basis images can be represented at $\mathbf{N_L}$+1 different spatial resolution levels. Thus, the desired spatial resolution scalability can be achieved.

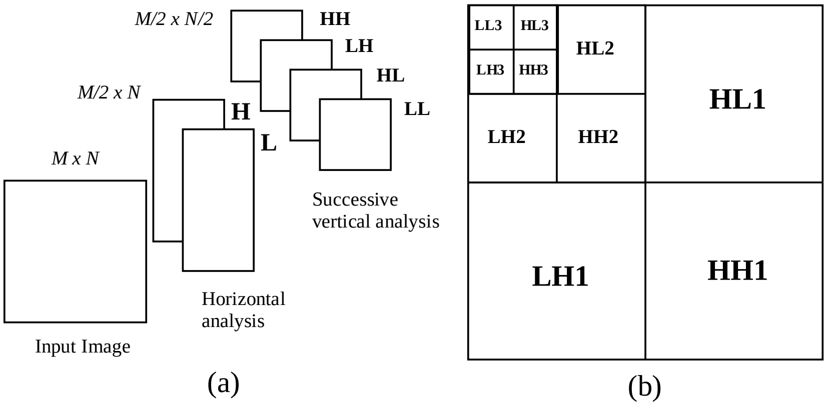


Fig. 5.3 Two-band 2D separable spatial wavelet analysis. (a) One-level 2D separable spatial wavelet analysis. (b) Multi-level 2D Mallat dyadic spatial wavelet analysis.

5.2.3 Lifted Wavelet Transform Operations

The lifting scheme of a wavelet transform mainly comprises a series of alternating prediction and update operations as shown in Fig. 5.4. The trivial transform partitions the input signal into an even-number indexed input sequence and an odd-number indexed input sequence, and feeds them into the upper port and the lower port respectively. A prediction operation $P_i(z)$ $(i \in \{1, 2, ..., D\}, D \in \mathbb{Z})$ uses a linear combination of the neighboring even-number indexed inputs to estimate an odd-number indexed input and is followed by a subtraction of this estimate from the odd-number indexed input to obtain a prediction error. An update operation $U_i(z)$ $(i \in \{1, 2, ..., D\})$ uses a linear combination of the modified prediction errors for updating an even-number indexed input and is followed by an addition of this value to the even-number indexed input. These prediction and update operations form a ladder structure of the lifted wavelet transform. The number D of the prediction and update steps contained in a lifted wavelet transform structure depends on the wavelet transform kernel in use. The output of the adder following the last update operation is the low-pass output after scaled by a constant factor G_L, and the output of the adder following the last prediction operation is the high-pass output after scaled by a constant factor G_H.

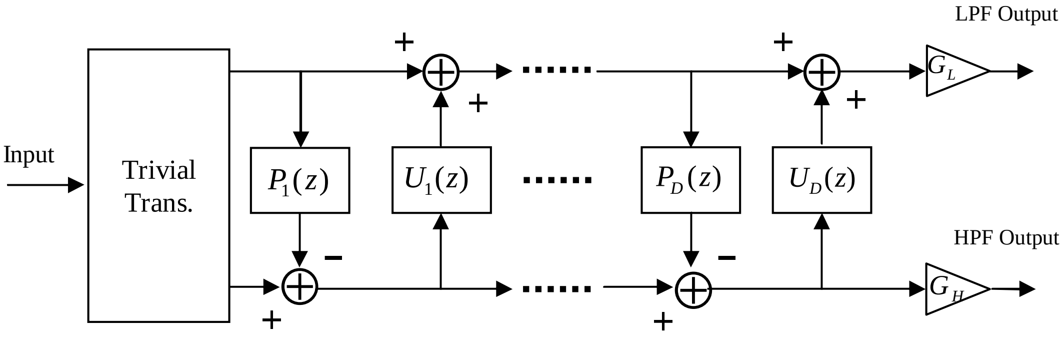


Fig. 5.4 Structure of a wavelet transform factorized into a lifting scheme with a series of prediction and update operations.

The lifted wavelet transform scheme alleviates the problem of complex filter banks in traditional wavelet transforms. It features memory saving and in-place computation. Both forward wavelet analysis and inverse wavelet synthesis can be factorized into the fast lifting operations. Moreover, the lifted wavelet transform structure allows the incorporation of other required operations in the prediction-update steps. All wavelet kernels can be put into their corresponding lifting structures.

For the cross-image Haar wavelet analysis, the high-pass subband decomposition image $S_{H,\,k}[\bar{\mathbf{x}}]$ $(k \in \{1, 2, ..., N/2\}$, N is the number of the basis images in the image dataset) corresponds to the normalized difference of two consecutive basis images, and the low-pass subband decomposition image $S_{L,\,k}[\bar{\mathbf{x}}]$ corresponds to the average of the two basis images. Mathematically, the cross-image Haar wavelet analysis factorized into the lifting operations is given by

$$\begin{aligned} S_{H,\,k}[\bar{\mathbf{x}}] &= \tfrac{1}{2}\big(\mathcal{I}_{\mathcal{B},\,2k}[\bar{\mathbf{x}}] - \mathcal{I}_{\mathcal{B},\,2k-1}[\bar{\mathbf{x}}]\big), \\ S_{L,\,k}[\bar{\mathbf{x}}] &= \mathcal{I}_{\mathcal{B},\,2k-1}[\bar{\mathbf{x}}] + S_{H,\,k}[\bar{\mathbf{x}}], \end{aligned} \tag{5.6}$$

where $\mathcal{I}_{\mathcal{B},\,2k}[\bar{\mathbf{x}}]$ and $\mathcal{I}_{\mathcal{B},\,2k-1}[\bar{\mathbf{x}}]$ are two consecutive basis images.

With the bi-orthogonal LeGall 5/3 wavelet transform, the cross-image lifting operations are expressed as following

$$\begin{aligned} S_{H,\,k}[\bar{\mathbf{x}}] &= \mathcal{I}_{\mathcal{B},\,2k+1}[\bar{\mathbf{x}}] - \tfrac{1}{2}\big(\mathcal{I}_{\mathcal{B},\,2k}[\bar{\mathbf{x}}] + \mathcal{I}_{\mathcal{B},2k+2}[\bar{\mathbf{x}}]\big), \\ S_{L,\,k}[\bar{\mathbf{x}}] &= \mathcal{I}_{\mathcal{B},\,2k}[\bar{\mathbf{x}}] + \tfrac{1}{4}\big(S_{H,\,k-1}[\bar{\mathbf{x}}] + S_{H,\,k}[\bar{\mathbf{x}}]\big). \end{aligned} \tag{5.7}$$

The lifted bi-orthogonal LeGall 5/3 for cross-image wavelet analysis can be intuitively presented in a data flow chart as shown in Fig. 5.5(a). The lifting operation of the LeGall 5/3 forward wavelet analysis consists of two steps of prediction and update operations. Based on the number of input samples, half number of nodes is calculated in each step. An elementary lifting operation unit includes one multiplication and two additions (Fig. 5.5(b)). The computation of the elementary lifting unit refers to

$$s = t + w_c \times (u + v) \tag{5.8}$$

where t, u and v are the inputs or intermediate data, w_c is a lifting coefficient and s is the output or intermediate data. The input samples, intermediate results and decomposition coefficients can all be stored in the same memory unit for memory saving and this easily enables an in-place computation. The implementation of the wavelet synthesis process with lifting operations can be obtained by straightforward inversion of the data flow.

For the spatial wavelet transform, one-dimensional forward bi-orthogonal Daubechies 9/7 wavelet analysis is factorized into the lifting operation as depicted in Fig. 5.6. On the left side, the input decomposition coefficients resulting from the cross-image analysis are represented by c_n ($n \in \{1, 2, ..., 2i{-}1,\ 2i,\ 2i{+}1, ..., N_c\}$); N_c is the number of decomposition coefficients. The input decomposition coefficients are extended periodically and symmetrically beyond image boundaries to ensure that the lifting operation is working properly at the boundaries. The lifting operations of the Daubechies 9/7 forward wavelet analysis consist of four steps. w_j ($j \in \{1, 2, 3, 4\}$) denotes the lifting coeffi-

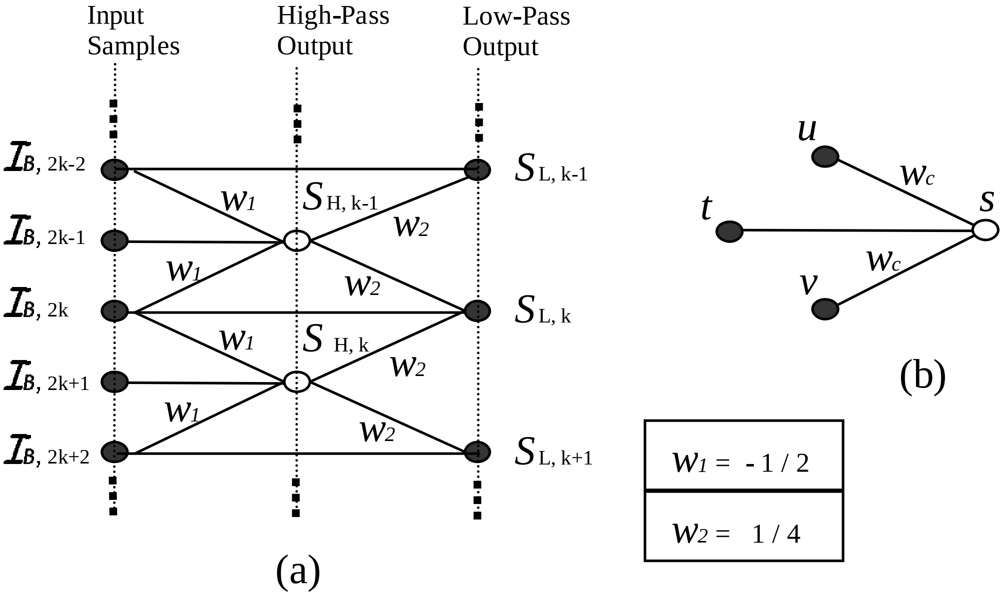


Fig. 5.5 Lifting operation of the bi-orthogonal LeGall 5/3 forward DWT. (a) LeGall 5/3 analysis operations. (b) An elementary lifting operation unit.

cients. The high-pass decomposition coefficients $c_{H,l}$ ($l \in \{1, 2, ..., i{-}1,\ i,\ i{+}1, ..., N_c/2\}$) and the low-pass decomposition coefficients $c_{L,m}$ ($m \in \{1, 2, ..., i-1,\ i,\ i+1, ..., N_c/2\}$) are output on the right side. The output high-pass coefficients $c_{H,l}$ are modified by a weight $G_H = 1.230174105$, and the low-pass coefficients $c_{L,m}$ are modified by a weight $G_L = 1/G_H$. The corresponding lifted Daubechies 9/7 wavelet synthesis can be carried out by inverting the data flow.

5.3 Lifted Displacement-Compensated Cross-Image Filtering

Cross-image filtering is less efficient when directly applying the wavelet transform across basis images in the image datasets, as the image displacements decorrelate the image samples at the same spatial locations in the consecutive basis images. To alleviate this problem, efforts at improving the coding efficiency for image dataset compression

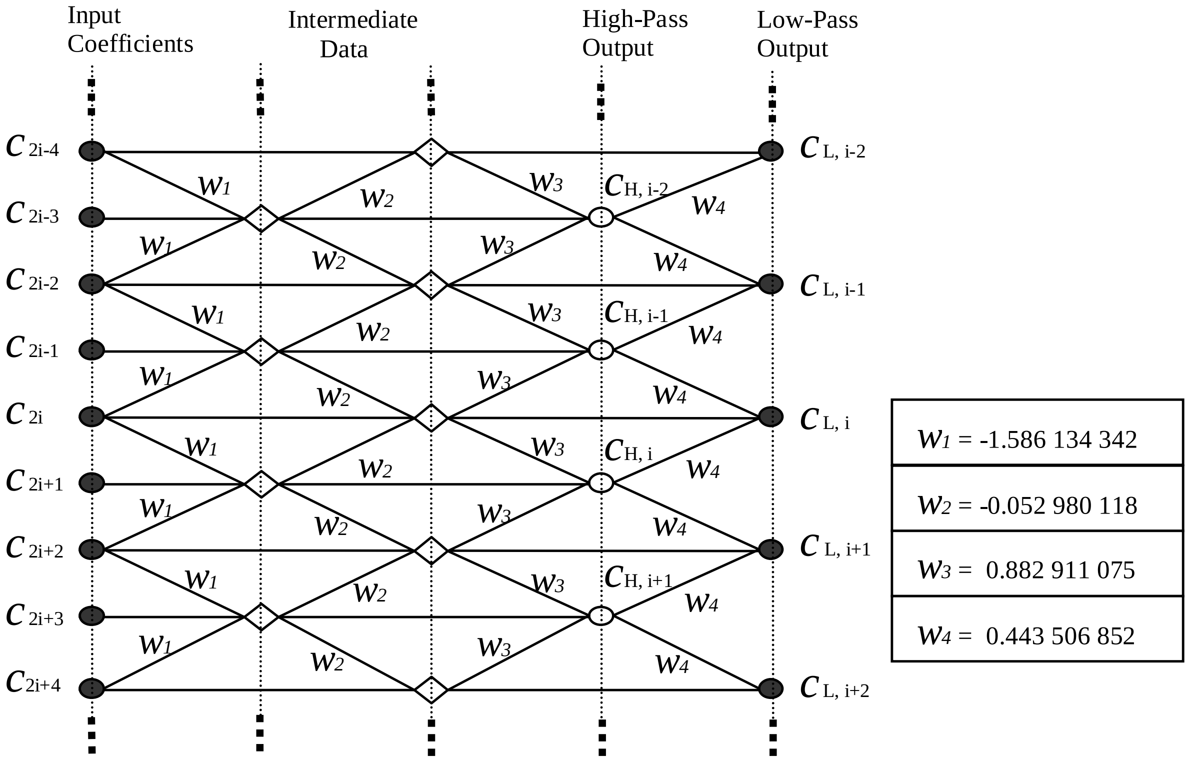


Fig. 5.6 Lifting operation of the 1D bi-orthogonal Daubechies 9/7 forward DWT for 2D separable spatial wavelet analysis.

were made in some developed coding schemes by the smart image re-binning approach [96], the image reorganization method [74], the image alignment scheme [55] or more generally through arbitrary frame warping operations [70]. However, the improvements on compression efficiency in the cross-image direction were quite limited by these measures. In addition, these measures are not effective for more complex displacement field distributions, such as the contraction and expansion displacement field distributions, and they bring about no further improvement in coding performance with longer wavelet kernels. The developments of motion-compensated temporal filtering *(MCTF)* under the framework of wavelet transform coding schemes [22, 64, 66] overcome the limitations of these measures. Adapting wavelet transforms enhanced by *MCTF* to compressing image datasets makes wavelet-based coding schemes more attractive to image dataset compression with the required coding scalability for image-based virtual environment navigation.

5.3.1 Displacement-Compensated Wavelet Transform

Displacement compensation is a successful technique to efficiently exploit the cross-image redundancy of the basis images in image datasets and significantly improve the coding performance of the compression schemes. It aligns the image samples along the displacement trajectory and enhances the potential of removing large amount of cross-image redundancy in the following wavelet analysis process. In order to compensate the image displacement between adjacent basis images in the image datasets, the displacement compensation makes the appropriate spatial coordinate shift based on the displacement vector distribution obtained in the displacement estimation procedure.

The displacement-compensated wavelet transform is an adaptive method in nature. Each image block or even each image sample can be assigned a displacement vector to adapt to the local image feature. The displacement compensation can be combined with a variety of wavelet transform kernels ranging from the Haar wavelet to long wavelet kernels for cross-image filtering. Any displacement model for representing and estimating displacement distributions can be incorporated into the displacement-compensated wavelet transform schemes. The displacement-compensated wavelet transform becomes a non-linear operation as a result of the combination of the non-linear displacement compensation with the linear wavelet transform.

5.3.2 Cross-Image Filtering with Lifting Operations

Displacement-compensated cross-image filtering with lifted wavelet transforms was initially applied to image sequence compression in [67] and [70]. The structure of the lifted wavelet transform with displacement compensation is shown in Fig. 5.7.

The non-linear displacement compensations are incorporated into the framework of the lifted wavelet transform scheme. DC_i^P and DC_i^U ($i \in \{1, 2, ..., D\}$, $D \in \mathbb{Z}$), separately denoting the displacement compensation for the prediction operation and the update operation, are respectively added before each prediction and update operation. The lifted displacement-compensated wavelet transform structure consists of a series of displacement-compensated alternating prediction and update operations.

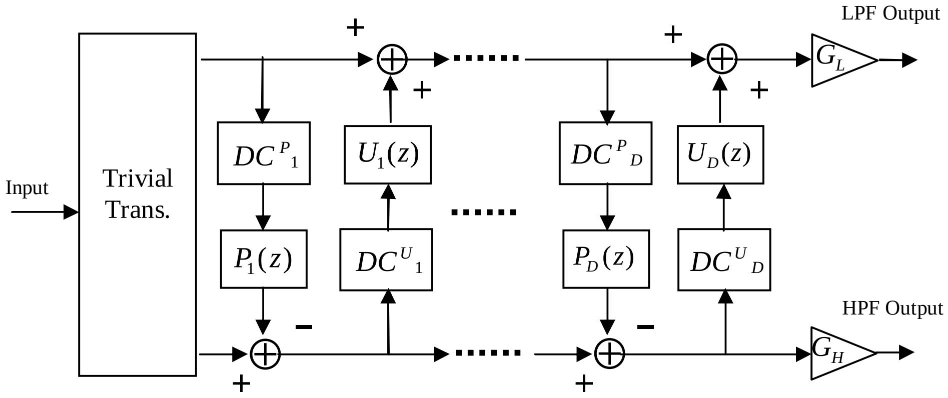


Fig. 5.7 Lifted displacement-compensated wavelet transform consisting of a series of displacement-compensated prediction and update operations.

The Haar wavelet transform with displacement compensation for cross-image analysis refers to the displacement-compensated normalized difference and average of two consecutive basis images. It can be mathematically represented as [70]

$$\begin{aligned} S_{H,\,k}[\bar{\mathbf{x}}] &= \tfrac{1}{2}\big(\mathcal{I}_{\mathcal{B},\,2k}[\bar{\mathbf{x}}] - D_{2k-1\to 2k}(\mathcal{I}_{\mathcal{B},\,2k-1})[\bar{\mathbf{x}}]\big), \\ S_{L,\,k}[\bar{\mathbf{x}}] &= \mathcal{I}_{\mathcal{B},\,2k-1}[\bar{\mathbf{x}}] + D_{2k\to 2k-1}(S_{H,\,k})[\bar{\mathbf{x}}], \end{aligned} \tag{5.9}$$

where $D_{2k-1\to 2k}$ designates the displacement-compensated mapping from basis image $\mathcal{I}_{\mathcal{B},2k-1}[\bar{\mathbf{x}}]$ to the coordinate system of the basis image $\mathcal{I}_{\mathcal{B},2k}[\bar{\mathbf{x}}]$, and $D_{2k\to 2k-1}$ designates the backward displacement-compensated mapping from basis image $\mathcal{I}_{\mathcal{B},2k}[\bar{\mathbf{x}}]$ to the coordinate system of the basis image $\mathcal{I}_{\mathcal{B},\,2k-1}[\bar{\mathbf{x}}]$. For the simple case of Haar wavelet, separate forward and backward displacement models can be used for displacement compensation.

For the bi-orthogonal LeGall 5/3 wavelet transform with displacement compensation, the multi-reference displacement-compensated forward transform in lifting operations is mathematically given by

$$\begin{aligned} S_{H,\,k}[\bar{\mathbf{x}}] &= \mathcal{I}_{\mathcal{B},\,2k+1}[\bar{\mathbf{x}}] - \tfrac{1}{2}\big(D_{2k\to 2k+1}(\mathcal{I}_{\mathcal{B},\,2k})[\bar{\mathbf{x}}] + D_{2k+2\to 2k+1}(\mathcal{I}_{\mathcal{B},2k+2})[\bar{\mathbf{x}}]\big), \\ S_{L,\,k}[\bar{\mathbf{x}}] &= \mathcal{I}_{\mathcal{B},\,2k}[\bar{\mathbf{x}}] + \tfrac{1}{4}\big(D_{2k-1\to 2k}(S_{H,\,k-1})[\bar{\mathbf{x}}] + D_{2k+1\to 2k}(S_{H,\,k})[\bar{\mathbf{x}}]\big). \end{aligned} \tag{5.10}$$

Efficient displacement compensation results in small ghosting artifacts in low-pass subband images $S_{L,\,k}[\bar{\mathbf{x}}]$ and small values of high-pass subband images $S_{H,\,k}[\bar{\mathbf{x}}]$. The low-pass subband images $S_{L,\,k}[\bar{\mathbf{x}}]$ are ideally as close to $\mathcal{I}_{\mathcal{B},2k}[\bar{\mathbf{x}}]$ as possible.

5.4 Wavelet Decomposition Coefficient Quantization

Unlike the traditional block-based hybrid coding schemes, which use quantization as the key to control the image distortion and bit-stream rate, and adopt the more complicated quantization matrix adapted to the features of the human vision system *(HVS)*, wavelet-based schemes apply a relatively simple uniform scalar quantization with small fine quantization step-sizes.

5.4.1 Uniform Quantization with Dead-Zone Thresholds

The decomposition wavelet coefficients in each subband are quantized by using a uniform scalar quantizer with a central dead-zone. Wavelet decomposition coefficients can be well approximated with a Laplacian probability distribution, and the uniform scalar dead-zone quantization is optimal for the feature of the Laplacian probability distribution and easily satisfies the requirement of the rate-distortion optimization [78].

The width of the central dead-zone is twice the quantization step size. The uniform quantization with a central dead-zone is illustrated in Fig. 5.8.

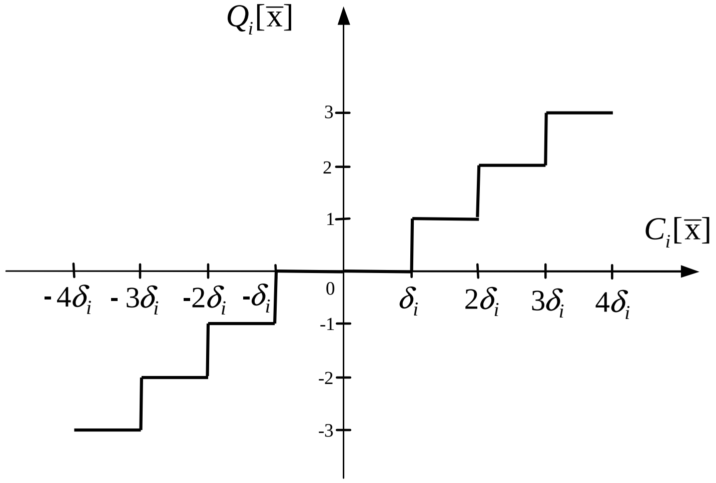


Fig. 5.8 Uniform quantization of wavelet decomposition coefficients with a central dead-zone of $2\delta_i$.

Suppose δ_i ($i \in \{1, 2, ..., I\}$, $I \in \mathbb{Z}$) represents the quantization step-size in a subband SB_i. $C_i[\bar{\mathbf{x}}]$ denotes the wavelet decomposition coefficient in SB_i, and $Q_i[\bar{\mathbf{x}}]$ denotes the quantizer index of $C_i[\bar{\mathbf{x}}]$. The quantizer scales all the decomposition coefficients in subband SB_i to quantization index values as

$$Q_i[\bar{\mathbf{x}}] = \mathrm{sign}(C_i[\bar{\mathbf{x}}]) \lfloor \frac{|C_i[\bar{\mathbf{x}}]|}{\delta_i} \rfloor. \tag{5.11}$$

The quantization step-size δ_i for each subband SB_i is determined based on the contribution of the quantization noise in the subband to the overall mean-squared difference *(MSD)*. The more the quantization noise in the subband contributes to the overall *MSD*, the smaller the quantization step-size is for quantizing the decomposition coefficients in that subband. According to this consideration, the L_2-norm of the subband SB_i is taken to determine the value of the quantization step-size δ_i.

5.4.2 Partial Decoding and Reconstruction of Quantized Coefficients

When a wavelet decomposition coefficient is quantized, the output of the uniform quantizer is the index of the quantized decomposition coefficient. This quantizer index $Q_i[\bar{\mathbf{x}}]$ is represented by a string of D_S-digit binary symbols $\mathcal{S}_B$. Decoding a part of $\mathcal{S}_B$ is equivalent to applying a scaled quantization step-size $\tilde{\delta}_i$ with respect to δ_i. In the case of partial decoding of D_P digits of the binary symbols out of the original D_S-digit binary symbols starting from the most significant bit *(MSB)* towards the least significant bit *(LSB)* of $\mathcal{S}_B$, the equivalent coarser quantization step-size becomes

$$\tilde{\delta}_i = \delta_i \cdot 2^{D_S - D_P}. \qquad (5.12)$$

This corresponds to image quality as well as bitrate scalable coding. It means that the encoder can cease the coding process at intermediate points to meet the bitrate requirement while generating streaming output with scaled image quality. On the other hand, the decoder can just receive or decode fractions of the bit-stream with limited bitrates and reconstruct the coded images with the corresponding image quality. Small original quantization step-sizes δ_i can be used to meet a wide range of scalable image quality and bitrate requirements.

The reconstruction values of quantized coefficients are chosen at the center points of the uncertainty intervals, and alternatively the statistical means of the probability distribution function *(PDF)* of the reconstruction errors can be utilized to determine the reconstruction values for potential improvements [73].

5.5 Context-Based Arithmetic Entropy Coding

Wavelet transforms need to be combined with effective entropy coding approaches to form an efficient compression scheme. Embedded progressive coding is a desired feature of the entropy coding approaches to obtain the target output bit-stream rates and control the distortion of the compressed images. Shapiro [73] developed the embedded zero-tree wavelet algorithm *(EZW)*. It is based on the hierarchical subband decomposition of the wavelet transforms, and generates a fully embedded bit-stream by assembling codes in the order of importance. Said and Pearlman [69] offered an alternative exposition of the *EZW* principles. They proposed a new extension and implementation of progressive embedded encoding called *SPIHT* (set partitioning in hierarchical trees). *EZW* and *SPIHT* are important embedded coding techniques widely applied in wavelet-based compression schemes. They exploit the inter-subband redundancy to enhance the compression efficiency while sacrificing the ability of resolution scalable coding and restricting the flexibility of random access to coded image datasets. Taubman [84] developed an image compression algorithm of embedded block coding with optimized truncation *(EBCOT)*. It introduces a block coding technique without utilizing the inter-subband dependency and provides the potential of random access to image segments. However, it increases the implementation complexity by making use of fractional bitplane coding to provide a large number of bit-stream truncation points for obtaining finely embedded bit-streams.

A new context-based arithmetic entropy coding approach named independent block with layered data partition *(IBLDP)* is proposed in this thesis. *IBLDP* combines the independent decomposition coefficient block coding with a compact quad-tree bitplane data representation, and features flexible spatial random access and bit-stream assembling, reduced computational expense and implementation complexity, and improved coding efficiency. As an embedded entropy coding approach, *IBLDP* independently generates a progressive code stream for each decomposition coefficient block in each subband. Although some coding efficiency is lost because no inter-subband redundancy is exploited as the necessary cost for obtaining the spatial resolution scalability, it increases the flexibility for combining the code streams of independent coefficient blocks in different orders in the output bit-stream according to the requirements of applications. *IBLDP* has a significantly simplified coding structure because of its efficient quad-tree building procedure, simplified bitplane coding passes and no inter-subband dependency employed. It is suitable for image dataset compression, especially for cubic-panorama image dataset compression, providing the required random access to and selective decoding of coded basis images of the image datasets by benefiting from independent block coding, and facilitating efficient interactive novel-view image rendering with the assistance of the simplified codec structure.

5.5.1 Binary Arithmetic Bitplane Encoder

The quantizer indices of the wavelet transform decomposition coefficients in a subband are coded with a binary arithmetic bitplane encoder. First of all, the quantized decomposition coefficients in subband SB_i are organized into rectangular coefficient block $CB_{i,\,j}$ $(\forall j \in \{1, 2, ..., J\},\ J \in \mathbb{Z})$. The width and the height of $CB_{i,\,j}$ are W_i and H_i respectively. Each quantized coefficient block $CB_{i,\,j}$ in SB_i is independently encoded to provide image segment random access, flexible bit-stream assembling and improved error resilience performance.

The binary symbols of the quantizer indices in a coefficient block $CB_{i,\,j}$ are encoded by the arithmetic bitplane encoder one bitplane after another in a fixed order starting from the *MSB* bitplane to the *LSB* bitplane, according to their contributions to the distortion reduction. The procedure of encoding a bitplane refers to a bitplane pass, and encoding a coefficient block $CB_{i,\,j}$ is performed through a series of bitplane passes. This multiple-pass block encoding process accommodates efficient and sufficiently accurate bit-rate control. Truncating the code-stream of each $CB_{i,\,j}$ is allowed at the boundaries of the bitplane passes to control the bitrate and *SNR*. Progressive bitplane coding facilitates embedded bit-stream assembling by putting the coded quantizer indices of decomposition coefficients into hierarchical representations in order of importance.

In order to provide fine progressive bitrate and *SNR* scalable coding, fractional bitplane or sub-bitplane passes are carefully designed in some bitplane coding schemes

[84]. As fine granularity scalability *(FGS)* of bit-rates or *SNR* is not the major concern in image dataset compression, the bitplane encoder in our compression scheme reduces the implementation complexity due to the simplified entropy coding approach without fractional bitplane processing. Meanwhile, the proposed compression scheme is still able to keep the same optimized rate-distortion features at bitplane boundaries even without the computation-intensive sub-bitplane passes.

Suppose $C_{i,j}[n_x, n_y]$ denotes a decomposition coefficient at $[n_x, n_y]$ in coefficient block $CB_{i,j}$. In the process of encoding a bitplane $\mathcal{P}_m$ ($m \in \{1, 2, ..., M\}$, $M \in \mathbb{Z}$), a decomposition coefficient $C_{i,j}[n_x, n_y]$ is marked "insignificant" and its significance state is set $s_{i,j}[n_x, n_y] = 0$ if the current binary symbol of its quantization index $Q_{i,j}[n_x, n_y]$ still remains zero up to $\mathcal{P}_m$, and it is marked "significant" and its significance state is set $s_{i,j}[n_x, n_y] = 1$ as soon as its first non-zero *MSB* is encountered. Binary quantizer index symbols following the first non-zero *MSB* are called refinement bits. Generally, the majority of the quantized coefficients should be insignificant, especially in the early bitplanes, as the wavelet transform analysis concentrates the signal energy to the lowest subband. Most decomposition coefficients are of very low amplitudes, and the distribution of the insignificant coefficients locally spreads in some way due to the spatial correlation between these coefficients. Taking this feature of quantized decomposition coefficient distribution into consideration, more compact decomposition coefficient representations, such as the zero-tree representation [73] and the hierarchical tree set partitioning representation [69], can be combined into entropy coding techniques. In our coding scheme, a quad-tree data partition representation similar to that used in [28, 32] is applied. The establishment of a quad-tree data structure is depicted in Fig. 5.9. The quad-tree nodes at the first or bottom level of the quad-tree structure consist of the binary quantizer index symbols of the decomposition coefficients $Q_{i,j}[n_x, n_y]$ in $CB_{i,j}$ at the current bitplane $\mathcal{P}_m$. Four consecutive quad-tree nodes at a lower quad-tree level correspond to a parent quad-tree node at the next higher quad-tree level. This parent node is significant and is assigned a significance status value equal to 1 if any of its four child nodes equals 1. Otherwise it is insignificant and assigned a value of 0. The quad-tree structure can be built in this way one level after another from the bottom level up to the top level. The number of the descendant nodes covered by a node at the top level increases to $2^{2\cdot(L_T-1)}$ where L_T is the number of quad-tree levels. By using the quad-tree set partition of quantized decomposition coefficients, the information content in the bitplanes becomes quite low. The binary arithmetic coding can then be applied to efficiently represent the quantizer indices in the bitplanes.

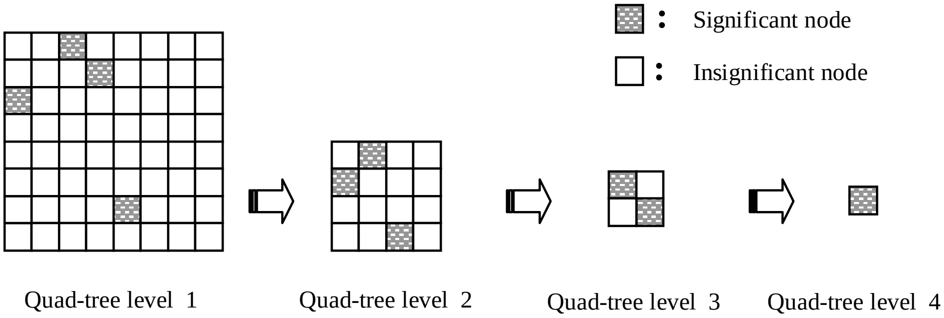


Fig. 5.9 Quad-tree construction for the compact representation of quantized decomposition coefficients in bitplanes.

The binary arithmetic bitplane encoding maps the probability estimations of the binary symbols into the recursively partitioned probability sub-intervals. The probability estimation is performed based on the contexts of the local decomposition coefficient distribution in a model. The resulting codeword refers to a binary fraction locating the sub-interval after the final interval partition. One single binary symbol is encoded each time. One codeword is generated for each coefficient block $CB_{i,\,j}$ to represent all the binary quantization index symbols in the entire $CB_{i,\,j}$.

5.5.2 Context-Based Probability Modeling

As arithmetic entropy coding is applied for representing $Q_{i,\,j}[n_x, n_y]$, the binary quantizer index symbols in each bitplane $\mathcal{P}_m$ are encoded in cooperation with their probability estimates. The probability estimation of the binary symbols is adaptively updated according to their histories and present states. This refers to a context-based probability estimation process. It makes the bitplane coding adapted to the local feature of the quantized decomposition coefficient distribution.

In the process of binary arithmetic bitplane coding, the probabilities of the current binary symbol of the quantizer index are to be estimated in each of the arithmetic sub-interval iterations. These probabilities are related to all the coded binary symbols preceding the current symbol as well as its spatial neighbors. To obtain the proper probability estimates, a context-based probability model including eight immediate neighbors of the current quantized decomposition coefficient Q_O is utilized (Fig. 5.10). The model "context" $\mathcal{C}_M$ is introduced in assistance of the probability estimation. It is related to the statuses of the quantized decomposition coefficients involved in the model. The model context scheme in [84] which carefully reduces the number of contexts in the probability modeling process is utilized in our arithmetic coding approach. In this scheme, the immediate neighboring coefficients of the current decomposition coefficient Q_O are grouped into three categories in the probability estimation model: the horizontal neighbors Q_W and Q_E; the vertical neighbors Q_N and Q_S; and the diagonal neighbors Q_{NW}, Q_{NE}, Q_{SW} and Q_{SE}.

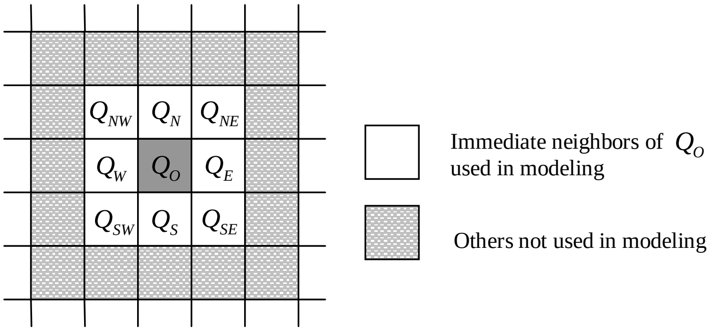

Q_{NW}
Q_N
Q_{NE}
Q_W
Q_O
Q_E
Q_{SW}
Q_S
Q_{SE}
Immediate neighbors of Q_O used in modeling
Others not used in modeling

Fig. 5.10 Context-based probability modeling with eight immediate neighbors of the current quantized decomposition coefficient Q_o.

For coding the binary symbols of insignificant decomposition coefficients in a bitplane $\mathcal{P}_m$, the accumulated values of the significance states of the three categories of coefficient neighbors are computed as follows, with their range of possible values:

$$\begin{aligned} \mu_h[n_x, n_y] &= s_{i,j}[n_x - 1, n_y] + s_{i,j}[n_x + 1, n_y], && 0 \leq \mu_h[n_x, n_y] \leq 2, \\ \mu_v[n_x, n_y] &= s_{i,j}[n_x, n_y - 1] + s_{i,j}[n_x, n_y + 1], && 0 \leq \mu_v[n_x, n_y] \leq 2, \\ \mu_d[n_x, n_y] &= s_{i,j}[n_x - 1, n_y - 1] + s_{i,j}[n_x + 1, n_y - 1] + \\ & \quad s_{i,j}[n_x - 1, n_y + 1] + s_{i,j}[n_x + 1, n_y + 1], && 0 \leq \mu_d[n_x, n_y] \leq 4. \end{aligned} \tag{5.13}$$

The neighborhood significance states used in the equations are obtained in the current bitplane pass up to the current binary quantizer index symbol and in the preceding bitplane pass as well. Based on these accumulated values of significance states, the number of contexts for coding the binary symbols of insignificant coefficients is simplified from $2^8 = 256$ to nine as shown in Table 5.2, in order to minimize the implementation complexity and reduce the cost of probability model adaptation.

LL,LH and HL Bands				**HH Bands**		
μ_h	μ_v	μ_d	$\mathcal{C}_M$	μ_d	$\mu_h + \mu_v$	$\mathcal{C}_M$
0	0	0	0	0	0	0
0	0	1	1	0	1	1
0	0	> 1	2	0	> 1	2
0	1	x	3	1	0	3
0	2	x	4	1	1	4
1	0	0	5	1	> 1	5
1	0	> 0	6	2	0	6
1	> 0	x	7	2	> 0	7
2	x	x	8	> 2	x	8

Table 5.2 Model contexts for adaptive binary arithmetic coding of insignificant coefficients based on the accumulated significance states of three neighborhood categories.

The coding of refinement bits of significant coefficients just use three contexts based on a complementary state variable ρ_s and the horizontal and vertical significance state variables μ_h, μ_v as well. ρ_s is initialized as 0 and becomes 1 after the first refinement bit is encountered. These three contexts are context 0 for $\rho_s = \mu_h = \mu_v = 0$; context 1 for $\rho_s = 0$ and $\mu_h + \mu_v \neq 0$; and context 2 for $\rho_s = 1$.

When a decomposition coefficient changes from an insignificant coefficient to a significant one, its sign needs to be coded. With the context-based arithmetic coding approach, coding the sign of a decomposition coefficient is related to the sign information of its immediate horizontal and vertical neighbors. τ_h and τ_v are used to represent the horizontal and vertical neighboring sign statuses respectively, and η_h and η_v are derived from τ_h and τ_v respectively to decide on the contexts used for coding the signs of quantized decomposition coefficients. They are given by

$$\begin{aligned}\tau_h[n_x, n_y] &= \text{sign}(Q_W) + \text{sign}(Q_E) = \text{sign}(Q_{i,j}[n_x - 1, n_y]) + \text{sign}(Q_{i,j}[n_x + 1, n_y]),\\ \tau_v[n_x, n_y] &= \text{sign}(Q_W) + \text{sign}(Q_E) = \text{sign}(Q_{i,j}[n_x - 1, n_y]) + \text{sign}(Q_{i,j}[n_x + 1, n_y]),\end{aligned} \tag{5.14}$$

$$\eta_h[n_x, n_y] = \begin{cases} -1, & \tau_h \in \{-1, -2\}, \\ 0, & \tau_h = 0, \\ 1, & \tau_h \in \{1, 2\}, \end{cases} \quad \eta_v[n_x, n_y] = \begin{cases} -1, & \tau_v \in \{-1, -2\}, \\ 0, & \tau_v = 0, \\ 1, & \tau_v \in \{1, 2\}. \end{cases} \tag{5.15}$$

Five contexts refined from the original $3^4 = 81$ possible configurations of the signs of horizontal and vertical neighbors are listed in Table 5.3. They are described based on the horizontal and vertical neighborhood sign information represented by η_h and η_v. These contexts are used in the probability estimation for coding the sign τ of the new significant coefficients. However, the actual binary symbol encoded by using these contexts is $\tau \cdot \tau_p$ instead of τ itself. τ_p is the sign prediction listed in Table 5.3.

η_h	η_v	τ_p	$\mathcal{C}_M$
1	1	1	4
1	0	1	3
1	-1	1	2
0	1	-1	1
0	0	1	0
0	-1	1	1
-1	1	-1	2
-1	0	-1	3
-1	-1	-1	4

Table 5.3 Model contents for adaptive binary arithmetic coding of coefficient signs based on the sign information of the neighborhood decomposition coefficients.

Context-based arithmetic coding has significant advantages over the traditionally applied Huffman coding. With changes in the symbol probabilities, new codewords need to be introduced in Huffman coding. In contrast, context-based arithmetic coding can easily adapt to the symbol probability changes by updating the probability estimation in context-based modeling without use of new codewords.

5.5.3 Independent Block with Layered Data Partition (IBLDP) Coding Algorithm

Definitions:

- $\mathbf{A}_{i,j}$: Significance state array consisting of the significance state elements $s_{i,j}[n_x, n_y]$ of the quantized decomposition coefficients $C_{i,j}[n_x, n_y]$ in coefficient block $CB_{i,j}$.

- $q_{i,\,j,\,m}[n_x, n_y]$: Binary symbol of quantizer index $Q_{i,\,j}[n_x, n_y]$ on the mth bitplane $\mathcal{P}_m$ ($m \in \{1, 2, ..., M\}$) in coefficient block $CB_{i,\,j}$.

- $T_{m,\,n}$: Quad-tree structure of the binary quantizer indices at quad-tree level n ($n \in \{1, 2, ..., N\}$) on the mth bitplane $\mathcal{P}_m$.

- $N_{m,\,n}[n_x, n_y]$: Quad-tree node in quad-tree structure $T_{m,\,n}$ at quad-tree level n of the mth bitplane $\mathcal{P}_m$.

Main Process:

- 1. Partition quantized decomposition coefficients $C_{i,j}$ in a subband SB_i ($i \in \{1, 2, ..., I\}$) starting with $i = 1$ into coefficient blocks $CB_{i,j}$ ($\forall j \in \{1, 2, ..., J\}$) of size $W_i \times H_i$.

- 2. Initialize the accompanying significance state array $\mathbf{A}_{i,\,j}$ for a coefficient block $CB_{i,j}$ ($j \in \{1, 2, ..., J\}$) starting with $j = 1$ in subband SB_i.

- 3. Initialize all the model contexts $\mathcal{C}_{M,\,k}$ ($\forall k \in \{1, 2, ..., K\}$) before the start of coding coefficient block $CB_{i,j}$.

- 4. At the mth bitplane $\mathcal{P}_m$ ($m \in \{1, 2, ..., M\}$) starting from the MSB bitplane $\mathcal{P}_M$ in descent order,

  ○ 4.1. build the quad-tree structure $T_{m,n}$ at all quad-tree levels L_n ($\forall n \in \{1, 2, ..., N\}$) starting from the bottom level $n = 1$.

  ∗ For $n = 1$, $N_{m,\,1}[n_x, n_y] = q_{i,\,j,\,m}[n_x, n_y]$.

  ∗ For $n > 1$, $N_{m,\,n}[n_x, n_y]$ is determined by the logical bitwise inclusive OR of its four child nodes: $N_{m,\,n}[n_x, n_y] = N_{m,\,n-1}[2n_x - 1, 2n_y - 1] \mid N_{m,\,n-1}[2n_x, 2n_y - 1] \mid N_{m,\,n-1}[2n_x - 1, 2n_y] \mid N_{m,\,n-1}[2n_x, 2n_y]$.

  ○ 4.2. At the highest quad-tree level L_N, scan each quad-tree node $N_{m,\,N}[n_x, n_y]$. For each $N_{m,\,N}[n_x, n_y]$,

∗ invoke the binary arithmetic coding process (a): *Coding Quad-tree Nodes*;

∗ if $N_{m,N}[n_x, n_y] = 1$,

invoke the sub-process: *Processing Each Quad-tree Node.*

∘ 4.3. If $m > 1$, $m = m - 1$, and return to step 4.

- 5. If $j < J$, $j = j + 1$, and return to step 2.

- 6. If $i < I$, $i = i + 1$, and return to step 1.

Sub-process:

- *Processing Each Quad-tree Node:*

Visit each child quad-tree node of $N_{m,n}[n_x, n_y]$ at $[n'_x, n'_y] \in \{[2n_x - 1, 2n_y - 1], [2n_x, 2n_y - 1], [2n_x - 1, 2n_y], [2n_x, 2n_y]\}$. Let $n' = n - 1$.

∘ For $n' = 1$,

∗ with $s_{i,j}[n'_x, n'_y] = 1$, encode the refinement bit $q_{i,j,m}[n'_x, n'_y]$ of the significant coefficient $C_{i,j}[n'_x, n'_y]$ by invoking the context-based binary arithmetic coding process (c): *Coding Refinement Bits*;

∗ with $s_{i,j}[n'_x, n'_y] = 0$,

- if $q_{i,j,m}[n'_x, n'_y] = 1$, encode the sign $\tau_{i,j}[n'_x, n'_y]$ of $Q_{i,j}[n'_x, n'_y]$ by invoking the context-based binary arithmetic coding process (b): *Coding Coefficient Signs*, and update significant state $s_{i,j}[n'_x, n'_y]$;

- encode $q_{i,j,m}[n'_x, n'_y]$ by invoking the context-based binary arithmetic coding process (d): *Coding Insignificant Bits.*

∘ For $n' > 1$,

∗ encode quat-tree node $N_{m,n'}[n'_x, n'_y]$ by invoking the binary arithmetic coding process (a): *Coding Quad-tree Nodes*;

∗ if $N_{m,n'}[n'_x, n'_y] = 1$, reset $n = n'$ and invoke the sub-process: *Processing Each Quad-tree Node.*

Binary Arithmetic Coding:

- (a). *Coding Quad tree Nodes:*

  Code the binary status symbols of quad-tree nodes $N_{m,n}[n_x, n_y]$ as uniformly distributed binary symbols with binary arithmetic coding. No adaptive model context is used for coding the quad-tree nodes.

- (b). *Coding Coefficient Signs:*

  With context-based binary arithmetic coding, obtain probability estimates based on the horizontal and vertical neighborhood coefficient sign configurations $\eta_h[n_x, n_y]$ and $\eta_v[n_x, n_y]$ combined with the sign prediction variable $\tau_p[n_x, n_y]$, utilizing the five model contexts $\mathcal{C}_{M,k}$ in Table 5.3.

- (c). *Coding Refinement Bits:*

  Code the refinement bits $q_{i,j,m}[n_x, n_y]$ of the quantizer index $Q_{i,j}[n_x, n_y]$ of significant decomposition coefficients $C_{i,j}[n_x, n_y]$ with $s_{i,j}[n_x, n_y] = 1$ by context-based binary arithmetic coding. Probability estimates are determined by the coefficient state variable $\rho[n_x, n_y]$ in combination with the significance state configurations of neighboring horizontal and vertical decomposition coefficients $\mu_h[n_x, n_y]$ and $\mu_v[n_x, n_y]$. Three model contexts $\mathcal{C}_{M,k}$ are used.

- (d). *Coding Insignificant Bits:*

  With context-based binary arithmetic coding, obtain the probability estimates based on the significance states of the eight immediate neighboring coefficients in the model and encoded the binary symbols $q_{i,j,m}[n_x, n_y]$ of insignificant coefficients $C_{i,j}[n_x, n_y]$ with $s_{i,j}[n_x, n_y] = 0$. Use the nine model contexts $\mathcal{C}_{M,k}$ in Table 5.2 determined by the combinations of three categories of neighborhood significance states $\mu_h[n_x, n_y]$, $\mu_v[n_x, n_y]$ and $\mu_d[n_x, n_y]$.

5.6 Subregion Random Access and Selective Decoding

Providing appropriate random access and selective decoding in an efficient coding scheme is a unique requirement in compressing image datasets. Novel-view images are synthesized at interactive rates from image segments taken from basis images. This requires frequent, fast retrieval and decoding of partial image data out of image dataset bit-streams. For cubic-panorama image dataset compression, the desired image segments for synthesizing novel-view images refer to rectangular subregion side-images from cubic-panorama basis images. Therefore, cubic-panorama image datasets need to be well organized in the compressed bit-streams, and only the coded partial images necessary for reconstructing the desired rectangular subregion side-images are to be loaded into the memory for efficient decoding and image rendering. Compression techniques supporting direct rendering from compressed image datasets are needed. High coding efficiency as the fundamental objective of image dataset compression should be kept while accommodating a suitable random access and selective decoding mechanism.

In our proposed cubic-panorama image dataset compression scheme, a mechanism of rectangular subregion side-image random access to basis images of cubic-panorama image datasets in the compressed bit-streams is facilitated. A specific hierarchical data structure is designed to accommodate spatial subregion image random access and maintain the coding performance at high levels. The corresponding bit-stream syntax supporting this data structure is formed to provide efficient selective decoding for interactive image rendering. Multi-level index tables are embedded in the bit-stream to assist spatial image random access. A proper tradeoff between the compression efficiency and the random access flexibility is reached.

5.6.1 Problem Description

Some compression techniques are very important components in image dataset coding schemes for achieving high compression efficiency. However, they make the random access and selective decoding more difficult to achieve with the compressed basis images. For example, cross-image prediction is an effective technique to exploit the redundancy across images; nevertheless, it limits the flexibility of random access to the beginning of the reference basis images. Wavelet transforms decorrelate image samples on a full-frame scale; however, they make the reconstructions of image samples connected to the decoding of neighbor samples. Also, arithmetic entropy coding complicates the procedure for locating the required image segments in the arithmetic codewords, although it is quite effective to improve the entropy coding efficiency. With subband block coding, a whole decomposition coefficient block has to be decoded even though only one sample in the block is needed for image rendering. Moreover, some compression techniques, like multiple-reference prediction and predicted images serving as reference images, could be applied to further improve the coding efficiency; however, they make the dependency chain longer and the random access and selective decoding more complicated. In addition, a popular way to provide random access and selective decoding is to utilize VQ-based techniques. Nevertheless, the compression efficiency of VQ-based techniques is quite limited.

Although dramatically high coding efficiencies were achieved in some of the developed coding schemes for image dataset compression [55, 58, 59], there is no random access and selective decoding mechanism supported in these schemes. The decoder must preload all the coded basis images. The whole image dataset needs to be decoded first before the rendering process can begin. This results in an initial delay and requires a large memory to hold the image dataset. On the other hand, some coding schemes for image dataset compression accommodate random access to coded image segments and selective decoding of the image datasets [53, 74, 75, 99], however, they have limited compression efficiencies. Some compression schemes apply an additional entropy coding process to enhance the compression efficiency after introducing a random access and selective decoding mechanism in initial compression [53, 74], but the entire bit-stream must be loaded into memory and decoded once before the coded image datasets are randomly accessible. Also, this incurs an initial delay, slows the decoding process and makes the coding schemes less efficient. Some other schemes add large overheads in the bit-streams to enable the random access and selective decoding mechanism and thus sacrifice the compression efficiency. It presents a challenge to achieve efficient compression while meeting the random access and selective decoding requirements.

5.6.2 Random Access Mechanism

A mechanism of rectangular subregion side-image random access to basis images of cubic-panorama image datasets in the compressed bit-streams is required in our coding scheme. In order to satisfy this requirement, a specific data structure as well as the corresponding bit-stream syntax supporting the data structure is designed based on the hierarchical subband structure of wavelet decomposition coefficients and the independent coefficient block coding technique adopted in the entropy coding approach.

A rectangular subregion side-image at a certain resolution level required by novel-view image rendering corresponds to a collection of coefficient blocks covering the same spatial subregion in the original basis images. It consists of a particular group of coefficient blocks from various subbands at different resolution levels. Independently coded coefficient blocks are used as the basic building blocks for accommodating the required spatial random access mechanism. The sizes of the coefficient blocks are chosen by balancing the coding efficiency and the random access flexibility. Coefficient blocks are organized in a layered structure to easily support the spatial subregion image access. The code streams of coefficient blocks are assembled in an efficient way to assist the spatial random access and spatial resolution scalable coding. Specific multi-level index tables combined with properly distributed start and end codes are utilized and embedded in the bit-stream. Therefore, the collection of the coefficient blocks required for reconstructing the rectangular subregion side-image can be conveniently located in various subbands at different resolution levels in the bit-stream, and a small part of bit-stream is required to be decoded for reconstructing these coefficient blocks. A tradeoff between the compression efficiency and the random access flexibility is reached.

5.6.3 Hierarchical Data Structure

A special data structure for facilitating rectangular subregion side-image random access at a specific spatial resolution level is designed for cubic-panorama image dataset compression. At the top layer of this hierarchical data structure, a whole cubic-panorama image dataset is divided into a number of groups of basis images *(GBI)*. A *GBI* consists of a collection of basis images. The number of basis images in a *GBI* is determined by the pattern of the cross-image wavelet analysis. For the Haar kernel applied with $\mathbf{L_C}$ levels of cross-image DWT, there are $2^{\mathbf{L_C}}$ basis images involved in a *GBI*.

Wavelet decomposition coefficients in a subband of a decomposed image are partitioned into coefficient blocks. A uniform size is chosen for coefficient blocks in all subbands at the same resolution level. Coefficient blocks at different resolution levels are allowed to have different block sizes. Bigger sizes of coefficient blocks result in more efficient decomposition coefficient representations, but constrain the flexibility of random access to partial image data. Coefficient blocks are organized into a layered structure for facilitating the required spatial random access and achieving efficient coding performance. A set of spatially consecutive coefficient blocks in the same subband of a decomposed image constitutes a coefficient block-set. Coefficient blocks are grouped into block-sets in the same manner in the **HL, LH** and **HH** subbands at the same resolution level. Block-sets corresponding to the same spatial region of the original basis images in all subbands at the same resolution level compose a block-set packet. Particularly, a block-set in the **LL** subband at the lowest resolution level is simultaneously regarded as a block-set packet. All the block-set packets corresponding to the same spatial region of the original basis images at all resolution levels constitute a packet cluster. A cluster union refers to a set of packet clusters from all decomposed images in the same *GBI*. This layered structure of decomposition coefficient block organization is illustrated in Fig. 5.11.

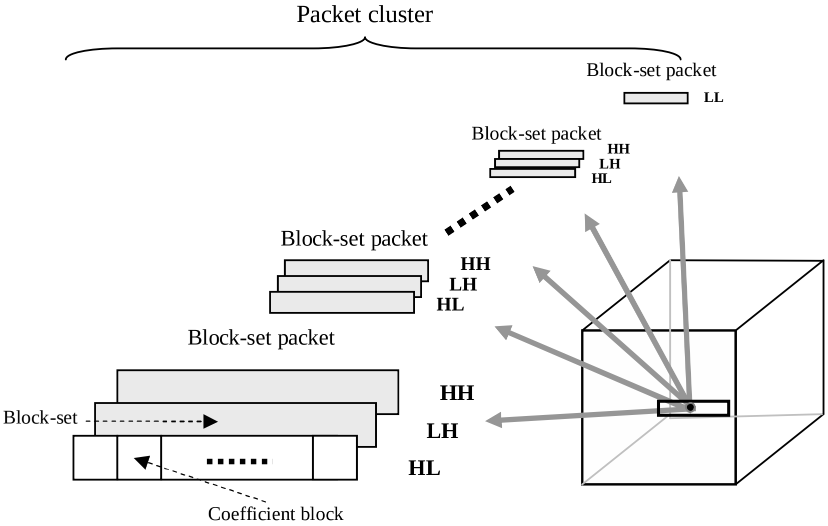


Fig. 5.11 Hierarchical data structure for facilitating spatial random access to coded basis images at a spatial resolution level.

5.6.4 Adapted Bit-Stream Syntax

All code streams from the coefficient blocks are assembled to form a single output bit-stream of the cubic-panorama image dataset. The bit-stream syntax is built to efficiently organize the code streams, support the corresponding hierarchical data structure and assist the implementations of spatial random access and selective decoding at a spatial resolution level.

The bit-stream of a cubic-panorama image dataset starts with an image dataset header. Involved in this header are the global encoding parameters, camera information, *GBI* pattern data, the number of *GBI* involved in the image dataset, the block and block-set sizes, subband quantization step sizes, and the setup information for starting the decoding process. An index table of the code length of each *GBI* in the image dataset is also included in the image dataset header.

Following the cubic-panorama image dataset header are the *GBI* payloads. There is a *GBI* header for each basis image group in the coded cubic-panorama image dataset. The coding parameters applied to the whole image group are put in this header. Also, a second-level index table indicating the start points of all the block-set cluster unions in the image group is involved in the *GBI* header. The entropy-coded displacement fields are put after the *GBI* header. The number of displacement fields involved in a *GBI* depends on the pattern of the cross-image wavelet transform. For the Haar kernel applied with $\mathbf{L_C}$ levels of cross-image *DWT*, there are $2^{\mathbf{L_C}} - 1$ displacement fields contained in a *GBI*. The code streams of cluster unions follow the coded displacement fields. There is a union header at the beginning part of each cluster union. A third-level index table showing the length of each packet cluster in the cluster union is inserted in the union header. The payload of a cluster union refers to the code steam of a series of packet clusters in each decomposed image.

The code stream of each packet cluster includes a set of coded block-set packets at various resolution levels. Each coded block-set packet consists of the code streams of three block-sets in the **HL, LH** and **HH** subbands at the same resolution level. The code stream of a block-set comprises context-based arithmetic codewords from all the coefficient blocks contained in the block-set. The organization of the bit-stream syntax is illustrated in Fig. 5.12.

5.6.5 Selective Decoding Approach

Given a rectangular subregion side-image in a basis image of a cubic-panorama image dataset required for novel-view image rendering at a certain resolution level, the *GBI* number of the basis image group containing this basis image is to be identified first and the corresponding *GBI* in the bit-stream can be positioned by using the first-level index

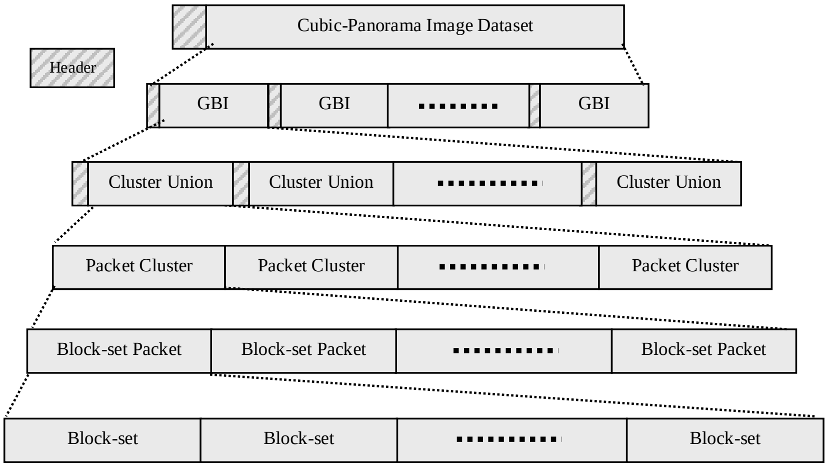


Fig. 5.12 Bit-stream syntax supporting the designed hierarchical data structure facilitating spatial random access at a spatial resolution level.

table in the image dataset header. Then, a group as small as possible of block-sets able to cover the given rectangular subregion side-image is determined based on the sizes of the coefficient blocks and block-sets. Each of these block-sets can be located in the bit-stream by using the second-level index table in the *GIB* header.

The decoding process starts at the beginning of the cluster union payload with the code stream of the the block-set in the lowest resolution subband **LL** at the highest decomposition level, continues, if a higher resolution than the lowest resolution is required, into the next block-set packet to decode the block-set in the **HL, LH** and **HH** subbands in turn until encountering another level end code, and then according to the required resolution level of the decoded subregion image, decides whether or not to get further into the block-set packet at the next higher resolution level until the required resolution level is reached. After finishing decoding the block-set in the first packet cluster, the decoding process moves forward to the beginning of the next packet cluster positioned by using the third-level index table in the cluster union header and follows the same way to decode the block-set in the next packet cluster until the block-set in all the required decomposed images of the *GBI* is decoded. All the coded block-sets in the block-set group corresponding to the desired rectangular subregion side-image are located in the bit-stream by utilizing the multi-level index tables and decoded by repeating the same procedure.

The wavelet decomposition coefficients in these block-sets are decoded with context-based binary arithmetic decoders. The reconstructed decomposition coefficients of a block-set in the **LL** subband of the highest spatial decomposition level refer to a block-set at the lowest resolution level. If higher resolution is required, they are combined with the decomposition coefficients of the block-set in the **HL, LH** and **HH** subbands of the next block-set packet through a wavelet synthesis process. The resultant decomposition coefficients can then be combined with the decomposition coefficients in the block-set packet at the next higher resolution level. This process continues until the desired resolution level is reached, and the decomposition coefficients of the block-set in all the required packet clusters in the cluster union are obtained. Finally the block-set is reconstructed by an inverse cross-image wavelet transform with all the reconstructed decomposition coefficients of the block-set in the cluster union. The required rectangular subregion side-image is obtained from all the reconstructed block-sets involved in the predetermined corresponding block-set group.

5.7 Structure of the Scalable Displacement-Compensated Lifted Wavelet-Based Scheme

The newly-developed scalable displacement-compensated lifted wavelet-based coding scheme is based on the framework of lifted wavelet transforms enhanced by displacement compensation and combined with embedded entropy coding. It is adapted to the features of cubic-panorama image dataset compression and provides a spatial subregion side-image random access mechanism required for compressing cubic-panorama image datasets. A block diagram depicting the structure of the scalable lifted wavelet-based coding scheme with cross-image displacement compensation is shown in Fig. 5.13. The main components involved in this structure are displacement estimation and compensation, lifted cross-image and spatial wavelet transforms, uniform quantization, embedded entropy coding, code stream assembling and bit-stream rate control.

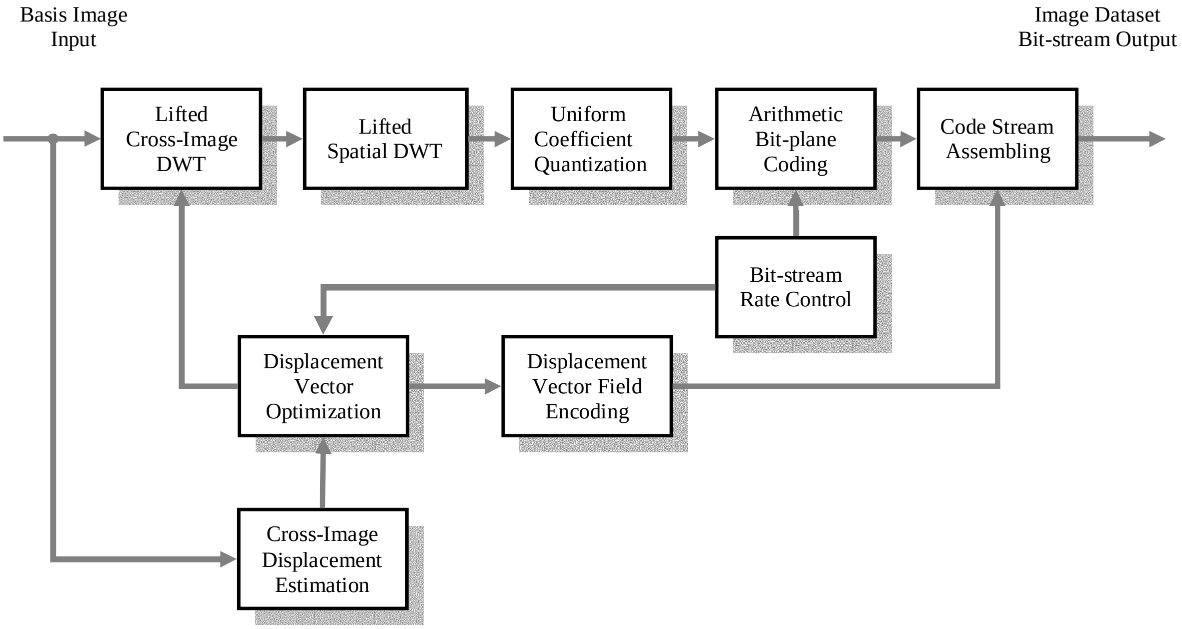

Basis Image Input
Image Dataset Bit-stream Output
Lifted Cross-Image DWT
Lifted Spatial DWT
Uniform Coefficient Quantization
Arithmetic Bit-plane Coding
Code Stream Assembling
Bit-stream Rate Control
Displacement Vector Optimization
Displacement Vector Field Encoding
Cross-Image Displacement Estimation

Fig. 5.13 Scalable lifted wavelet-based coding scheme with cross-image displacement compensation for cubic-panorama image dataset compression.

The cross-image displacement estimation is performed by a block-based full-search approach for every block in the current basis images. In order to accelerate the displacement estimation process and reduce the computational complexity, a hierarchical searching procedure using multi-resolution basis images for obtaining displacement vectors is applied. Half pixel accuracy of displacement vector representations is used in cross-image displacement estimation. The resulting block displacement vectors are refined in a displacement vector optimization process by balancing the increased overhead consumed for more accurate displacement vector representations and the reduced bit rate due to smaller prediction errors [14]. The number of uncovered pixels is calculated based on the displacement field distributions. No cross-image wavelet analysis is performed for the associated basis images if the uncovered pixels outnumber a predefined threshold (e.g. 40%).

Lifted wavelet transforms are applied to convert basis image samples into compact wavelet decomposition coefficient representations and provide the hierarchical data structure for the required scalable coding. Short kernel wavelet transforms are used for cross-image wavelet analysis to reduce the coding delay and avoid large memory occupancy. Mainly, the Haar wavelet kernel is applied for cross-image direction wavelet analysis. Also, the bi-orthogonal LeGall 5/3 wavelet transform is alternatively used for cross-image analysis with enhanced coding efficiency but less flexible random access and selective decoding. Displacement compensation is incorporated into the cross-image wavelet transforms for improved redundancy reduction across basis images by making use of the displacement vector fields obtained in displacement estimation. Following the cross-image wavelet transform, a two-dimensional separable Daubechies 9/7 spatial wavelet transform is performed for all the resulting decomposed images. All these wavelet analysis processes are factorized into lifting operations for fast decomposition and efficient memory usage.

The wavelet decomposition coefficients in each subband resulting from the two-dimensional separable Daubechies 9/7 spatial wavelet transform are scaled with a uniform threshold quantizer. One quantization step size is used in each subband. Then, the quantized decomposition coefficients are partitioned into coefficient blocks. The *IBLDP* (independent block with layered data partition) embedded entropy coding approach is applied for encoding the quantized decomposition coefficient blocks. Each coefficient block is independently encoded with an embedded context-based arithmetic entropy encoder. Code streams from all coefficient blocks are assembled into a single output bit-stream of the compressed cubic-panorama image dataset. The bit-stream rate control component truncates the code streams of coefficient blocks at the boundaries of coded bitplanes and thus changes the values of the equivalent quantization step-sizes.

In this way, it controls the image quality or *SNR* of the coded basis images as well as the bitrates of the output bit-stream.

5.8 Experimental Results

The newly-developed scalable lifted wavelet-based coding scheme with displacement compensation for cubic-panorama image dataset compression is implemented in a software codec. This software-based codec is developed based on a motion-compensated wavelet-based software codec released by the Center for Next Generation Video of Rensselaer Polytechnic Institute [14, 27]. It is applied to encoding the testing cubic-panorama image datasets to examine the newly-developed cubic-panorama image dataset compression scheme and provide support for verifying the proposed algorithms and theoretical contributions. The testing cubic-panorama image datasets used in the experiments are *Lab*, *Corridor*, *Campus* and *Marie-Curie*. To simplify the discussion, our developed compression scheme is denoted as *DCW-IBC* (Displacement-Compensated Wavelet-based scheme with Independent Block Coding) for short.

In order to examine the results of displacement-compensated wavelet transforms with various levels of cross-image analysis applied to encoding cubic-panorama image datasets, Haar wavelet is used as the cross-image wavelet analysis kernel in the proposed compression scheme *DCW-IBC* for compressing the testing cubic-panorama image datasets. Fig. 5.14 and Fig.5.15 show the compared coding performances (in terms of the average PSNR values of the luminance **Y**-component in dB versus the bitrates in kilobits per cube) of cross-image analysis at various levels $L_C = 0$, 1, 2, 3 applied to encoding the cubic-panorama image datasets *Lab* and *Corridor* respectively. Big performance improvements can be observed for cross-image analysis at $L_C = 1$ compared with $L_C = 0$ (without cross-image wavelet analysis), and obvious performance improvements are demonstrated at $L_C = 2$ compared with $L_C = 1$ for these two cubic-panorama image datasets. However, there is no obvious performance difference at level $L_C = 3$ and above compared with $L_C = 2$. Also, it can be observed that the performance improvements of encoding cubic-panorama image dataset *Lab* is bigger than those of encoding cubic-panorama image dataset *Corridor* between different levels showed in the figures while the PSNR performance of encoding *Corridor* is higher than that of encoding *Lab*, as there is less content texture information in image dataset *Corridor* than in image dataset *Lab*. Comparison results of cross-image analysis at levels $L_C = 0, 1, 2$ applied to encoding cubic-panorama image dataset *Campus* and *Marie-Curie* are shown in Fig. 5.16 and Fig. 5.17. There is no obvious performance difference at level $L_C = 2$ and above compared with $L_C = 1$ as there are much bigger position shifts of the camera capturing these outdoor image datasets compared with the location changes of the camera capturing the indoor image datasets and the performance improvements with more levels of cross-image analysis are quite limited compared with encoding indoor cubic-panorama image datasets.

Alternatively, the LeGall 5/3 wavelet kernel is utilized for cross-image wavelet analysis. Fig. 5.18 and Fig. 5.19 show the results of LeGall 5/3 applied for encoding the cubic-panorama image datasets *Lab* and *Corridor* in comparison with the results of the Haar wavelet kernel applied for cross-image wavelet analysis. The coding performance improvements of LeGall 5/3 over Haar range from 0.78 dB to 0.93 dB for *Lab*, and from 0.51 dB to 0.64 dB for *Corridor*. With LeGall 5/3 and Haar applied to encoding the cubic-panorama image datasets *Campus* and *Marie-Curie*, the coding performance improvements are 0.15 dB to 0.23 dB and 0.25 dB to 0.35 dB respectively (Fig. 5.20 and Fig. 5.21). Again, the performance improvements for the outdoor cubic-panorama image datasets *Campus* and *Marie-Curie* are less than those for the indoor cubic-panorama image datasets *Lab* and *Corridor* with different wavelet kernels applied for cross-image wavelet analysis.

Fig. 5.22 to Fig. 5.29 demonstrate the experimental results of spatial resolution scalable coding of the proposed cubic-panorama image dataset compression scheme *DCW-IBC*. The original input basis images (extended horizontal views) from the testing cubic-panorama image datasets are shown in Fig. 5.22 for *Lab*, Fig. 5.23 for Corridor, Fig. 5.24 for *Campus* and Fig. 5.25 for *Marie-Curie*. The reconstructed multi-resolution basis images (extended horizontal views) of *Lab* compressed at 0.2 bpp, Y-PSNR = 38.72 dB respectively in 1/16 resolution (with quarter resolutions in both horizontal and vertical directions), 1/4 resolution (with half resolutions in both horizontal and vertical directions) and full resolution are shown in Fig. 5.26. The reconstructed multi-resolution basis images of *Corridor* compressed at 0.2 bpp, Y-PSNR = 44.40 dB respectively in 1/16 resolution, 1/4 resolution and full resolution are shown in Fig. 5.27. The reconstructed 1/16 resolution, 1/4 resolution and full resolution basis images of *Campus* compressed at 0.4 bpp, Y-PSNR = 37.05 dB are shown in Fig. 5.28, and the reconstructed 1/16, 1/4 and full resolution basis images of *Marie-Curie* compressed at 0.4 bpp, Y-PSNR = 38.58 dB are shown in Fig. 5.29.

For the compression scheme comparison purpose, *MC-EZBC* [28], which is a comparable compression scheme based on the framework of motion-compensated three-dimensional wavelet transforms and an entropy coding approach combining set partitioning with context modeling, is used in our experiments. *MC-EZBC* uses the same across-image and spatial wavelet transform scheme as *DCW-IBC*, but a different embedded entropy coding approach. It treats the cubic-panorama image datasets as generic video sequences for the compression purpose. Fig. 5.30 and Fig. 5.31 show the compared results of both *DCW-IBC* and *MC-EZBC* schemes applied for encoding the testing cubic-panorama image datasets *Lab* and *Corridor*. The coding performances are demonstrated by setting the bit-stream rates of the *MC-EZBC* scheme at the same values as those of the *DCW-IBC* scheme and obtaining the average PSNR values of the luminance **Y**-component for comparison. The results show that the proposed *DCW-IBC* scheme outperforms the *MC-EZBC* scheme over a wide rage of bitrates. Fig. 5.32 and Fig. 5.33 show the compared coding performances of *DCW-IBC* and *MC-EZBC* applied to encoding the testing cubic-panorama image datasets *Campus* and *Marie-Curie*. Again the experimental results show that the newly-developed scheme provides better PSNR performance than that of the *MC-EZBC* scheme.

The coding performances of the newly-developed *DCW-IBC* scheme compared with the *MC-EZBC* scheme for all three color image components **Y, U** and **V** are demonstrated in Table 5.4 and Table 5.5. The PSNR performance of *DCW-IBC* and *MC-EZBC* applied for encoding the cubic-panorama image datasets *Lab* and *Corridor* are shown in Table 5.4 at 0.06, 0.09, 0.12, 0.2 and 0.4 bpp. The performance gains with *DCW-IBC* applied for encoding the testing image dataset *Lab* are 0.46 dB, 0.35 dB and 0.38 dB on average for the **Y, U** and **V** components respectively over the range 0.06 - 0.4 bpp, and for encoding the testing image dataset *Corridor* are 0.57 dB, 0.50 dB and 0.51 dB on average for the **Y, U** and **V** components respectively. The PSNR performance of *DCW-IBC* and *MC-EZBC* applied for encoding the cubic-panorama image datasets *Campus* and *Marie-Curie* are shown in Table 5.5 at 0.06, 0.09, 0.12, 0.2 and 0.4 bpp. The average gains of **Y, U** and **V** are 0.34 dB, 0.24 dB and 0.27 dB with *DCW-IBC* applied for encoding *Campus*, and 0.40 dB, 0.27 dB and 0.29 dB for encoding *Marie-Curie*.

The average PSNR performances of the **Y** component versus the compression ratio for *DCW-IBC* and *MC-EZBC* applied to encoding the cubic-panorama image datasets *Lab*, *Corridor*, *Campus* and *Marie-Curie* are demonstrated in Fig. 5.34 to Fig. 5.37. As an efficient coding scheme for cubic-panorama image dataset compression, the newly-developed scheme achieves improved PSNR performance with the same compression ratio as that of the comparable *MC-EZBC* scheme. This also means a bitrate saving at the same PSNR level.

Besides supporting the required spatial-resolution scalable coding and facilitating the spatial random access mechanism for cubic-panorama image dataset compression, the newly-developed compression scheme provides improved coding efficiency for compressing cubic-panorama image datasets. Experimental results of the software implementation of the scalable displacement-compensated lifted wavelet-based compression scheme show its superior coding performance over the comparable *MC-EZBC* scheme over a wide range of bit-stream rates.

5.9 Summary

A scalable lifted wavelet-based coding scheme with displacement compensation is developed for cubic-panorama image dataset compression. This image dataset compression scheme is based on the framework of lifted wavelet transforms enhanced by displacement compensation and combined with an efficient embedded entropy coding approach. Wavelet transforms generate the efficient representations of image datasets with hierarchically structured decomposition coefficients and provide the potential of spatially scalable coding required by image dataset compression. The lifting operations put wavelet analysis and synthesis into fast and memory-saving in-place computations and make wavelet transforms more feasible in image dataset compression. Displacement compensation significantly improves the coding efficiency of the wavelet transform in cross-image direction.

The proposed *IBLDP* embedded entropy coding approach combines embedded independent coefficient block coding with layered bitplane data partition. Unlike most wavelet-based entropy coding algorithms, which improve the coding efficiency by exploiting the redundancies across subbands and between coefficient blocks and thus sacrifice the resolution scalability and the spatial random access flexibility, no inter-subband and inter-block redundancies are exploited in our proposed entropy coding approach. Coding a subband is independent of other subbands. The independent coefficient block coding in a subband makes the spatial resolution scalability and spatial random image access possible, improves the error resilience performance and easily facilitates parallel processing. It also enables flexible code stream organization and thus the flexible bit-stream syntax. Although some coding efficiency is sacrificed with independent coefficient block coding, it reduces the decoding delay, simplifies the implementation structure and easily facilitates interactive image decoding.

This displacement-compensated lifted wavelet-based coding scheme facilitates an effective rectangular subregion side-image random access mechanism for cubic-panorama-based image rendering. A specific hierarchical data structure is designed to accommodate the spatial image random access and maintain the coding performance at high levels. The corresponding bit-stream syntax supporting this data structure is formed to provide efficient selective decoding for interactive image rendering. Multi-level index tables are embedded in the bit-stream to assist spatial image random access. This newly-developed image dataset compression scheme features a proper compromise between the coding efficiency and the random access flexibility.

Experiments were conducted to examine the newly-developed scalable displacement-compensated lifted wavelet-based compression scheme for cubic-panorama image dataset compression and provide supports for verifying the proposed algorithms and theoretical contributions. Experimental results demonstrate that this image dataset compression scheme achieves superior coding performance over that of the comparable *MC-EZBC* scheme for cubic-panorama image dataset compression.

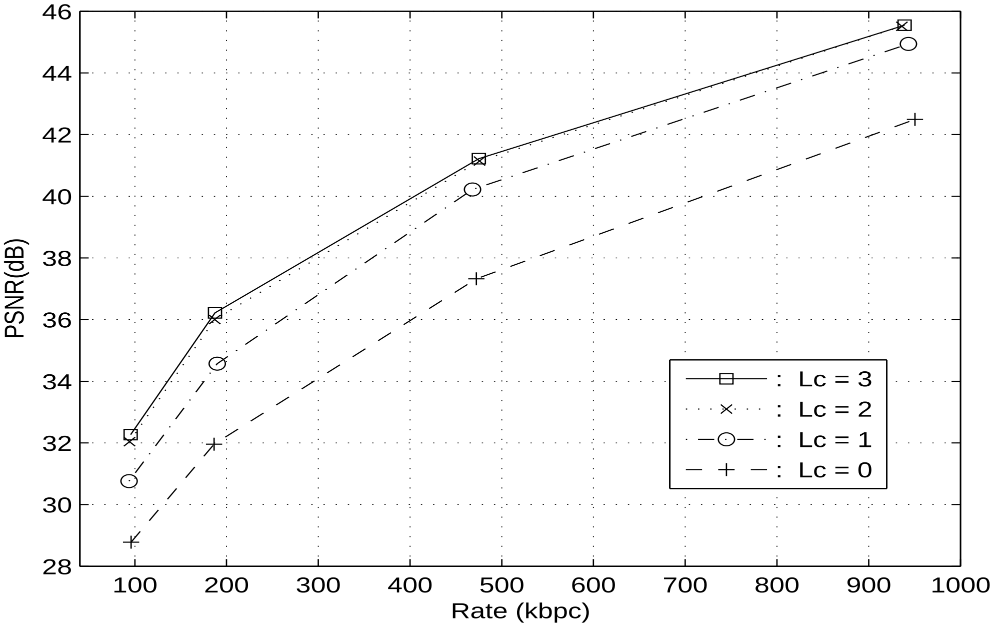


Fig. 5.14 Performance comparison of cross-image analysis at L_C = 0, 1, 2, 3 applied to encoding cubic-panorama image dataset *Lab*.

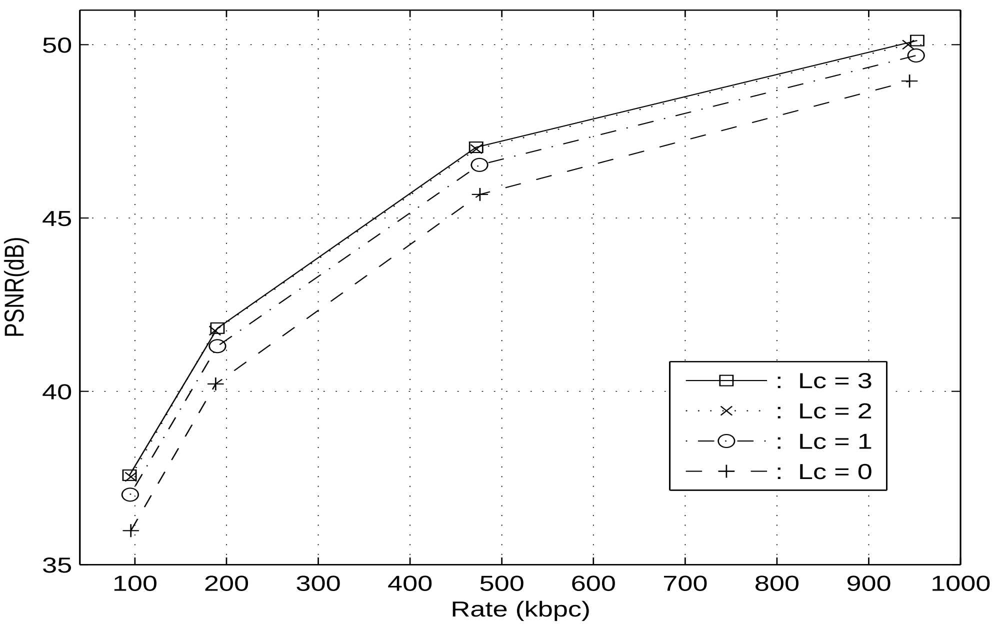


Fig. 5.15 Performance comparison of cross-image analysis at L_C = 0, 1, 2, 3 applied to encoding cubic-panorama image dataset *Corridor*.

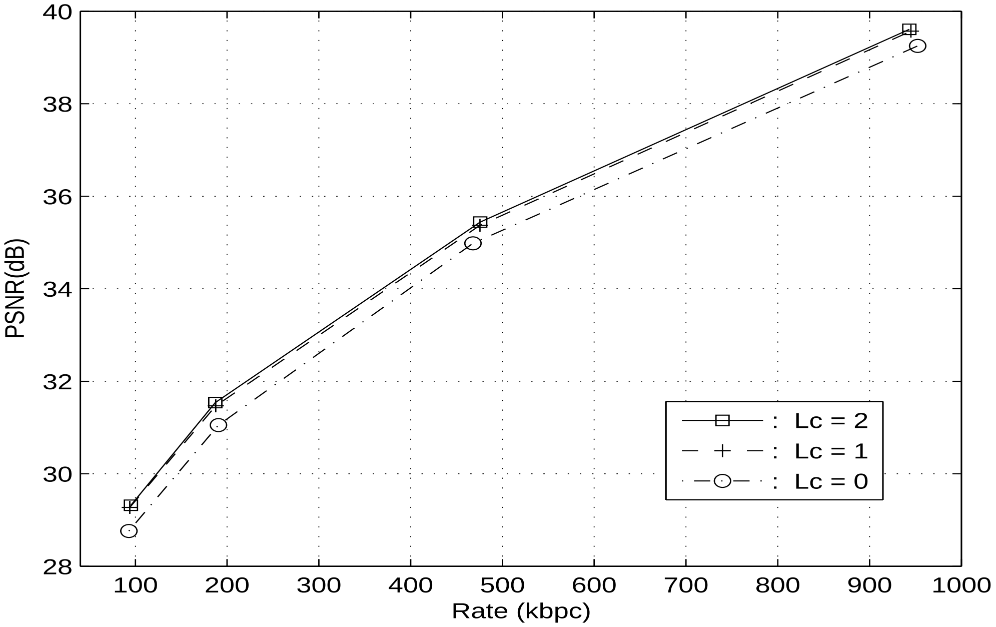


Fig. 5.16 Performance comparison of cross-image analysis at $L_C = 0, 1, 2$ applied to encoding cubic-panorama image dataset *Campus*.

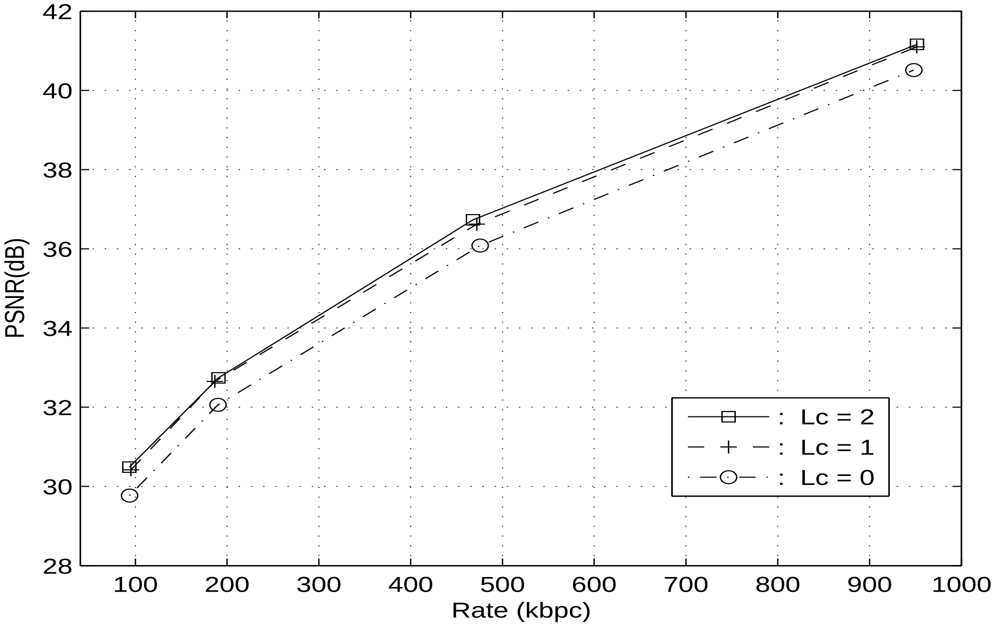


Fig. 5.17 Performance comparison of cross-image analysis at $L_C = 0, 1, 2$ applied to encoding cubic-panorama image dataset *Marie-Curie*.

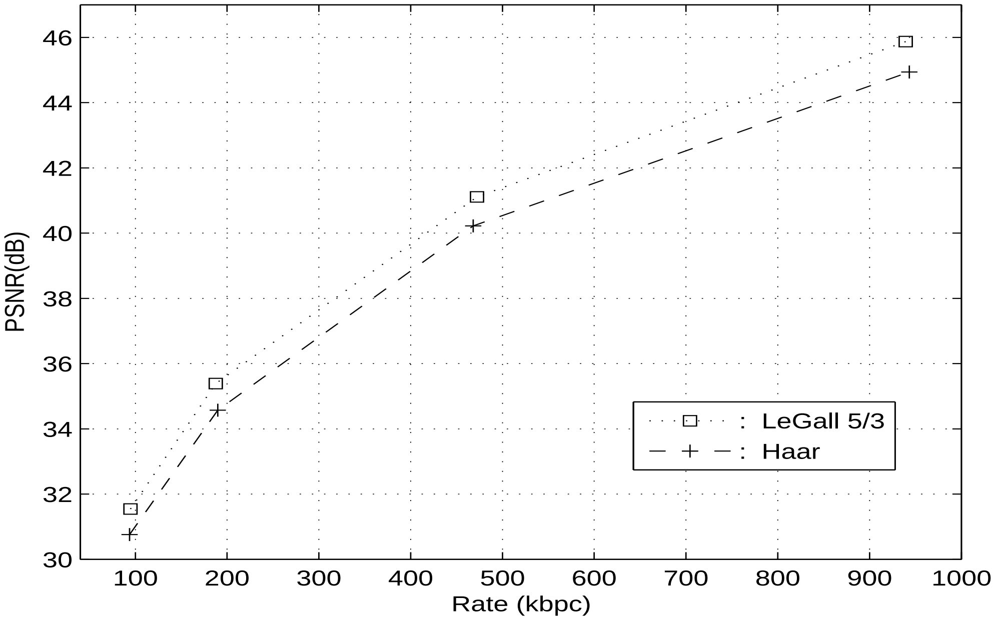


Fig. 5.18 Performance comparison of cross-image wavelet kernels Haar and LeGall 5/3 for encoding cubic-panorama image dataset *Lab*.

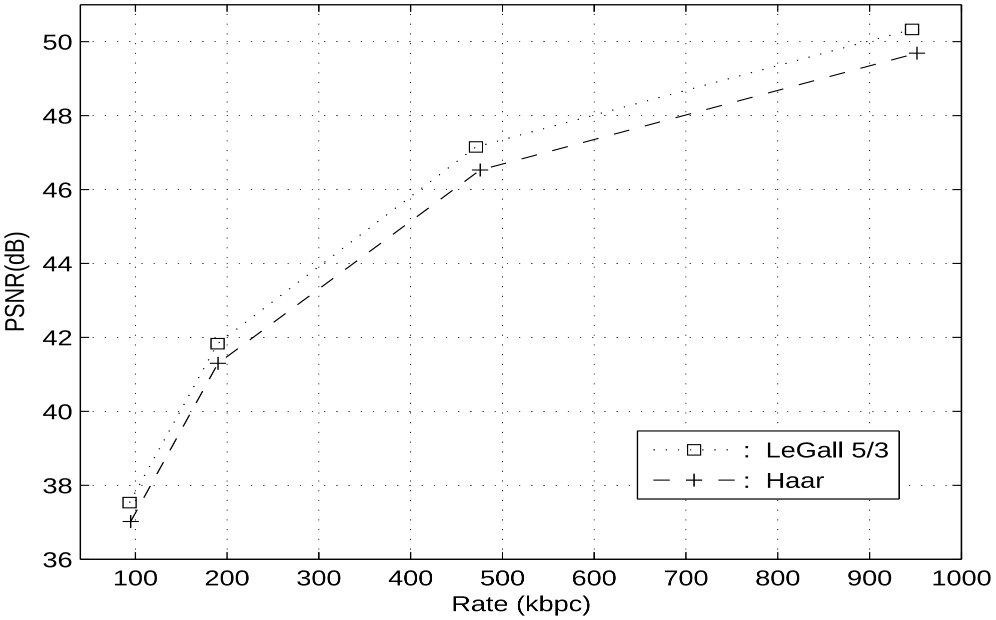


Fig. 5.19 Performance comparison of cross-image wavelet kernels Haar and LeGall 5/3 for encoding cubic-panorama image dataset *Corridor*.

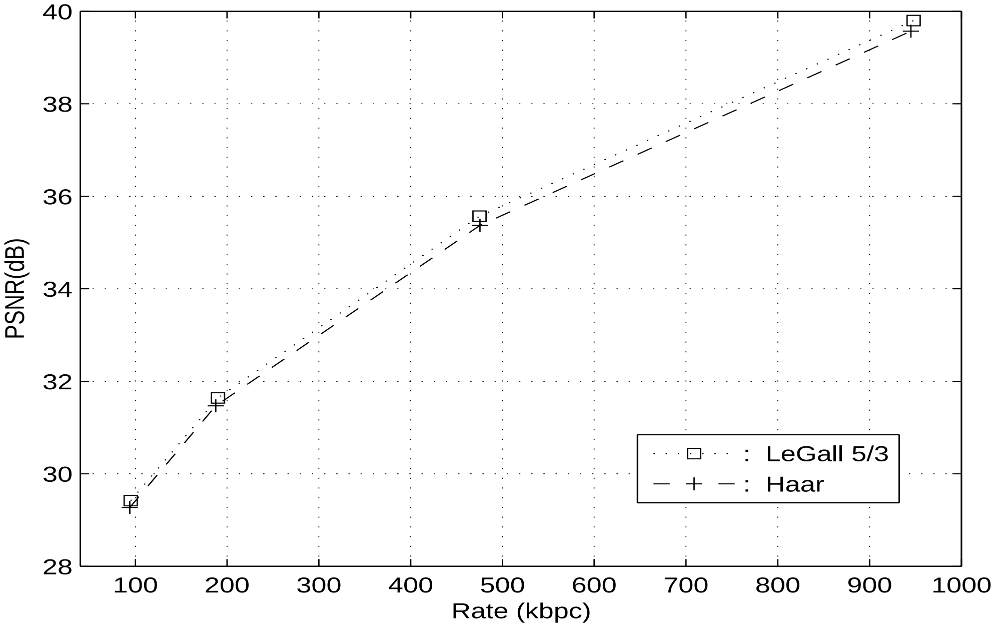


Fig. 5.20 Performance comparison of cross-image wavelet kernels Haar and LeGall 5/3 for encoding cubic-panorama image dataset *Campus*.

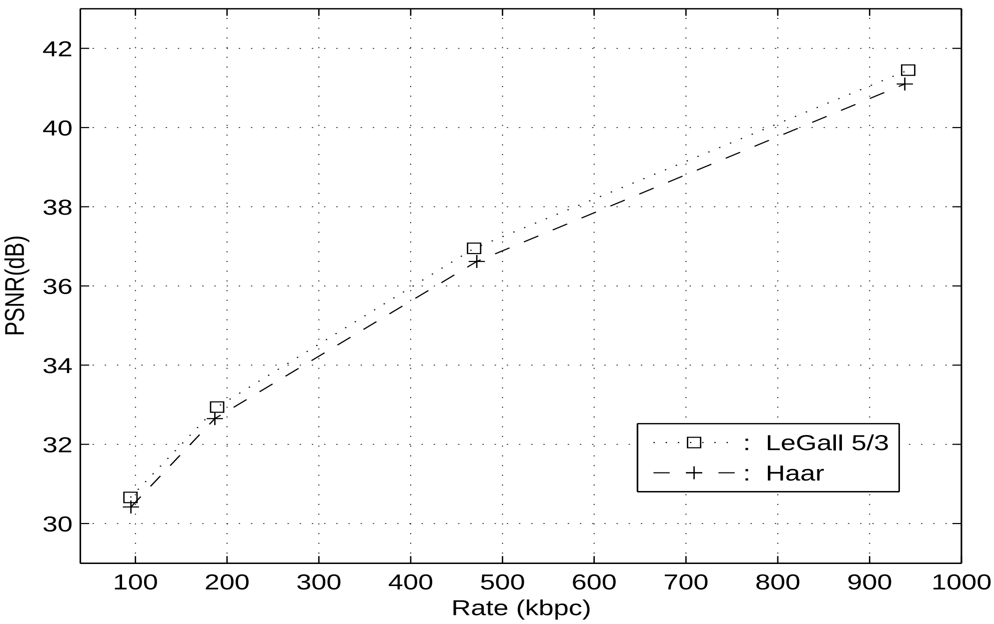


Fig. 5.21 Performance comparison of cross-image wavelet kernels Haar and LeGall 5/3 for to encoding cubic-panorama image dataset *Marie-Curie*.

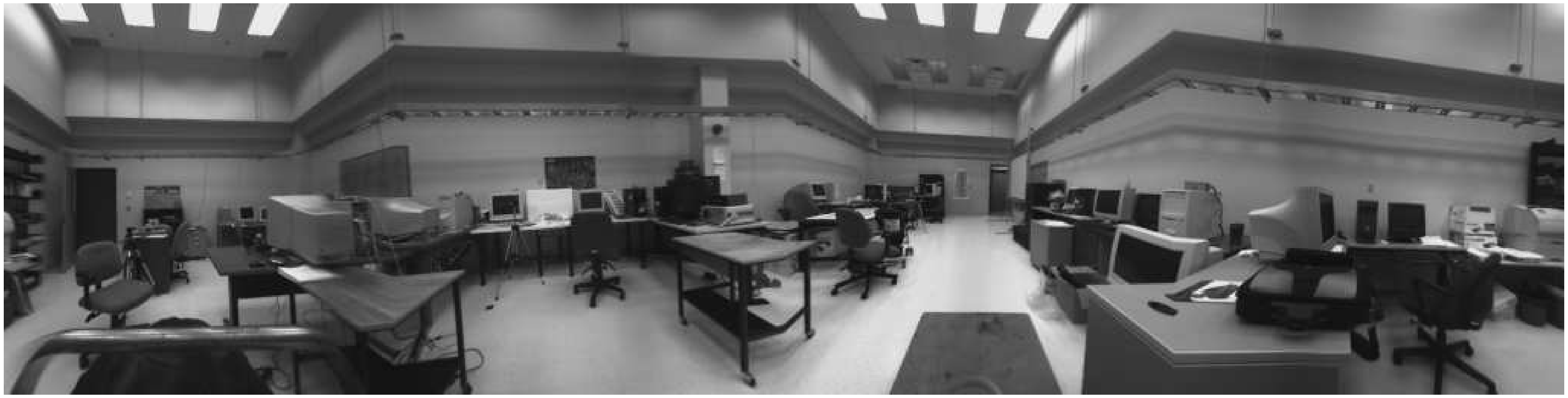


Fig. 5.22 Extended horizontal view of an input basis image from *Lab* used as a testing cubic-panorama image dataset in the experiments.

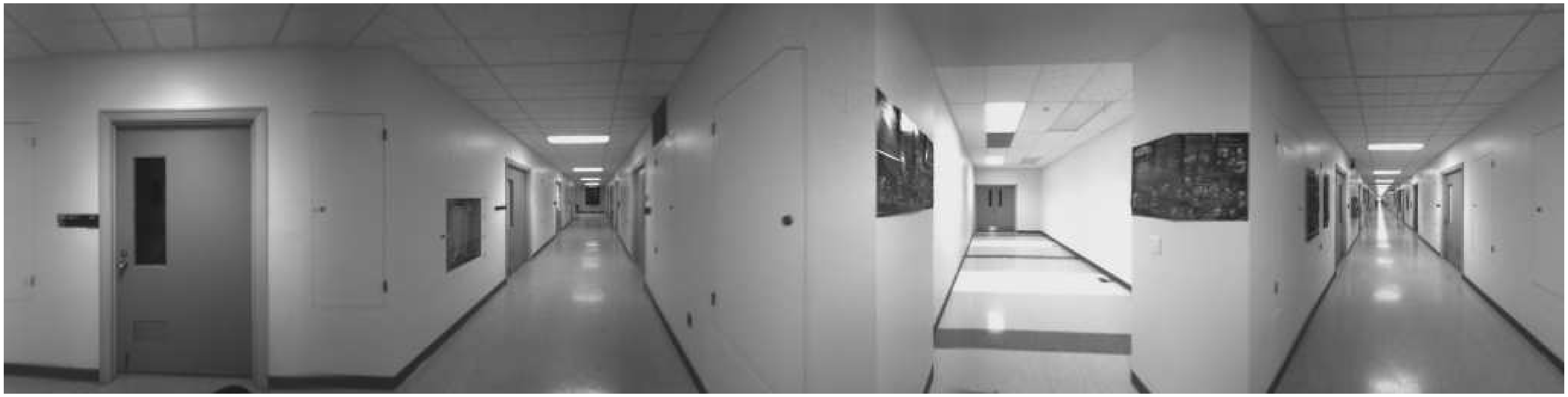


Fig. 5.23 Extended horizontal view of an input basis image from *Corridor* used as a testing cubic-panorama image dataset in the experiments.

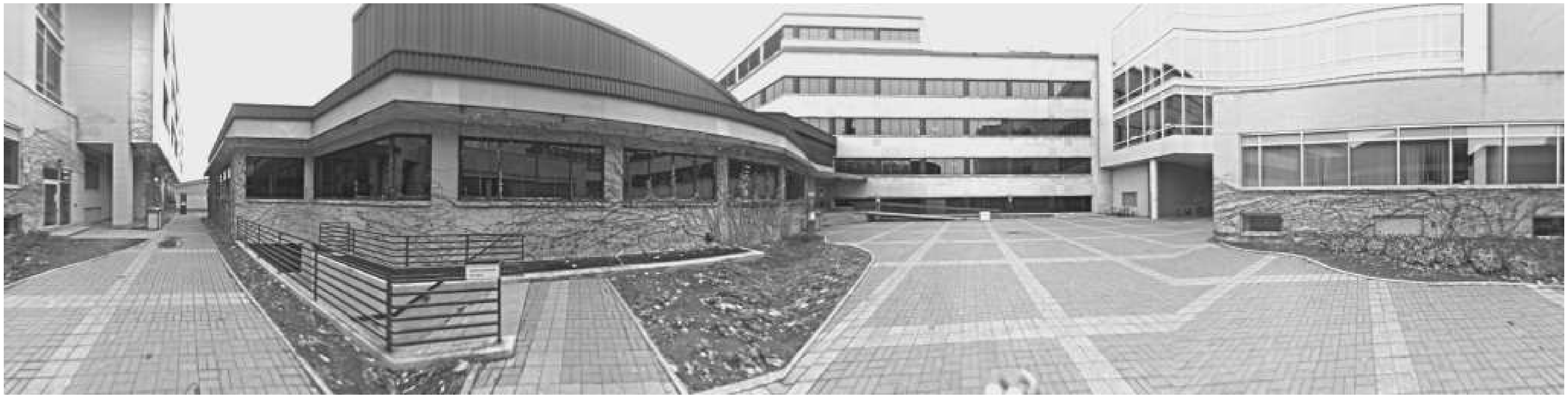


Fig. 5.24 Extended horizontal view of an input basis image from *Campus* used as a testing cubic-panorama image dataset in the experiments.

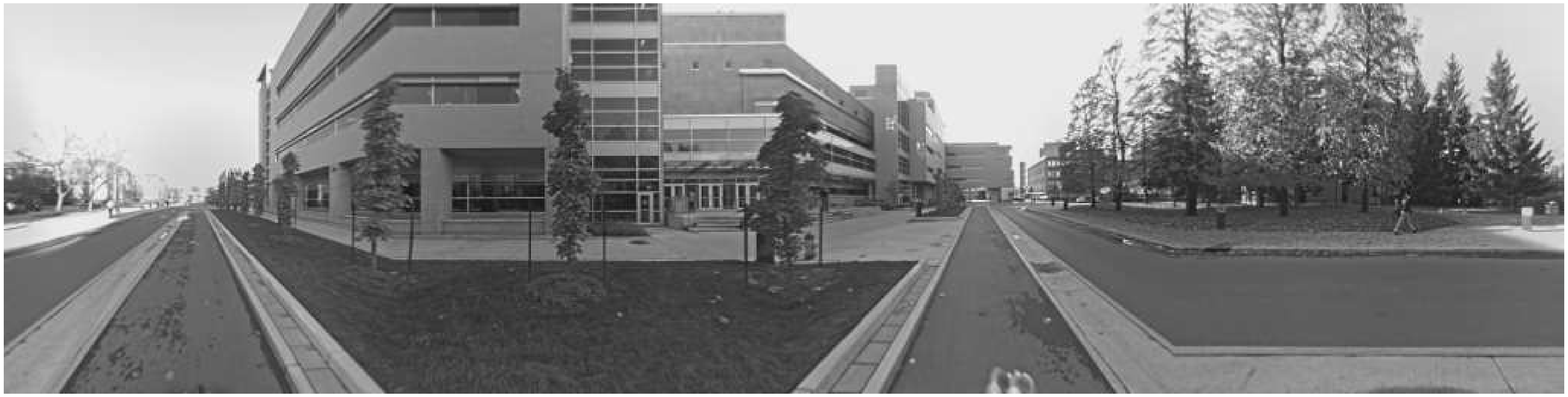


Fig. 5.25 Extended horizontal view of an input basis image from *Marie-Curie* used as a testing cubic-panorama image dataset in the experiments.

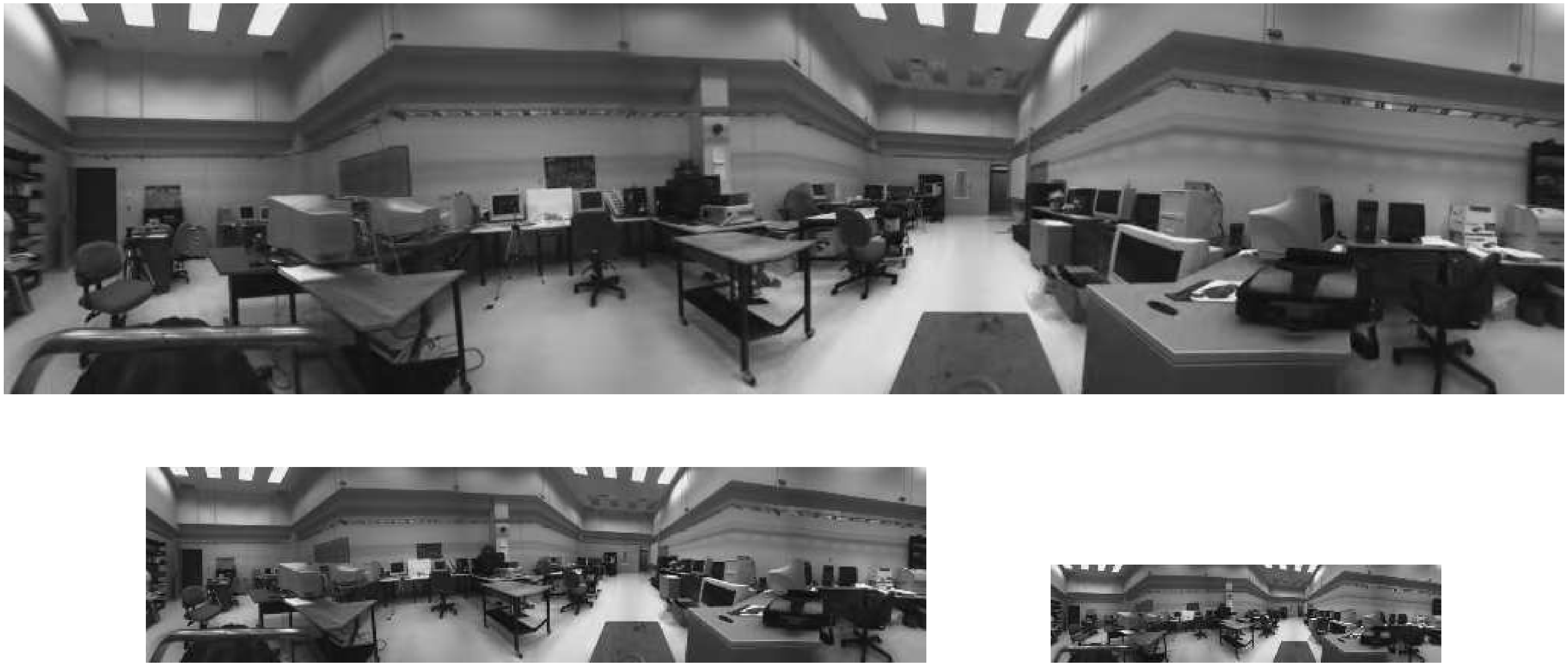


Fig. 5.26 Reconstructed multi-resolution basis images (extended horizontal views) in cubic-panorama image dataset *Lab*: the full resolution image (compressed at 0.2 bpp, Y-PSNR = 38.72 dB) on the top, the 1/4 resolution image on the bottom left and the 1/16 resolution image on the bottom right.

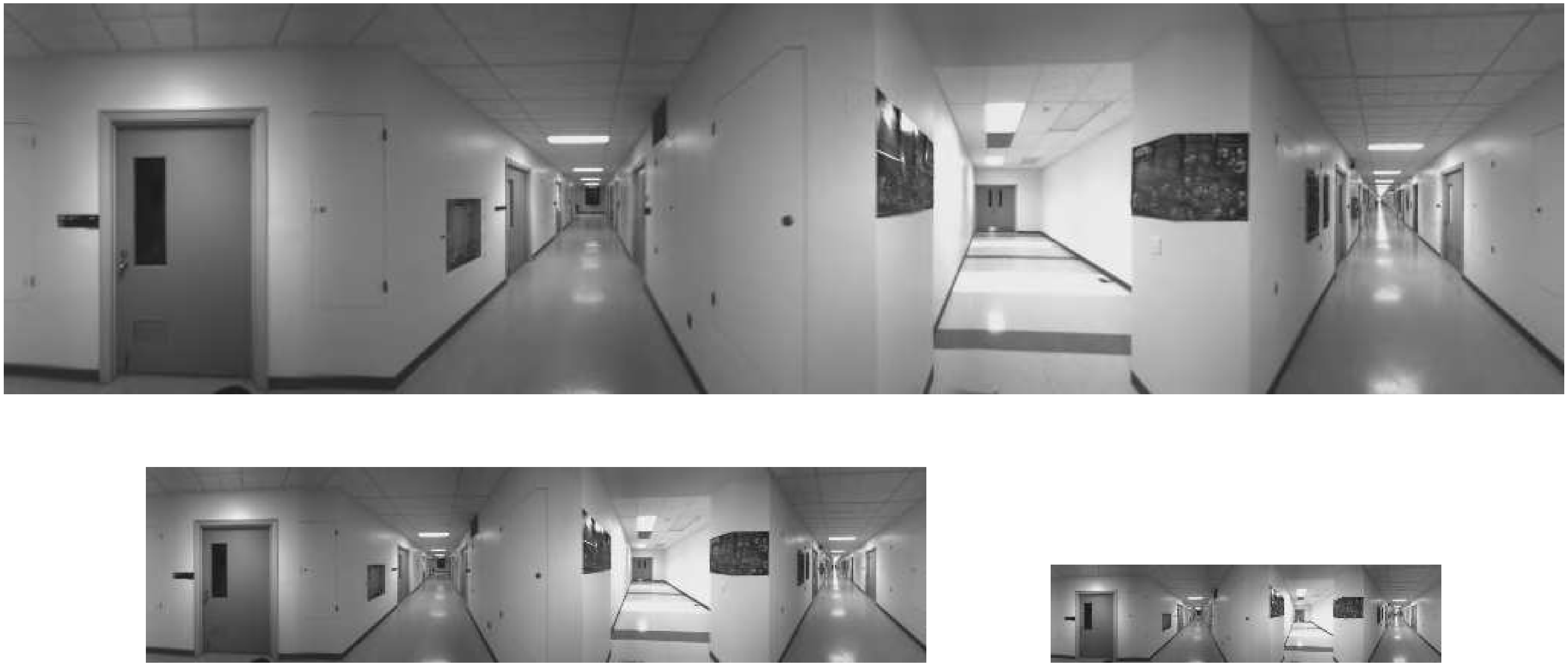


Fig. 5.27 Reconstructed multi-resolution basis images (extended horizontal views) in cubic-panorama image dataset *Corridor*: the full resolution image (compressed at 0.2 bpp, Y-PSNR = 44.40 dB) on the top, the 1/4 resolution image on the bottom left and the 1/16 resolution image on the bottom right.

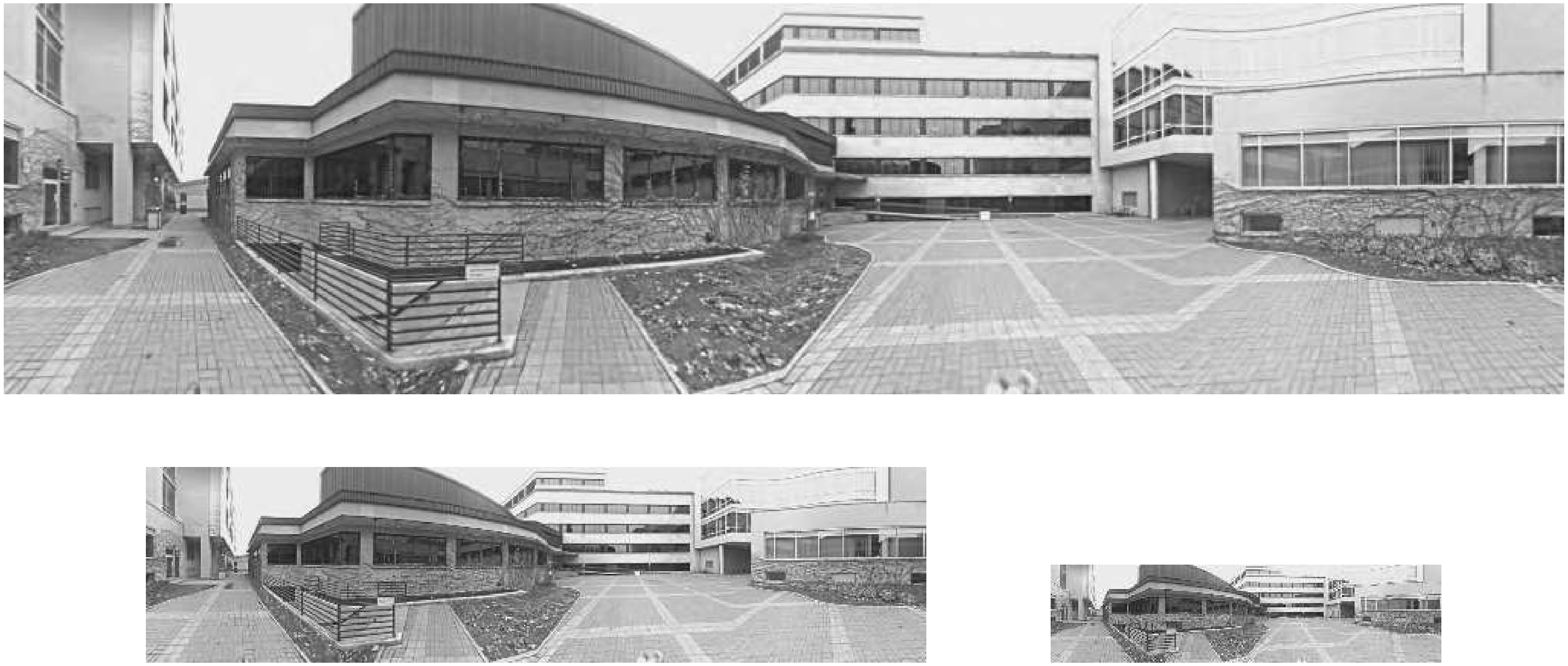


Fig. 5.28 Reconstructed multi-resolution basis images (extended horizontal views) in cubic-panorama image dataset *Campus*: the full resolution image (compressed at 0.4 bpp, Y-PSNR = 37.05 dB) on the top, the 1/4 resolution image on the bottom left and the 1/16 resolution image on the bottom right.

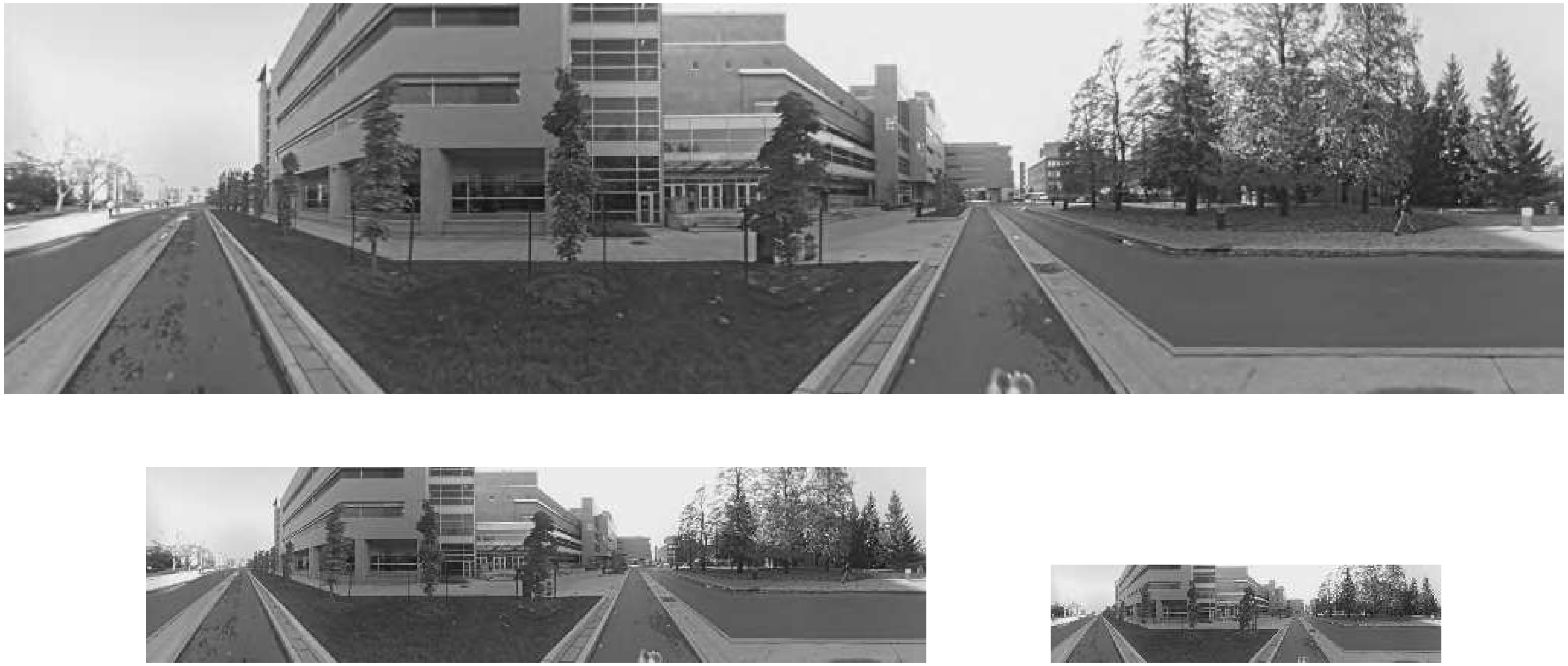


Fig. 5.29 Reconstructed multi-resolution basis images (extended horizontal views) in cubic-panorama image dataset *Marie-Curie*: the full resolution image (compressed at 0.4 bpp, Y-PSNR = 38.58 dB) on the top, the 1/4 resolution image on the bottom left and the 1/16 resolution image on the bottom right.

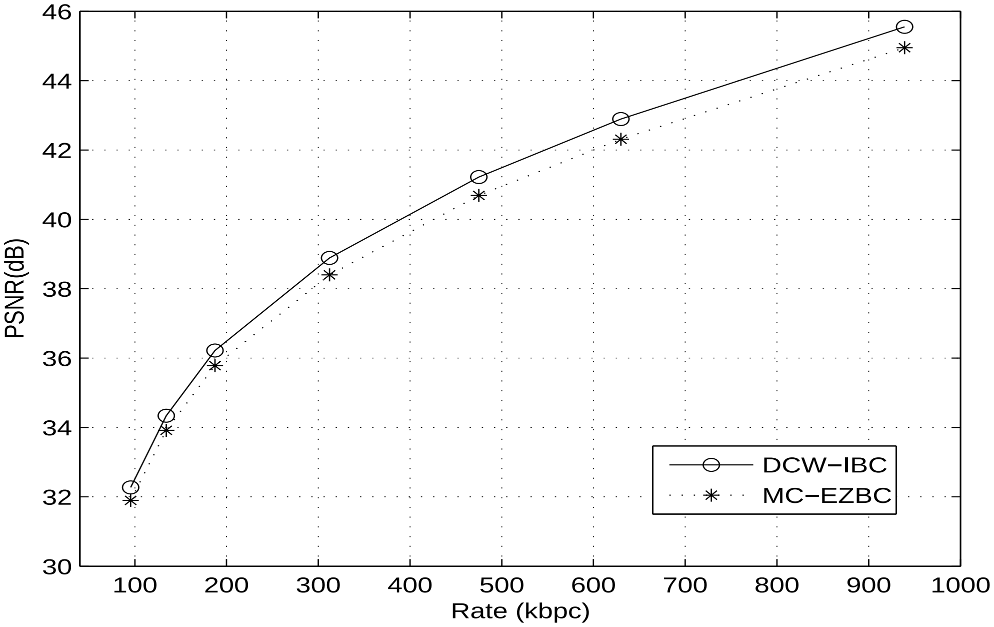


Fig. 5.30 Compression performance comparison of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Lab*.

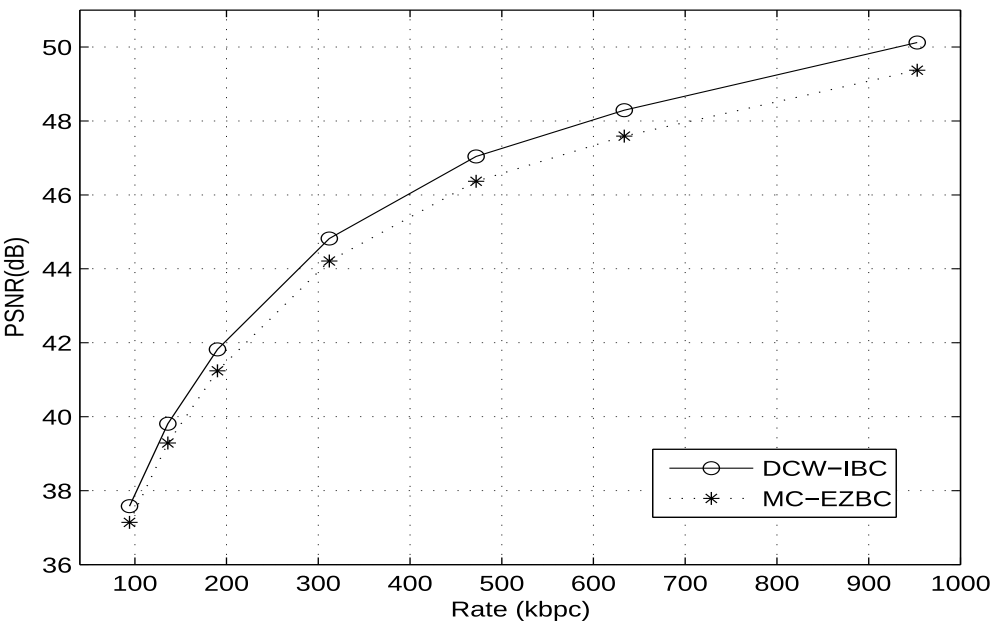


Fig. 5.31 Compression performance comparison of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Corridor*.

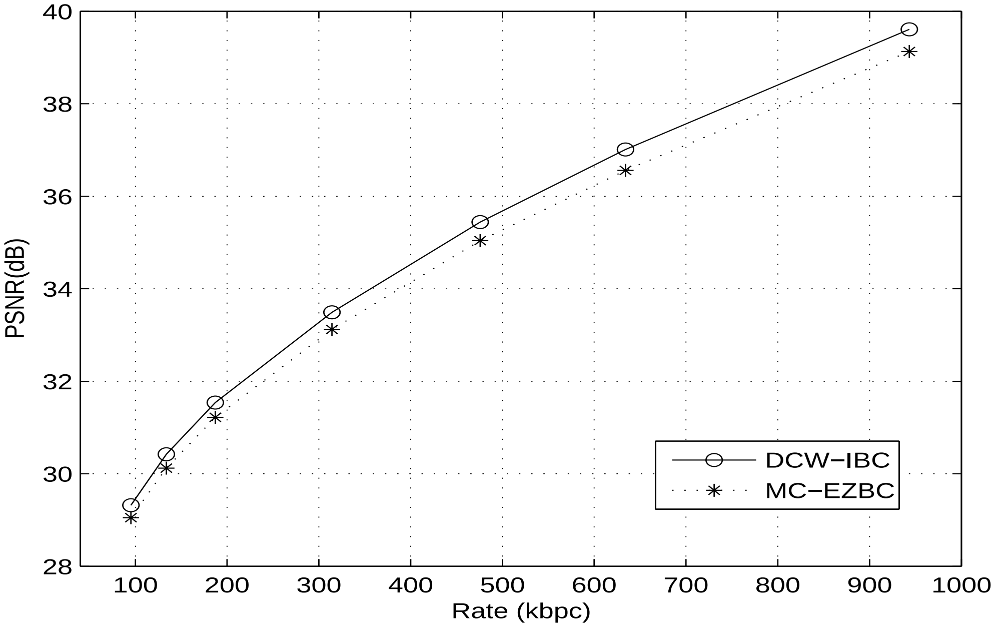


Fig. 5.32 Compression performance comparison of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Campus*.

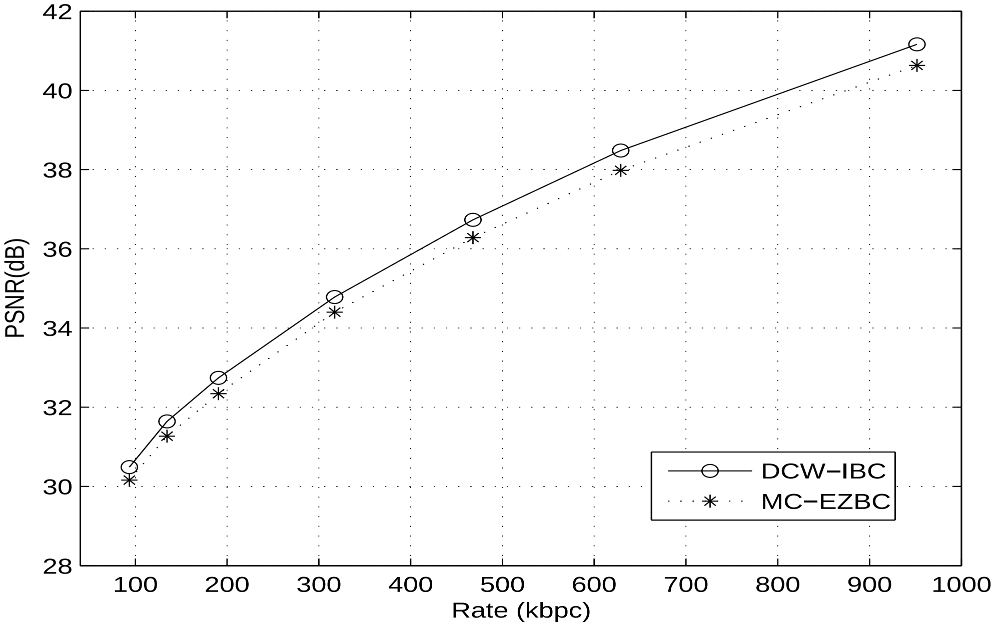


Fig. 5.33 Compression performance comparison of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Marie-Curie*.

Bit Rate	Image Dataset	Comp.[a]	*MC-EZBC* PSNR (dB)	*DCW-IBC* PSNR (dB)	Gain (dB)
0.06 bpp	*Lab*	Y	31.90	32.27	+0.37
		U	40.92	41.21	+0.29
		V	39.34	39.66	+0.32
	Corridor	Y	37.14	37.58	+0.44
		U	42.87	43.23	+0.36
		V	42.06	42.46	+0.40
0.09 bpp	*Lab*	Y	33.92	34.34	+0.42
		U	42.01	42.31	+0.30
		V	40.72	41.07	+0.35
	Corridor	Y	39.29	39.81	+0.52
		U	44.37	44.80	+0.43
		V	43.66	44.14	+0.48
0.12 bpp	*Lab*	Y	35.78	36.22	+0.44
		U	43.04	43.36	+0.32
		V	41.69	42.06	+0.37
	Corridor	Y	41.24	41.82	+0.58
		U	45.77	46.30	+0.53
		V	45.16	45.67	+0.51
0.2 bpp	*Lab*	Y	38.40	38.89	+0.49
		U	44.49	44.90	+0.41
		V	43.50	43.89	+0.39
	Corridor	Y	44.21	44.82	+0.61
		U	47.76	48.33	+0.57
		V	47.37	47.90	+0.53
0.4 bpp	*Lab*	Y	42.31	42.89	+0.58
		U	47.76	48.18	+0.42
		V	47.08	47.53	+0.45
	Corridor	Y	47.59	48.29	+0.70
		U	50.91	51.52	+0.61
		V	50.69	51.34	+0.65

[a] Component

Table 5.4 Coding performance of *DCW-IBC* and *MC-EZBC* applied for encoding the experimental cubic-panorama image datasets *Lab* and *Corridor* at 0.06, 0.09, 0.12, 0.2, 0.4 bpp.

Bit Rate	Image Dataset	Comp.[a]	*MC-EZBC* PSNR (dB)	*DCW-IBC* PSNR (dB)	Gain (dB)
0.06 bpp	*Campus*	Y	29.05	29.32	+0.27
		U	38.44	38.62	+0.18
		V	38.90	39.11	+0.21
	Marie-Curie	Y	30.16	30.49	+0.33
		U	36.83	37.02	+0.19
		V	37.50	37.72	+0.22
0.09 bpp	*Campus*	Y	30.12	30.42	+0.30
		U	39.06	39.27	+0.21
		V	39.51	39.74	+0.23
	Marie-Curie	Y	31.27	31.64	+0.37
		U	37.20	37.47	+0.27
		V	38.07	38.36	+0.29
0.12 bpp	*Campus*	Y	31.22	31.54	+0.32
		U	39.77	40.00	+0.23
		V	40.32	40.57	+0.25
	Marie-Curie	Y	32.34	32.74	+0.40
		U	38.02	38.27	+0.25
		V	38.80	39.09	+0.29
0.2 bpp	*Campus*	Y	33.12	33.49	+0.37
		U	40.92	41.16	+0.24
		V	41.51	41.78	+0.27
	Marie-Curie	Y	34.40	34.78	+0.38
		U	39.25	39.52	+0.27
		V	40.03	40.34	+0.31
0.4 bpp	*Campus*	Y	36.56	37.01	+0.45
		U	43.04	43.39	+0.35
		V	43.49	43.89	+0.40
	Marie-Curie	Y	37.98	38.48	+0.50
		U	41.60	41.97	+0.37
		V	42.51	42.87	+0.36

[a] Component

Table 5.5 Coding performance of *DCW-IBC* and *MC-EZBC* applied for encoding the experimental cubic-panorama image datasets *Campus* and *Marie-Curie* at 0.06, 0.09, 0.12, 0.2, 0.4 bpp.

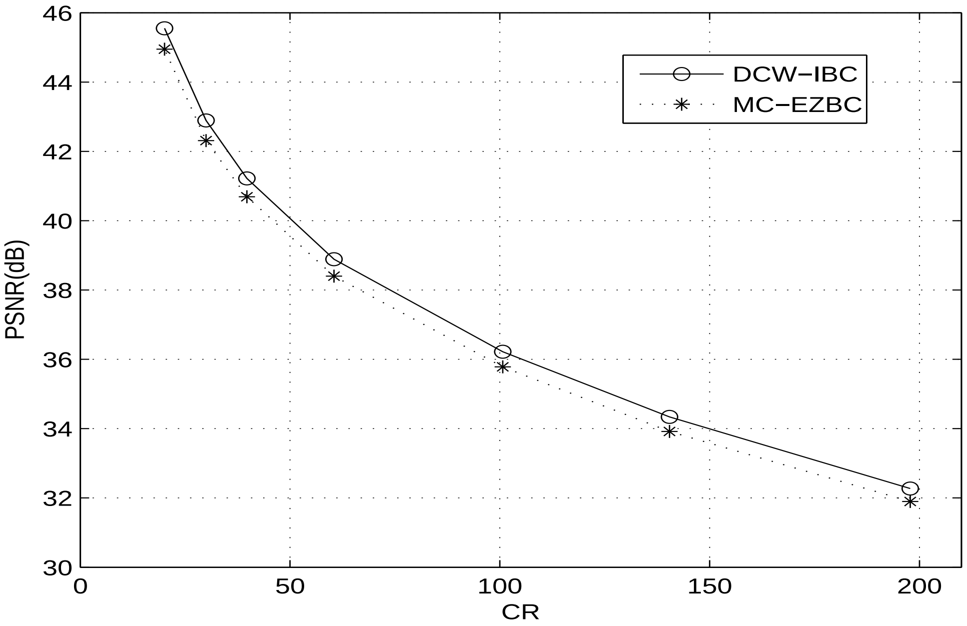


Fig. 5.34 PSNR versus compression ratio of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Lab*.

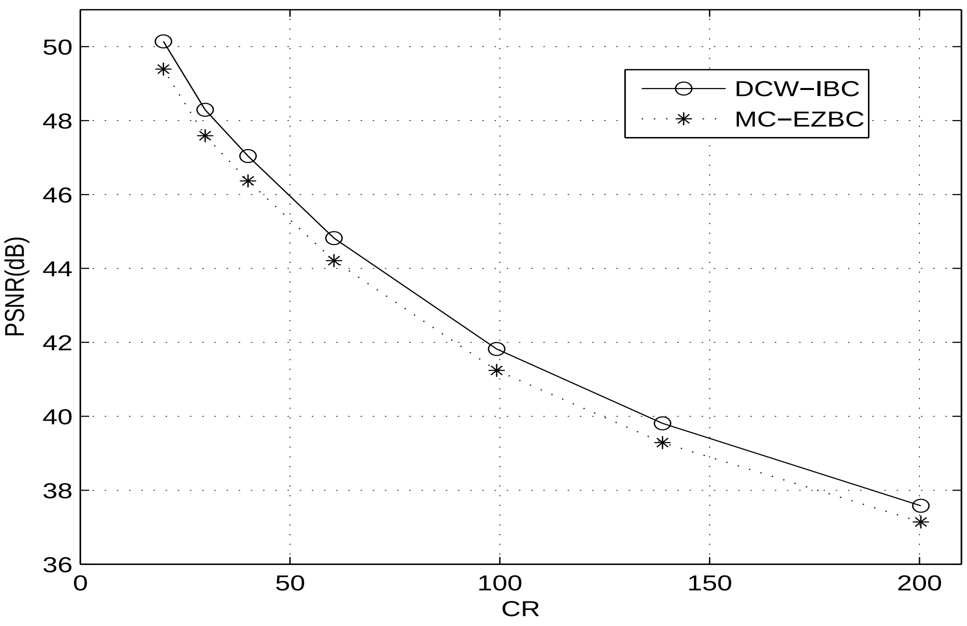


Fig. 5.35 PSNR versus compression ratio of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Corridor*.

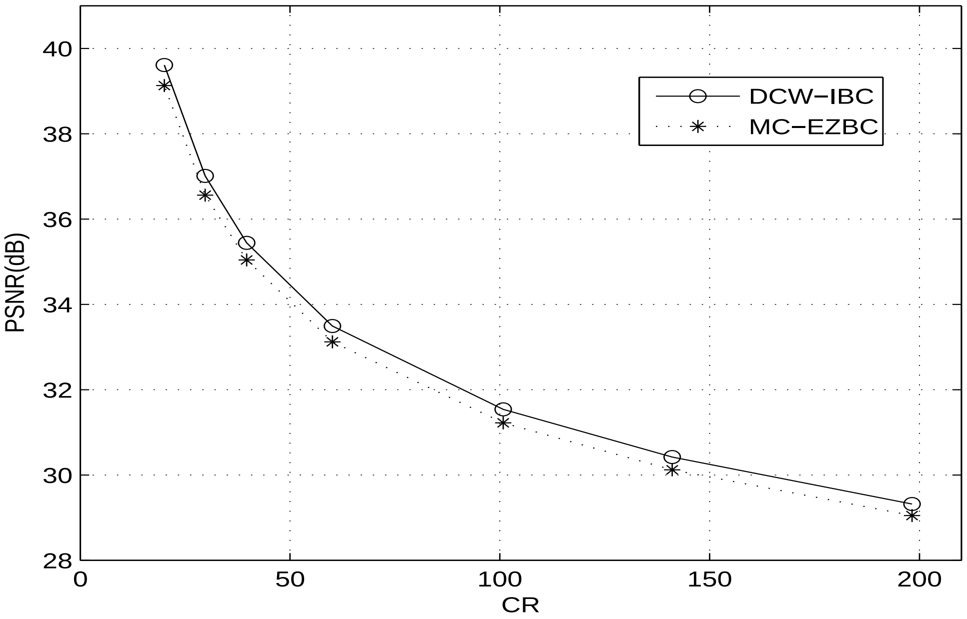


Fig. 5.36 PSNR versus compression ratio of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Campus*

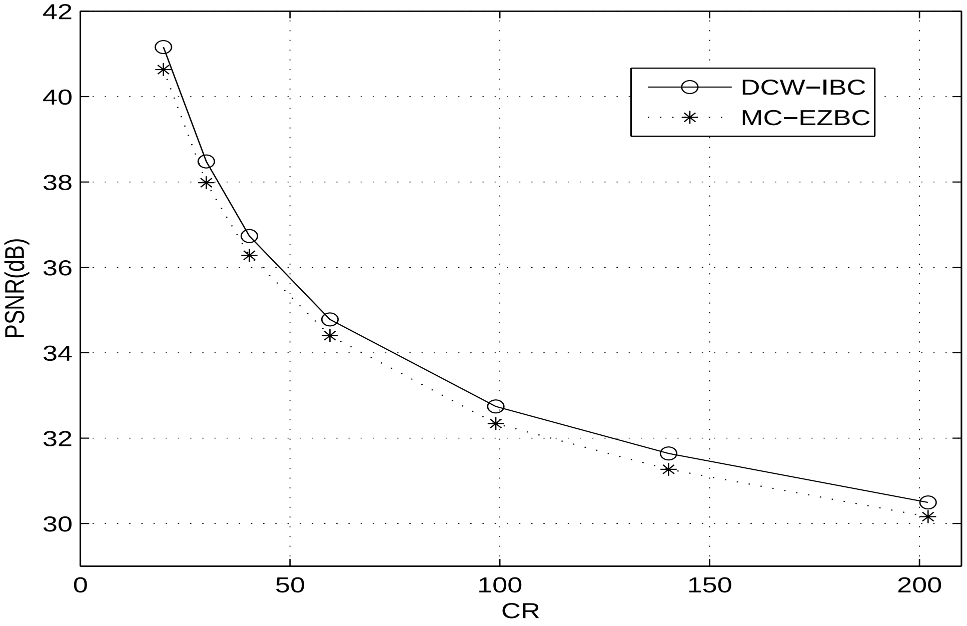


Fig. 5.37 PSNR versus compression ratio of *DCW-IBC* and *MC-EZBC* applied to encoding cubic-panorama image dataset *Marie-Curie*.

Chapter 6

Global Displacement Estimation and Compensation for Cubic-Panorama Image Dataset Compression

One of the main features of the image datasets used for image-based virtual environment navigation is that the dominant cross-image displacement in the image datasets demonstrates a certain regular pattern for each type of image dataset. This cross-image displacement feature, depending on the motion pattern of the camera generating the image datasets, could be utilized to effectively improve the coding performance of image dataset compression. Based on the dominant cross-image displacement feature of cubic-panorama image datasets, global displacement estimation and compensation for cubic-panorama image dataset compression is investigate in this chapter. A unique block-based scaled depth estimation technique is proposed for obtaining the scaled depth maps of the predicted images. A new global displacement-compensated image prediction algorithm combining the block-based scaled depth estimation technique is presented. By taking into consideration all kinds of camera motions including camera translations with parameter components perpendicular and parallel to the camera optical axis, a novel global displacement estimation and compensation approach is developed and adapted to the coding scheme for compressing cubic-panorama image datasets, resulting in enhanced coding efficiency of cubic-panorama image dataset compression.

This chapter is organized as following. The first section gives an introduction to global displacement estimation and displacement-compensated image prediction, followed by a review of the relevant research developments in global displacement estimation and compensation. After the first introductory section, a discussion about global displacement analysis is conducted in Section 6.2. The issue of displacement-compensated image prediction is dealt with in Section 6.3. Global displacement with block-based scaled depth estimation is discussed and an image prediction algorithm with global displacement estimation is proposed. The features of global displacements in cubic-panorama image datasets are investigated and a coding scheme for cubic-panorama image dataset compression with global displacement compensation is presented in Section 6.4. Experimental results of the coding scheme are demonstrated in Section 6.5. Finally, the discussion on global displacement estimation and compensation for cubic-panorama image dataset compression is summarized in Section 6.6.

6.1 Introduction

Displacement-compensated image prediction is an important component in image sequence compression systems. With the aid of displacement estimation and compensation, it derives the required predicting image as close as possible to the to-be-encoded predicted image, and thereafter enables efficient image sequence compression by exploiting the great deal of redundancy across images. On the other hand, global displacement estimation, which is usually attributed to determining the displacement model parameters associated with the motion of the camera capturing the image sequence, can lead to compact representations of displacement vector fields. Therefore, the combination of displacement-compensated image prediction with global displacement estimation has the potential of significantly improving the compression performance if applied to image dataset compression.

There are quite a few global displacement estimation techniques which are well developed and can be applied for global displacement-compensated image prediction. Usually, for global displacement estimation and compensation, the effect of camera motions is modeled with a number of global displacement parameters. For example, four-parameter global displacement models are used for representing the associated camera motions in [13, 19, 97]; a five-parameter model is used in [48]; six-parameter models in [26, 103]; a seven-parameter model in [97] and eight-parameter models in [18, 77]. Generally, global-displacement model parameters are determined by solving a minimization problem to best adapt the proposed global-displacement model to the available data derived from the image sequences.

Global displacement estimation techniques can be classified by the types of camera motions covered in the models. A linear global displacement model is employed in [13] for representing the global image displacement associated with camera zoom and pan. An iterative least-squares estimation *(ILSE)* algorithm is applied for obtaining the model parameters based on the determined local block displacement vectors. By keeping a symmetrical block structure in each iteration step, it decreases the computational complexity without sacrificing the estimation accuracy. The global displacement models in [19, 48, 97] cover not only camera zoom and pan, but also camera tilt and swing. In order to reduce computational complexity, Zakhor and Lari [97] presented an algorithm accomplished by matching the corresponding locations of one-bit binary edges instead of eight-bit intensity blocks. To overcome the problem of global displacement estimation suffering from feature matching errors, local displacement disturbances and measurement noises, Kim [48] proposed a recursive total least-squares *(RTLS)* algorithm for image sequences in noisy situations. Evans et al. [19] described a computationally efficient technique for dominant global image displacement estimation. A histogram-type approach is incorporated by generating a ranked list of the estimated model parameters and averaging the parameter values close to the median.

A six-parameter affine model is employed in [26, 103]. He et al. [26] presented a fast global-displacement estimation approach. A three-level multi-resolution pyramid of the input images is built. A Gauss-Newton iterative method is used for initial parameter estimation at the top level, and a Levenberg-Marquardt optimizing method is applied at the intermediate and base levels to refine the estimation. Zhang et al. [103] proposed a new global-displacement estimation algorithm by segmenting images into foreground and background portions. Only the displacement vectors associated with the background may be used for global displacement estimation.

Camera translations are rarely handled in the developed global-displacement estimation techniques. In the case of global image displacements associated only with camera zoom, pan, tilt, and swing, the global displacement parameters alone are adequate for deducing the locations of corresponding pixels. However, if a camera translation occurs, the depth information of the scenes is required in addition to the global displacement parameters to determine the relationship between the locations of corresponding pixels in successive images. With an effort to involve camera translations in the model, a robust and fast approach for global displacement estimation based on an eight-parameter perspective model is presented in [18]. It is applicable to the scenes able to be approximated by a planar surface for covering camera translations. A camera motion estimation technique for MPEG video is described in [20]. The camera translation parallel to the image plane is approximated by simply averaging the estimated rotation displacement field. However, the camera translation along the optical axis direction is not treated in this technique.

In this chapter, based on a perspective projection model for global image displacement analysis, a new global displacement-compensated image prediction algorithm [45] is presented, taking into consideration all types of camera motions involving camera translations with parameter components perpendicular and parallel to the camera axis. A unique block-based scaled depth estimation technique is incorporated in the image prediction algorithm. The global displacement-compensated predicting image is generated by using the rotational and translational model parameters combined with the estimated scaled block-depth map. A novel global displacement estimation and compensation approach is developed for cubic-panorama image dataset compression, resulting in improved coding performance.

6.2 Global Displacement Analysis

This section is devoted to the discussion about global displacement analysis. A perspective projection model for global image displacement estimation and compensation is presented, covering all kinds of motions of the camera in capturing an image sequence. Based on this analysis model, the formulae of the image coordinate transformations and the global displacement estimation are derived.

6.2.1 Perspective Projection Model

The perspective projection model used for the analysis of global displacement estimation and compensation is shown in Fig. 6.1. By choosing the focal point of the camera as the origin of the coordinate system, the camera optical axis as the **Z** axis and the image plane parallel to the **XOY** plane, the perspective projection relationship results in

$$X = F \cdot x/z, \qquad Y = F \cdot y/z, \tag{6.1}$$

where $(x,\ y,\ z)$ denotes the coordinate of a scene point, F presents the focal length of the camera and $(X,\ Y)$ denotes the coordinate of the scene point projected on the image plane of the camera.

With this analysis model, the predicted current image can be regarded as an image captured by the camera undergoing a motion, which is a combination of a translation with a rotation relative to the position for capturing the reference image. A generic translation can be represented by a translation column vector $\mathbf{T} = (T_x, T_y, T_z)^T$, and a generic rotation can be described by three successive rotations of angles θ, ψ, φ around

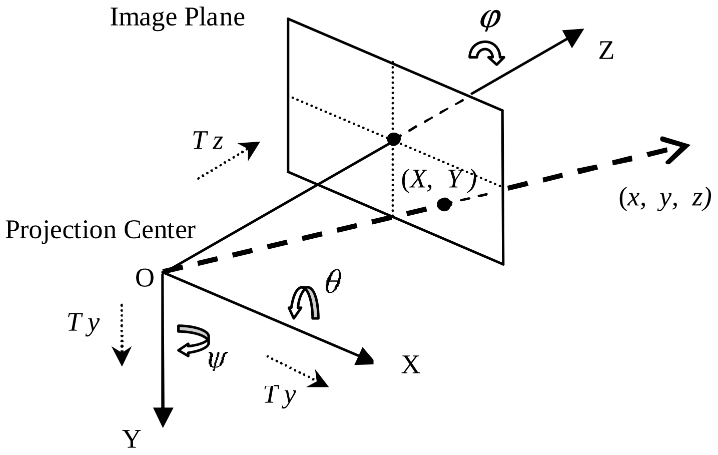


Fig. 6.1 Perspective projection model used for the analysis of global displacement estimation and compensation.

the **X, Y** and **Z** axes respectively. So, the global image displacement associated with this camera motion is modeled with a parameter set $\mathbb{P}_M = \{T_x, T_y, T_z, \theta, \psi, \varphi\}$ consisting of six translational and rotational global displacement model parameters.

The coordinates of a static scene point with respect to the coordinate system with its origin fixed at the focus of the camera hold the relationship [29]

$$(x_r, y_r, z_r)^T = \mathbf{R} \cdot (x_c, y_c, z_c)^T + \mathbf{T}, \tag{6.2}$$

where (x_r, y_r, z_r) denotes the coordinate of a scene point before the camera motion, and (x_c, y_c, z_c) denotes its coordinate after the camera motion. The rotation matrix $\mathbf{R}$ can be represented with respect to the rotation angles θ, ψ, φ as [29]

$$\begin{aligned}\mathbf{R} &= (r_{ij})_{3\,3} \\ &= \begin{pmatrix} \cos\varphi\cos\psi & \cos\varphi\sin\psi\sin\theta - \sin\varphi\cos\theta & \cos\varphi\sin\psi\cos\theta + \sin\varphi\sin\theta \\ \sin\varphi\cos\psi & \sin\varphi\sin\psi\sin\theta + \cos\varphi\cos\theta & \sin\varphi\sin\psi\cos\theta - \cos\varphi\sin\theta \\ -\sin\psi & \cos\psi\sin\theta & \cos\psi\cos\theta \end{pmatrix}.\end{aligned} \tag{6.3}$$

In the case that the camera experiences a motion of a translation combined with a tiny rotation, $\sin\theta \approx \theta$, $\sin\psi \approx \psi$, $\sin\varphi \approx \varphi$; $\cos\theta \approx \cos\psi \approx \cos\varphi \approx 1$. Then, the rotation matrix $\mathbf{R}$ becomes

$$\mathbf{R} \approx \begin{pmatrix} 1 & -\varphi & \psi \\ \varphi & 1 & -\theta \\ -\psi & \theta & 1 \end{pmatrix}. \tag{6.4}$$

6.2.2 Image Coordinate Transformation

With (X_r, Y_r) and (X_c, Y_c) denoting the coordinates of a scene point projected on the camera image plane before and after the camera undergoes a motion respectively, the expressions of X_r, Y_r as functions of X_c, Y_c together with the determined model parameter set $\mathbb{P}_M = \{T_x, T_y, T_z, \theta, \psi, \varphi\}$ or the corresponding rotation matrix $\mathbf{R}$ and translation vector $\mathbf{T}$ need to be derived for image coordinate transformations:

$$\begin{aligned} X_r &= f_x\big(\,(X_c, Y_c) \mid {\scriptstyle \mathbb{P}_M}\big), \\ Y_r &= f_y\big(\,(X_c, Y_c) \mid {\scriptstyle \mathbb{P}_M}\big). \end{aligned} \tag{6.5}$$

The model parameters or $\mathbf{R}$ and $\mathbf{T}$ can be obtained by replacing the coordinates of image correspondence points into the following equation [92]

$$(X_r, Y_r, F) \cdot \mathbf{E} \cdot (X_c, Y_c, F)^T = 0 \tag{6.6}$$

where the essential matrix $\mathbf{E}$, expressed with the components of the translation vector $\mathbf{T}$ and the elements of the rotation matrix $\mathbf{R}$, is given by

$$\mathbf{E} = (e_{ij})_{\,3\,3} = \begin{pmatrix} T_z \cdot r_{21} - T_y \cdot r_{31} & T_z \cdot r_{22} - T_y \cdot r_{32} & T_z \cdot r_{23} - T_y \cdot r_{33} \\ T_x \cdot r_{31} - T_z \cdot r_{11} & T_x \cdot r_{32} - T_z \cdot r_{12} & T_x \cdot r_{33} - T_z \cdot r_{13} \\ T_y \cdot r_{11} - T_x \cdot r_{21} & T_y \cdot r_{12} - T_x \cdot r_{22} & T_y \cdot r_{13} - T_x \cdot r_{23} \end{pmatrix}. \tag{6.7}$$

While the rotation matrix **R** can be determined from the essential matrix **E**, the translation column vector **T** can only be obtained to within a scaling factor α, as **E** can only be determined up to a scaling factor: $\mathbf{T} = \alpha \cdot \Delta\mathbf{T} = \alpha \cdot (\Delta x, \Delta y, \Delta z)^T$. Using (6.1) and (6.2), X_r and Y_r can be represented by utilizing X_c, Y_c as well as the rotational and translational parameters as

$$\begin{aligned} X_r &= F \cdot \frac{(r_{11} \cdot X_c + r_{12} \cdot Y_c + r_{13} \cdot F) \cdot z_c + \alpha F \cdot \Delta x}{(r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F) \cdot z_c + \alpha F \cdot \Delta z}, \\ Y_r &= F \cdot \frac{(r_{21} \cdot X_c + r_{22} \cdot Y_c + r_{23} \cdot F) \cdot z_c + \alpha F \cdot \Delta y}{(r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F) \cdot z_c + \alpha F \cdot \Delta z}. \end{aligned} \tag{6.8}$$

6.2.3 Displacement Estimation Formulae

For an image sample located at (X_c, Y_c) in the current image, the corresponding displacement vector is $\mathbf{D} = (D_x, D_y) = (X_r - X_c, Y_r - Y_c)$ relative to its matching image sample at (X_r, Y_r) in the reference image. Based on the image coordinate transformation equations derived in last section, the horizontal and vertical components D_x, D_y of the displacement vector **D** are given by

$$\begin{aligned} D_x &= \frac{[X_c F(r_{11} - r_{33}) + Y_c(F r_{12} - X_c r_{32}) + F^2 r_{13} - X_c^2 r_{31}] z_c + \alpha F(F\Delta x - X_c \Delta z)}{(r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F) \cdot z_c + \alpha F \cdot \Delta z}, \\ D_y &= \frac{[Y_c F(r_{22} - r_{33}) + X_c(F r_{21} - X_c r_{31}) + F^2 r_{23} - Y_c^2 r_{32}] z_c + \alpha F(F\Delta y - Y_c \Delta z)}{(r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F) \cdot z_c + \alpha F \cdot \Delta z}. \end{aligned} \tag{6.9}$$

In the case that no camera translation is involved in the camera motion, the expressions of the two components of the displacement vector can then be simplified as

$$\begin{aligned} D_x &= \frac{X_c F(r_{11} - r_{33}) + Y_c(F r_{12} - X_c r_{32}) + F^2 r_{13} - X_c^2 r_{31}}{r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F}, \\ D_y &= \frac{Y_c F(r_{22} - r_{33}) + X_c(F r_{21} - X_c r_{31}) + F^2 r_{23} - Y_c^2 r_{32}}{r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F}. \end{aligned} \tag{6.10}$$

As shown in (6.9) and (6.10), the displacement estimation formulae include the depth information z_c of the scene due to camera translations, and could be significantly simplified in the case that no camera translation occurs. Most existing global displacement estimation algorithms deal with global image displacements corresponding to camera zoom and rotations (pan, tilt and swing), avoiding the more challenging issue related to camera translations and the scene depth distribution. New global displacement estimation algorithms need to be developed in the case that camera translations occur and cannot be ignored in displacement estimation.

6.3 Global Displacement-Compensated Image Prediction

A new global displacement-compensated image prediction algorithm is presented, taking into consideration all kinds of camera motions involving camera translations with parameter components perpendicular and parallel to the camera optical axis. A unique block-based scaled depth estimation technique is proposed for global displacement estimation and compensation. Global displacement-compensated image prediction is performed by using scaled block-depth estimates and global displacement model parameters.

6.3.1 Global Displacement Estimation (GDE)

Firstly, an image matching process is performed to obtain a number of image correspondence pairs located at $(X_{r,\,m}, Y_{r,\,m})$ and $(X_{c,\,m}, Y_{c,\,m})$ $(m \in \{1, 2, ..., M\}, M \in \mathbb{Z})$ in the reference image and the predicted current image respectively. As these coordinates of the correspondence pairs are to be used in the following global displacement parameter estimation, a sufficient number M of correspondence pairs are needed to obtain robust model parameter estimates. A variety of image matching techniques can be employed based on their effectiveness and efficiency in performing feature point selection, detection and matching.

The determination of the essential matrix $\mathbf{E}$ is a typical optimization process of estimation error minimization. By replacing the coordinates of the correspondence pairs into the image coordinate transformation equation (6.6), the sum of the squared errors is obtained by the following expression

$$\begin{aligned} S_E = \sum_{m=1}^{M} (X_{r,\,m} X_{c,\,m} e_{11} + X_{r,\,m} Y_{c,\,m} e_{12} + X_{r,\,m} F e_{13} + Y_{r,\,m} X_{c,\,m} e_{21} + \\ Y_{r,\,m} Y_{c,\,m} e_{22} + Y_{r,\,m} F e_{23} + X_{c,\,m} F e_{31} + Y_{c,\,m} F e_{32} + F^2 e_{33})^2. \end{aligned} \tag{6.11}$$

A linear least-squares estimation algorithm is applied by solving a group of equations resulting from $\dfrac{\partial S_E}{\partial e_{ij}} = 0$ $(i \in \{1,2,3\},\ j \in \{1,2,3\})$ to determine the values of the elements in the essential matrix $\mathbf{E}$.

The translation column vector $\mathbf{T} = (T_x, T_y, T_z)^T = \alpha \cdot (\Delta x, \Delta y, \Delta z)^T$ can be obtained from the resultant essential matrix $\mathbf{E}$. By solving the homogeneous equations corresponding to $\mathbf{E}^T \mathbf{T} = 0$ [91], we have

$$\begin{aligned} \Delta x &= \frac{e_{21}e_{32} - e_{22}e_{31}}{[(e_{22}e_{31} - e_{21}e_{32})^2 + (e_{11}e_{32} - e_{12}e_{31})^2 + (e_{11}e_{22} - e_{12}e_{21})^2]^{1/2}}, \\ \Delta y &= \frac{e_{12}e_{31} - e_{11}e_{32}}{[(e_{22}e_{31} - e_{21}e_{32})^2 + (e_{11}e_{32} - e_{12}e_{31})^2 + (e_{11}e_{22} - e_{12}e_{21})^2]^{1/2}}, \\ \Delta z &= \frac{e_{11}e_{22} - e_{12}e_{21}}{[(e_{22}e_{31} - e_{21}e_{32})^2 + (e_{11}e_{32} - e_{12}e_{31})^2 + (e_{11}e_{22} - e_{12}e_{21})^2]^{1/2}}, \end{aligned} \tag{6.12}$$

while $\Delta x, \Delta y, \Delta z$ satisfy the normalization condition

$$(\Delta x)^2 + (\Delta y)^2 + (\Delta z)^2 = 1. \tag{6.13}$$

The rotation parameters of the global displacement model are derived from the essential matrix $\mathbf{E}$ with the singular value decomposition *(SVD)* given by

$$\mathbf{E} = \mathbf{U} \cdot \Xi \cdot \mathbf{V}^T. \tag{6.14}$$

Here, Ξ is a diagonal matrix of the same dimension as $\mathbf{E}$ with nonnegative diagonal elements in decreasing order; $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices. Then, the desired rotation matrix is obtained by

$$\mathbf{R} = \mathbf{U} \cdot \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & s \end{pmatrix} \cdot \mathbf{V}^T, \text{ or } \mathbf{R} = \mathbf{U} \cdot \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & s \end{pmatrix} \cdot \mathbf{V}^T, \qquad (6.15)$$

where $s = \det(\mathbf{U}) \cdot \det(\mathbf{V})$. The rotation angles θ, ψ, φ can be determined upon request by using (6.3) directly from the rotation matrix $\mathbf{R}$.

Outlier removal is involved in the model parameter estimation process to obtain more accurate parameter estimates through discarding 10% of the correspondence pairs that have bigger coordinate differences than others at each iterate step. Then, the model parameters are re-computed with the remaining correspondence pairs. In this way, the correspondence pairs are refined and the model parameters are compute recursively until the model parameters converge, that is, no significant value change of the model parameter set from the last set.

6.3.2 Block-Based Scaled-Depth Estimation

In the more generic case that the camera experiences a motion involving a translation, global displacement estimation and compensation cannot be completed by just using the global displacement model parameters. Also, the scene depth information is needed for displacement estimation and compensation. A block-based scaled depth estimation technique is proposed to obtain the scene depth distribution map for the predicted image based on (6.8).

As the translation column vector $\mathbf{T}$ can only be determined up to a scale factor α based on the essential matrix $\mathbf{E}$, we combine the unknown depth z_c and the scale factor α together and call $\mathbf{d}_s = z_c/\alpha$ the scaled depth. Replacing the newly-defined scaled depth $\mathbf{d}_s$ into (6.8), we obtain

$$\begin{aligned} X_r &= F \cdot \frac{(r_{11} \cdot X_c + r_{12} \cdot Y_c + r_{13} \cdot F) \cdot \mathbf{d}_s + F \cdot \Delta x}{(r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F) \cdot \mathbf{d}_s + F \cdot \Delta z}, \\ Y_r &= F \cdot \frac{(r_{21} \cdot X_c + r_{22} \cdot Y_c + r_{23} \cdot F) \cdot \mathbf{d}_s + F \cdot \Delta y}{(r_{31} \cdot X_c + r_{32} \cdot Y_c + r_{33} \cdot F) \cdot \mathbf{d}_s + F \cdot \Delta z}. \end{aligned} \tag{6.16}$$

The current image is partitioned into rectangular blocks $\mathbf{B}_{c,\,n}$ ($n \in \{1, 2, ..., N\}, N \in \mathbb{Z}$) of image samples. The scaled depth values of all the image samples in a block are approximated with a single value $\mathbf{d}_{s,\,n}$ called the scaled block-depth. This scaled block-depth needs to be estimated to determine the locations of the predicting block in the reference image for image prediction.

The scaled block-depth estimation is a procedure of searching for a minimum of the block prediction error with respect to the scaled block-depth $\mathbf{d}_{s,\,n}$:

$$\min_{\mathbf{d}_{s,\,n}} \sum_{(X_c,\,Y_c) \in \mathbf{B}_{c,\,n}} \left[\mathcal{I}_{\mathcal{B},\,r}(X_r, Y_r) - \mathcal{I}_{\mathcal{B},\,c}(X_c, Y_c)\right]^2, \tag{6.17}$$

where $\mathcal{I}_{\mathcal{B},\,r}(X_r, Y_r)$ is the sample image intensity in the reference image at (X_r, Y_r) and $\mathcal{I}_{\mathcal{B},\,c}(X_c, Y_c)$ the sample image intensity in the current image at (X_c, Y_c). A direct Nelder-Mead simplex search method [49] for finding a local minimum of the block prediction error with respect to the scaled block-depth $\mathbf{d}_{s,\,n}$ is applied after replacing (6.16) into (6.17). This refers to an unrestrained nonlinear optimization process. For each sample block, the optimized search for scaled block-depth starts with an initial value equal to the previously determined $\mathbf{d}_{s,\,n}$ of an immediate neighbor block.

6.3.3 Image Prediction Algorithm

The displacement-compensated predicting image is generated by utilizing the global displacement parameters combined with the estimated scaled block-depth map. The estimates of the scaled block-depths are substituted into the coordinate transformation formulae (6.16). Together with the determined global rotational and translational model parameters, the locations of the predicting pixels in the reference image are obtained. The intensity values of the predicting pixels are derived by pixel interpolation, as the predicting pixels obtained from the transformation formulae may not be located on the sampling grids even though all the pixels in the predicted image are on the sampling grids. Robust and reliable model parameters and effective block-depth estimation are necessary for precise displacement-compensated image prediction.

Compared with other image prediction techniques, more accurate predicting images can be obtained by using the developed global displacement-compensated image prediction with scaled block-depth estimation. As a result of fewer parameters required for global displacement compensation and reduced image prediction residues, the developed approach is more efficient when applied for image dataset compression.

In summary, the algorithm of the global displacement-compensated image prediction consists of the following steps:

- Perform image matching between the reference image and the predicted current image to obtain a sufficient number of correspondence pairs;

- Compute the global displacement model parameters by replacing the coordinates of correspondence pairs into the image transformation equation and refine the resulting parameters through outlier removal;

- Perform scaled block-depth estimation by substituting the determined global displacement parameters into the coordinate transformation formulae and obtaining the optimized value of the scaled block-depth for each block in the predicted image so that a minimum *MSD* can be achieved between the reference block and the predicted block;

- Determine the locations of displacement-compensated predicting pixels corresponding to each block in the predicted image by making use of the perspective projection transformation formulae with the optimized scaled block-depth and the global displacement model parameters;

- Interpolate the intensity values of the global displacement-compensated predicting pixels based on the pixel values on the sample grids in the reference image;
- Generate the global displacement-compensated predicting image by combining the interpolated pixel intensity values in all the predicting blocks associated with all the blocks of the predicted current image.

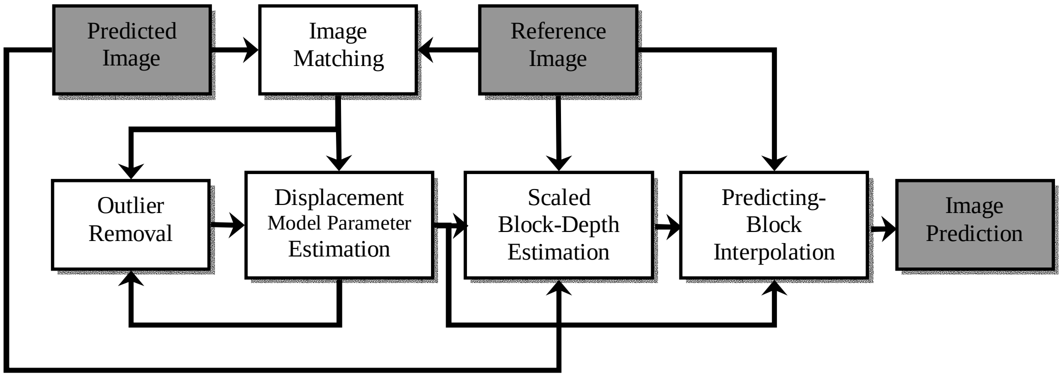

Predicted Image
Image Matching
Reference Image
Outlier Removal
Displacement Model Parameter Estimation
Scaled Block-Depth Estimation
Predicting-Block Interpolation
Image Prediction

Fig. 6.2 Image prediction algorithm with scaled block-depth estimation.

6.4 Cubic-Panorama Image Dataset Compression with Global Displacement Compensation

In this section, the features of global image displacements in cubic-panorama image datasets are investigated. Cubic-panorama basis images are adapted to the perspective projection model for global displacement compensation, and a coding scheme for cubic-panorama image dataset compression with global displacement estimation and compensation is presented.

6.4.1 Global Displacement Features of Cubic-Panorama Image Datasets

A basis image $\mathcal{I}_{\mathcal{B},\,k}$ $(k \in \{1, 2, ..., K\},\ K \in \mathbb{Z})$ in a cubic-panorama image dataset $\tilde{\mathbb{S}}_D$ consists of six side images $\mathcal{I}_{\mathcal{B},\,k} = \left\{\mathcal{I}_{\mathcal{B},\,kw} \middle| \forall w \in \{u,b,l,f,r,d\}\right\}$. Each side image $\mathcal{I}_{\mathcal{B},\,kw}$ is projected on one face of the cube and can be regarded as a standard perspective planar image with a 90^o field of view in both the image height and the image width directions parallel to each side image plane. The perspective projection model introduced in 6.2.1. is suitable for describing the global displacement features in each of the side images of the cubic-panorama basis images.

Suppose that each side image as a perspective planar image is taken by a separate camera. Altogether there are six such virtual cameras utilized for respectively producing the six side images of cubic-panorama basis images. As shown in Fig. 6.3, the center point of the cube serves as the common projection center of the six side images and locates the focal points of all six cameras at a common spot. A basis coordinate system

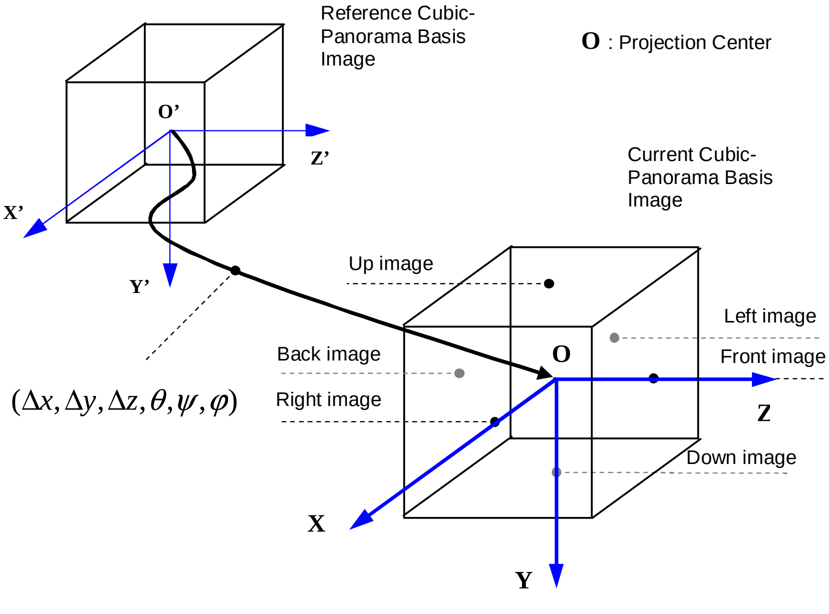

Reference Cubic-Panorama Basis Image
O : Projection Center
O'
Z'
X'
Y'
Current Cubic-Panorama Basis Image
Up image
Left image
Back image
O
Front image
$(\Delta x, \Delta y, \Delta z, \theta, \psi, \varphi)$
Right image
Z
Down image
X
Y

Fig. 6.3 Global displacement analysis across basis images of a cubic-panorama image dataset in the basis coordinate system.

is utilized with its origin located at the cube center and the **Z** axis going through the image plane center of the front side image $\mathcal{I}_{\mathcal{B},\,kf}$. Each camera undergoes a movement from the position where it produces the reference side image $\mathcal{I}_{\mathcal{B},\,rw}$ to the next position where it produces the current side image $\mathcal{I}_{\mathcal{B},\,cw}$. As six side images are bound together to make up a cubic-panorama basis image, the six cameras experience the same movement from one position to another to generate the next cubic-panorama basis image. This movement causes global image displacement across basis images and can be modeled with a set of six parameters $\mathbb{P}_D = \{\Delta x, \Delta y, \Delta z, \theta, \psi, \varphi\}$ as discussed previously in this chapter. Generally, in a practical process of acquiring the raw image sequences, the trajectory of cameras is well planned on a horizontal plane parallel to the coordinate system plane **XOZ**. The dominant global image displacements are mainly caused by the model parameters $\Delta x, \Delta z$ associated with the transverse movements of the cameras with small rotations in capturing the raw image sequences.

The model parameters are obtained by the approach proposed in the previous section. As the same set of model parameters is used for all side images of a basis image due to a common camera movement, image matching can be performed with side images on one cube face, the front cube face for example, by choosing enough number of correspondence feature point pairs to obtain the model parameters and apply the resultant model parameters in displacement-compensated image prediction for the whole basis image. Alternatively, all side images on the six cube faces can be used in their respective image matching process to obtain the model parameters separately. Then, the average values of the parameters are determined as more precise and reliable model parameters used in the following global image displacement compensation process.

6.4.2 Displacement Compensation in Cubic-Panorama Image Datasets

Despite the fact that six virtual cameras experience the same movement characterized by the parameter set $\mathbb{P}_D = \{\Delta x, \Delta y, \Delta z, \theta, \psi, \varphi\}$ to produce the side images making up a cubic-panorama basis image, the computations of image coordinate transformations and the displacement-compensated predicting images are different by applying these parameters to side images on each cube face. Side images on different cube faces have different orientations with respect to the common projection center. Localized coordinate systems are used separately to adapt the perspective projection model to side images on each cube face (Fig. 6.4). The same model parameter set still can bring about different computational results due to the different orientations of the localized coordinate systems. The model parameters in $\mathbb{P}_D = \{\Delta x, \Delta y, \Delta z, \theta, \psi, \varphi\}$ with respect to the basis coordinate system need to be converted to the corresponding normalized translation magnitudes along the coordinate axis directions and rotation angles around the coordinate axes with respect to each localized coordinate system. The resulting model parameters are shown in Table 6.1.

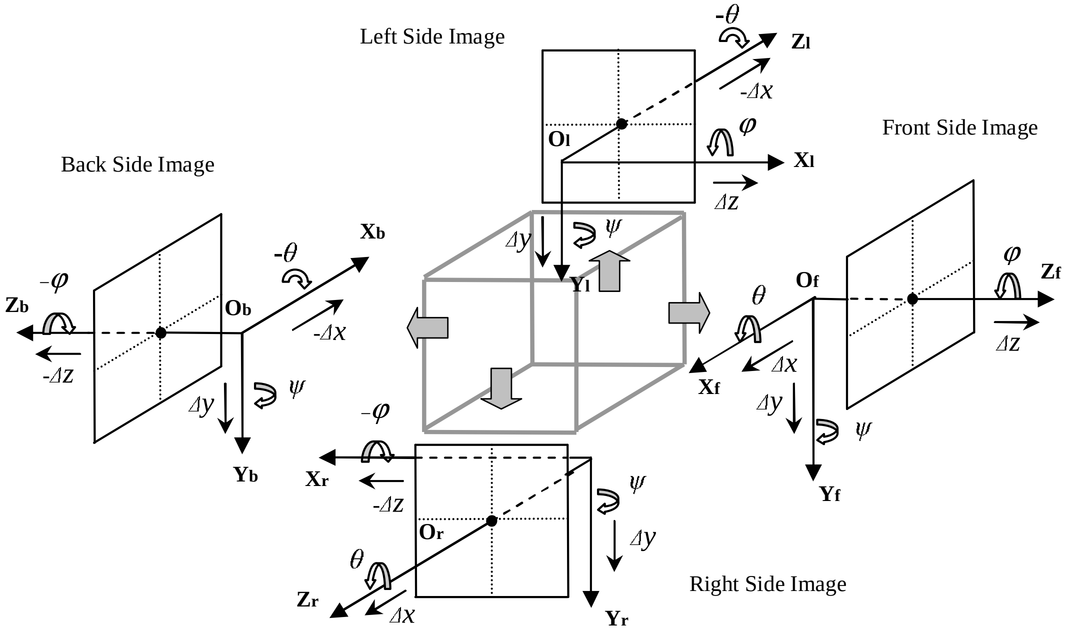


Fig. 6.4 Localized coordinate systems used to separately adapt the perspective projection model to side images on the cube faces.

For the purpose of determining image coordinate transformations and the displace-

Cube Face Image	Norm.[a] Translation (along axis directions)			Rotation Angle (around coordinate axes)		
	X	**Y**	**Z**	**X**	**Y**	**Z**
Up side	$-\Delta x$	$-\Delta z$	$-\Delta y$	$-\theta$	$-\varphi$	$-\psi$
Back side	$-\Delta x$	Δy	$-\Delta z$	$-\theta$	ψ	$-\varphi$
Left side	Δz	Δy	$-\Delta x$	φ	ψ	$-\theta$
Front side	Δx	Δy	Δz	θ	ψ	φ
Right side	$-\Delta z$	Δy	Δx	$-\varphi$	ψ	θ
Down side	Δx	$-\Delta z$	Δy	θ	$-\varphi$	ψ

[a] Normalized.

Table 6.1 Global displacement model parameters corresponding to the localized coordinate systems for the side images on the cube faces

ment compensated predicting images, the associated translation vector $\mathbf{T}_w\,(w \in \{u, b, l, f, r, d\})$ for the side images on each cube face corresponding to each localized coordinate system can be obtained in a straightforward manner from the corresponding normalized translation magnitudes $\Delta x, \Delta y, \Delta z$ in Table 6.1 with a proper order and proper signs, and the associated rotation matrix $\mathbf{R}_w\ (w \in \{u, b, l, f, r, d\})$ corresponding to each localized coordinate system can be derived based on the corresponding rotation angles θ, ψ, φ given in the table.

With the side images $\mathcal{I}_{\mathcal{B},\,kf}$ on the front face of the cube, the associated rotation matrix $\mathbf{R}_f = (r_{f,\,ij})_{\,3\,3}$ can be calculated by directly using formula (6.3) presented in Section 6.2.1. The rotation matrix $\mathbf{R}_b = (r_{b,\,ij})_{\,3\,3}$ for the side images $\mathcal{I}_{\mathcal{B},\,kb}$ on the back face of the cube can be obtained from $\mathbf{R}_f$ by the relationship equations: $r_{b,\,12} = -r_{f,\,12}$; $r_{b,\,21} = -r_{f,\,21}$; $r_{b,\,23} = -r_{f,\,23}$; $r_{b,\,32} = -r_{f,\,32}$ and $r_{b,\,ij} = r_{f,\,ij}$ for all other elements of $\mathbf{R}_b$.

For the side images $\mathcal{I}_{\mathcal{B},\,kl}$ on the left face of the cube, the associated rotation matrix $\mathbf{R}_l = (r_{l,\,ij})_{\,3\,3}$ is given by the following equation based on the rotation angles corresponding to the localized coordinate system for the left side images:

$$\mathbf{R}_l = \begin{pmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta & \sin\psi \\ \sin\varphi\sin\psi\cos\theta - \cos\varphi\sin\theta & \cos\varphi\cos\theta + \sin\varphi\sin\psi\sin\theta & -\sin\varphi\cos\psi \\ -\cos\varphi\sin\psi\cos\theta - \sin\varphi\sin\theta & \sin\varphi\cos\theta - \cos\varphi\sin\psi\sin\theta & \cos\varphi\cos\psi \end{pmatrix}. \tag{6.18}$$

Then, the associated rotation matrix $\mathbf{R}_r = (r_{r,\,ij})_{\,3\,3}$ for the side images $\mathcal{I}_{\mathcal{B},\,kr}$ on the right face of the cube can be obtained from $\mathbf{R}_l$ by the relationship equations: $r_{r,\,12} = -r_{l,\,12}$; $r_{r,\,21} = -r_{l,\,21}$; $r_{r,\,23} = -r_{l,\,23}$; $r_{r,\,32} = -r_{l,\,32}$ and $r_{r,\,ij} = r_{l,\,ij}$ for all other elements of $\mathbf{R}_r$.

With the up side images $\mathcal{I}_{\mathcal{B},\,ku}$, the associated rotation matrix $\mathbf{R}_u = (r_{u,\,ij})_{\,3\,3}$ can be derived from the corresponding rotation angles as following

$$\mathbf{R}_u = \begin{pmatrix} \cos\varphi\cos\psi & \cos\varphi\sin\psi\cos\theta + \sin\varphi\sin\theta & \cos\varphi\sin\psi\sin\theta - \sin\varphi\cos\theta \\ -\sin\psi & \cos\psi\cos\theta & \cos\psi\sin\theta \\ \sin\varphi\cos\psi & \sin\varphi\sin\psi\cos\theta - \cos\varphi\sin\theta & \sin\varphi\sin\psi\sin\theta + \cos\varphi\cos\theta \end{pmatrix}. \tag{6.19}$$

Then, the associated rotation matrix of the down side images $\mathcal{I}_{\mathcal{B},\,kd}$ can be obtained from $\mathbf{R}_u$: $\mathbf{R}_d = (r_{d,\,ij})_{\,3\,3}$ where $r_{d,\,12} = -r_{u,\,12}$; $r_{d,\,21} = -r_{u,\,21}$; $r_{d,\,23} = -r_{u,\,23}$; $r_{d,\,32} = -r_{u,\,32}$ and $r_{d,\,ij} = r_{u,\,ij}$ for all other elements of $\mathbf{R}_d$.

Based on the above obtained translation vectors $\mathbf{T}_w$ $(w \in \{u, b, l, f, r, d\})$ and the rotation matrices $\mathbf{R}_w\,(w \in \{u, b, l, f, r, d\})$, displacement vector fields can be determined for the predicted basis images of cubic-panorama image datasets with the assistance of scaled block-depth estimation by applying (6.17) while replacing (6.16) into the formula. After that, the global displacement compensation can be performed for cubic-panorama basis images by invoking (6.9) for displacement-compensated image prediction.

6.4.3 Compression Scheme with Global Displacement Compensation

A compression scheme with the global displacement estimation and compensation approach (*GDEC*) for cubic-panorama image dataset compression is depicted in Fig. 6.5. It is designed by modification to a standardized video compression scheme with incorporation of the global displacement estimation and compensation. This compression scheme is based on the framework of block-based hybrid coding of displacement-compensated cross-image prediction combined with spatial transformation.

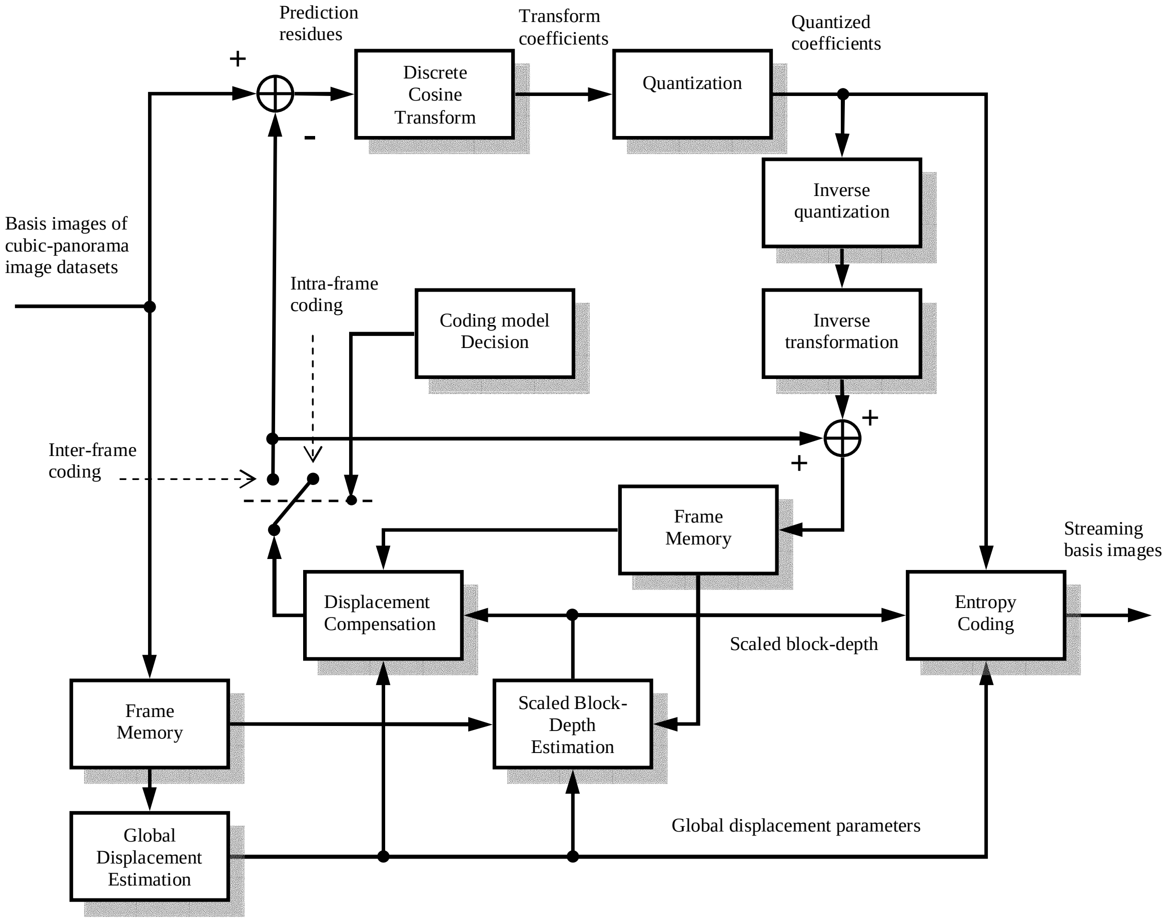


Fig. 6.5 Coding scheme with global displacement estimation and compensation (*GDEC*) for cubic-panorama image dataset compression.

By making use of a generic standard video coding scheme to build a basis structure to incorporate the proposed *GDEC* approach into the block-based hybrid coding framework, this compression system takes the basis image input from cubic-panorama image datasets. The coding model decision component divides the input basis images into intra reference images and inter prediction images. Reference basis images are intra-coded by bypassing the inter prediction step. Global displacement estimation is performed for each predicted basis image by image matching between the input predicted current basis image and an input reference basis image stored in the input frame memory. Then, the resultant global displacement parameters are applied to scaled block-depth estimation using the input predicted basis image taken from the input frame memory and a reconstructed reference basis image taken from the frame memory located in the prediction loop. After that, displacement compensation is performed by making use of the reconstructed reference basis image, the global displacement model parameters as well as the scaled block-depth estimates to generate a predicting image for the current predicted input basis image.

The image redundancies across basis images are removed through the cross-image prediction enhanced by global displacement compensation. The prediction residues of the input predicted basis image and the displacement-compensated predicting image are feeded into the spacial discrete cosine transform component to decorrelate image samples in the spatial domain. Then, the resulting transform coefficients are scaled in the quantization component and the resultant quantized transform coefficients are sent to the entropy coding component to generate the streaming basis image output. Meanwhile, the quantized transform coefficients are also sent to the displacement-compensated prediction loop for producing the reconstructed reference images. In the entropy coding process, the global displacement model parameters and the scaled block-depth maps are encoded and combined with the coded image sample coefficients into an output image dataset bitstream.

6.5 Experimental Results

The proposed coding scheme depicted in Fig. 6.5 for cubic-panorama image dataset compression is implemented in a software-based codec obtained by modification to a baseline codec of H.264/MPEG-4 AVC Reference Software [38] to incorporate the global displacement estimation and compensation (*GDEC*) approach into a block-based hybrid coding scheme for cubic-panorama image dataset compression. Experiments with testing cubic-panorama image datasets were carried out to verify the effectiveness of the designed coding scheme with *GDEC* applied for cubic-panorama image dataset compression. The experimental results are compared with those of the baseline H.264/MPEG-4 AVC reference software [38] with the traditional block matching algorithms (*BMA*) commonly used in all standardized video sequence coding schemes.

Resulting from the global displacement estimation process, an example of the global displacement parameter estimates for the predicted basis images in a group of 12 basis images of the testing cubic-panorama image dataset *Lab* (Fig. 4.13 and Table 4.1) is shown in Table 6.2. It can be observed that the dominant normalized translation is along the **Z** axis with respected to the basis coordinate system attached directly to the front side image, and the rotation angles along the coordinate axes are quite small. This is the typical situation for cubic-panorama image datasets especially for the indoor cubic-panorama image datasets captured along a segment of a well planned image acquisition trajectory. Although these parameters demonstrate a dominant translation along the camera axis direction directly applicable to the front side images, the dominant translations of other side images may be along the directions perpendicular to the camera axis direction because of the different orientations of the side images. Thus, the complicated translation situation is revealed in cubic-panorama image datasets.

Using the block-based hybrid coding scheme with *GDEC* applied to encoding the testing cubic-panorama image datasets, the experimental results of the average PSNR for the luminance component **Y** of the reconstructed basis images versus the bitstream

Predicted Basis Image	Translation Magnitude (normalized values)			Rotation Angle (in degrees)		
	Δx	Δy	Δz	θ	ψ	φ
P01	-0.0345	-0.0868	0.9956	0.0917	-0.0401	0.0172
P02	0.0456	-0.0166	0.9988	-0.0573	0.2235	-0.0859
P03	-0.0098	-0.0900	0.9959	0.0401	-0.1089	0.0516
P04	0.0053	-0.0219	0.9997	-0.1261	-0.1604	-0.0172
P05	0.0331	-0.0333	0.9989	-0.0286	-0.1089	-0.1031
P06	-0.0062	-0.0927	0.9957	0.1432	-0.2292	-0.0458
P07	-0.0080	-0.0257	0.9996	-0.1948	-0.3953	0.0172
P08	0.0214	-0.0350	0.9992	-0.1203	-0.0516	-0.1261
P09	0.0205	-0.0301	0.9993	-0.1833	-0.0057	0.0401
P10	0.0108	-0.0405	0.9991	-0.1146	-0.1891	-0.0401
P11	0.0123	-0.0297	0.9995	-0.1719	-0.2693	0.0749

Table 6.2 Global displacement parameter estimates for a group of predicted basis images in experimental cubic-panorama image dataset *Lab*

rate are shown in Fig. 6.6 for cubic-panorama image dataset *Lab*, and in Fig. 6.7 for cubic-panorama image dataset *Corridor* (Fig. 4.14 and Table 4.1). Meanwhile, for the comparison purpose, experimental results of the same compression scheme structure with the traditional *BMA* in place of *GDEC* applied to the same testing cubic-panorama image datasets are provided, and shown in these two figures. A typical exhaustive search in a coarse-to-fine manner over a fixed-size window is used in the traditional *BMA*.

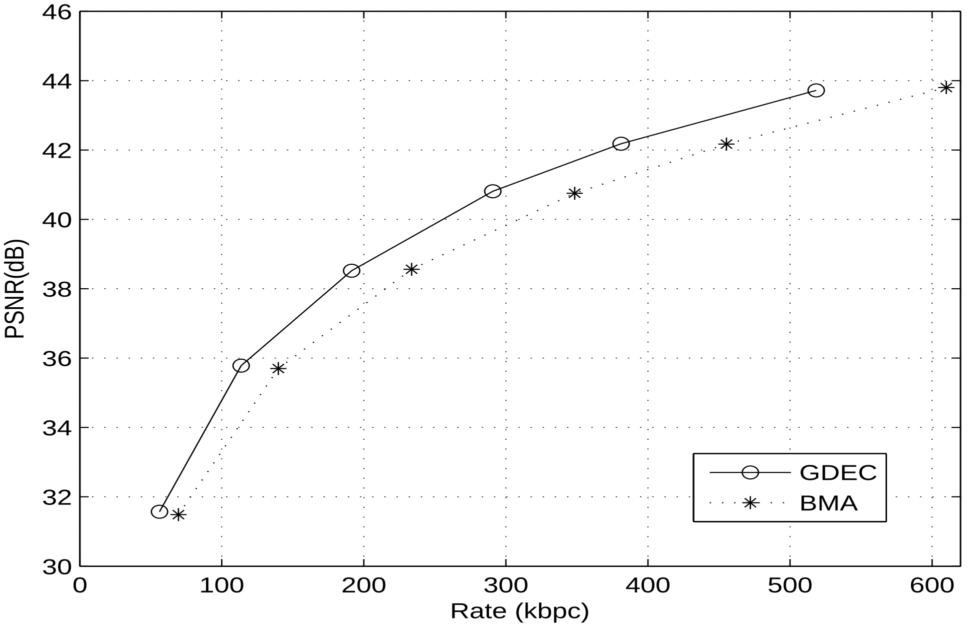


Fig. 6.6 Compression performance comparison of *GDEC* and *BMA* applied to encoding cubic-panorama image dataset *Lab*.

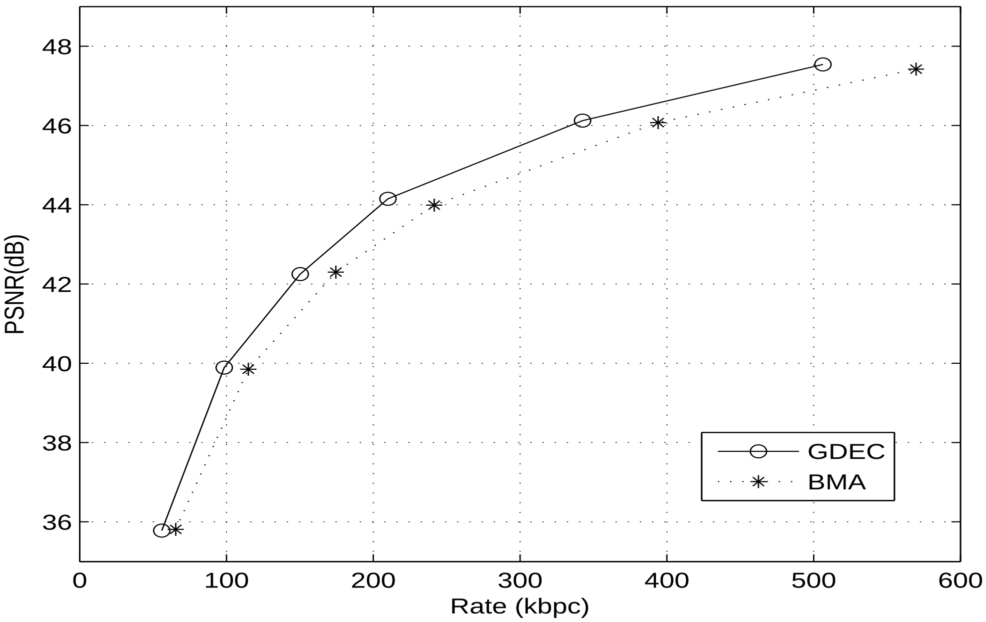


Fig. 6.7 Compression performance comparison of *GDEC* and *BMA* applied to encoding cubic-panorama image dataset *Corridor*.

Regarding to the image dataset coding performance in term of PSNR versus the bitstream rate, significant improvements over the traditional *BMA* are achieved with the *GDEC* approach applied to encoding the testing cubic-panorama image datasets. More accurate predicting images are obtained using the *GDEC* approach compared with using the traditional *BMA* especially on the cube faces with contraction or expansion displacement field distributions. On the other hand, with the same levels of reconstructed basis image quality, the bitstream rates of the *GDEC* approach are lower than these of the *BMA*. Table 6.3 and Table 6.4 show the experimental results of the bitrate reductions brought about with the *GDEC* approach compared with the *BMA*. Over a wide rang of bitrates from approximately 100 kbpc to 500 kbpc, about 17% of

Displacement Compensation Approach	PSNR **Y** (dB)	PSNR **U** (dB)	PSNR **V** (dB)	Bitrate (kbpc)	Bitrate Reduct.
BMA	35.70	42.05	40.74	139.92	—
GDEC	35.78	42.13	40.90	113.57	18.83%
BMA	38.56	43.21	42.08	233.59	—
GDEC	38.63	43.30	42.14	191.45	18.04%
BMA	40.75	44.23	43.26	348.43	—
GDEC	40.81	44.32	43.29	290.80	16.54%
BMA	42.17	45.00	44.21	455.23	—
GDEC	42.18	45.05	44.27	381.16	16.27%
BMA	43.70	46.07	45.39	605.07	—
GDEC	43.81	46.14	45.52	513.58	15.12%

Table 6.3 Bitrate reduction of the coding scheme with *GDEC* applied to encoding experimental cubic-panorama image dataset *Lab*

bitrate reduction on average is obtained for encoding the testing cubic-panorama image dataset *Lab*, and about 13% of bitrate reduction on average for encoding the testing cubic-panorama image dataset *Corridor*.

The compression performances of the block-based hybrid coding schemes with *GDEC* and *BMA* with respect to compression ratios are shown in Fig. 6.8 applied for encoding cubic-panorama image dataset *Lab* and Fig. 6.8 applied for encoding cubic-panorama image dataset *Corridor*. As a result of fewer parameters required for displacement compensation and reduced prediction residues, the developed *GDEC* approach is more efficient for compressing the testing cubic-panorama image datasets. Experimental results demonstrate the superior performance of the developed *GDEC* approach over the traditional *BMA* applied to cubic-panorama image dataset compression.

Displacement Compensation Approach	PSNR **Y** (dB)	PSNR **U** (dB)	PSNR **V** (dB)	Bitrate (kbpc)	Bitrate Reduct.
BMA	39.85	43.60	42.94	114.76	—
GDEC	39.89	43.67	43.01	98.37	14.28%
BMA	43.99	46.68	46.23	241.42	—
GDEC	44.26	46.98	46.56	210.09	12.98%
BMA	45.03	47.55	47.19	305.72	—
GDEC	45.10	47.63	47.29	265.15	13.27%
BMA	46.07	48.80	48.50	393.95	—
GDEC	46.12	48.92	48.57	342.70	13.01%
BMA	47.42	50.16	49.98	569.89	—
GDEC	47.54	50.29	50.04	506.35	11.15%

Table 6.4 Bitrate reduction of the coding scheme with *GDEC* applied to encoding experimental cubic-panorama image dataset *Corridor*

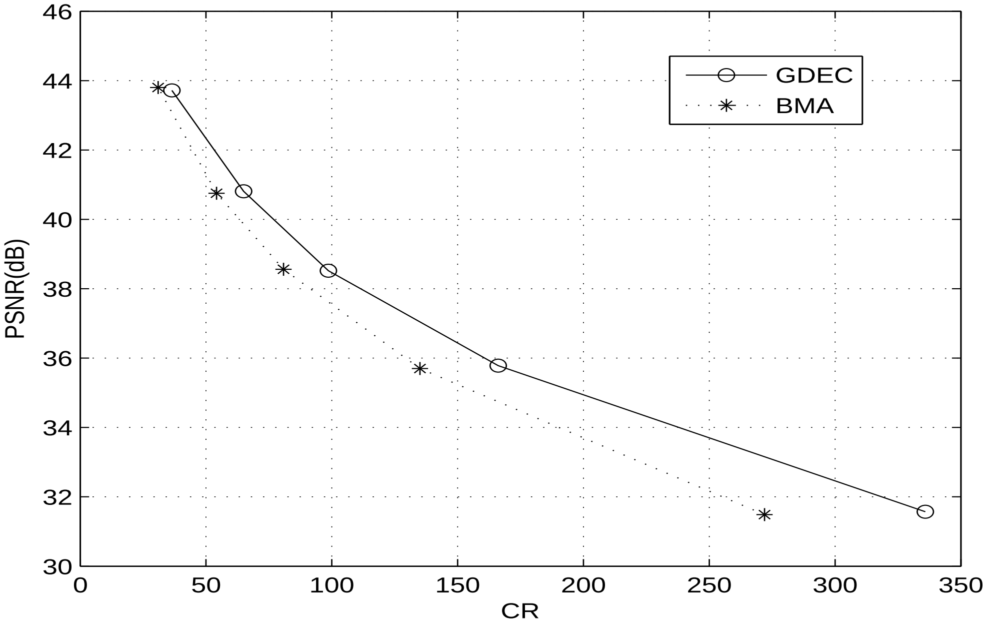


Fig. 6.8 PSNR versus compression ratio of the coding schemes with *GDEC* and *BMA* for encoding cubic-panorama image dataset *Lab*.

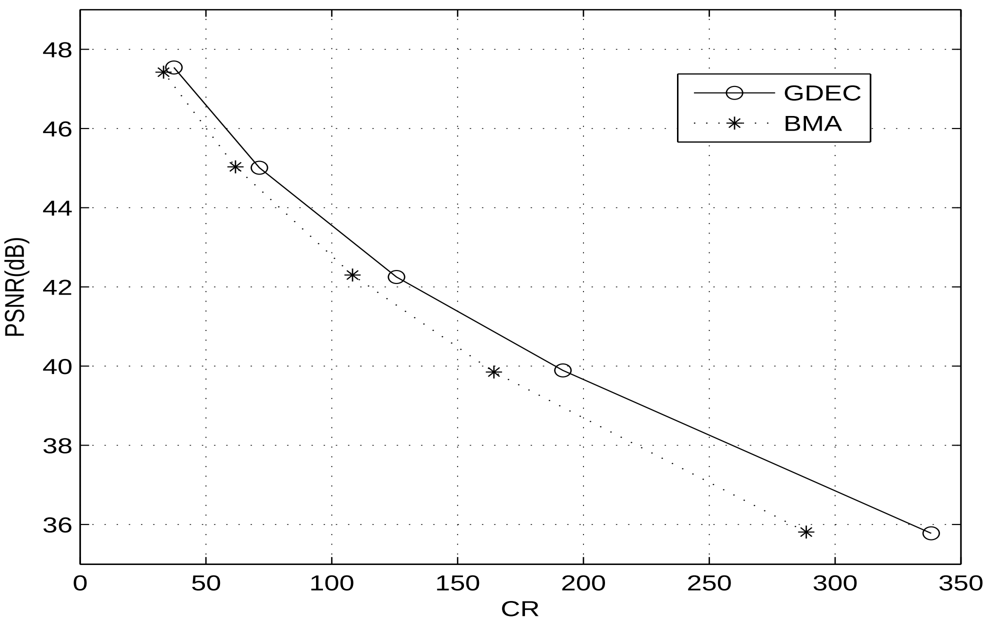


Fig. 6.9 PSNR versus compression ratio of the coding schemes with *GDEC* and *BMA* for encoding cubic-panorama image dataset *Corridor*.

6.6 Summary

Based on a perspective projection model corresponding to the six-parameter camera motions, a new displacement-compensated image prediction algorithm is presented, taking into consideration all kinds of camera motions involving camera translations with parameter components perpendicular and parallel to the camera optical axis. A novel global displacement estimation and compensation approach is developed for cubic-panorama image dataset compression. By replacing the coordinates of correspondence pairs into the transformation formulae, the essential matrix is determined by using a least-squares estimation algorithm. The rotational and translational parameters of the global displacement model are derived from the essential matrix by singular value decomposition (SVD). A unique block-based scaled depth estimation technique is proposed for obtaining the scaled depth map of the predicted image. The displacement-compensated predicting image is generated by using the global rotational and translational model parameters combined with the scaled block-depth map. Compared with the traditional block matching algorithm (BMA) for displacement estimation and compensation, more accurate predicting images are obtained with the proposed displacement-compensated image prediction algorithm. As a result of fewer parameters required for displacement compensation and reduced prediction residues, the developed global displacement estimation and compensation approach is more efficient when applied for cubic-panorama image dataset compression. Experimental results of the testing cubic-panorama image datasets show the effectiveness of the developed global displacement estimation and compensation approach, and also demonstrate its superior performance over the traditional *BMA* applied for cubic-panorama image dataset compression.

Chapter 7

Conclusions

Image dataset compression is one of the most important and fundamental issues in developing an image-based virtual environment navigation system. It is significantly different from the generic image and video sequence compression; coding techniques and schemes specially designed for image dataset compression need to be developed. A specific form of image dataset — the cubic-panorama image dataset, is becoming more popular and is expected to be widely applied with its superior features over other forms of image datasets. This thesis addresses the key problems in the design of a scheme for cubic-panorama image dataset compression to adapt the coding methods to the characteristics of cubic-panorama image datasets, satisfy the special requirements for cubic-panorama image-based rendering and efficiently represent cubic-panorama image datasets in the compressed streaming output.

7.1 Contributions

This thesis work faces some challenges presented in the coding scheme design for compressing image datasets, particularly the cubic-panorama image datasets. It would be, to my knowledge, the first publicly available work on cubic-panorama image dataset compression. In summary, the following contributions are made in this thesis.

- The common characteristics of image datasets used for image-based virtual environment navigation are systematically summarized, and the requirements for image dataset compression, the key issues and strategies for compressing image datasets are generalized. All these provide a valuable reference to researchers who are interested in working on image dataset compression. Although the research on image and video sequence compression has been conducted for quite a few years, image dataset compression is a relatively new research area.

- A spatially consistent representation of cubic-panorama image datasets is proposed with a block padding algorithm for image dataset compression. In conventional compression schemes, cubic-panorama image datasets would be treated as planar image sequences with a rectangular support. It is inevitable to sacrifice some coding efficiency because of the inconsistency on the boundaries of connected side images. With the proposed spatially consistent representation of cubic-panorama image datasets, the search for displacement vectors in reference images can be naturally extended across side image boundaries into neighbor side images. This eliminates the image boundary constraints on displacement vector search. A block padding algorithm for constructing the reference blocks is designed for displacement estimation and compensation. Optimized matching reference blocks can be obtained to reduce the prediction errors and improve the compression efficiency.

- A new context-based arithmetic entropy coding approach named independent block with layered data partition *(IBLDP)* is proposed for image dataset compression with wavelet-based coding schemes. Unlike the widely-applied *EZW* [73] and *SPIHT* [69] wavelet entropy coding approaches, *IBLDP* does not rely on exploiting inter-subband redundancy to improve coding efficiency and consequently removes the restriction of inter-subband coding against resolution-scalable coding. Different from most wavelet entropy coding approaches like *SPECK* [32] and *EZBC* [27] which put all decomposition coefficients of a decomposed image in a subband into a coding dependency chain, *IBLDP* partitions the decomposition coefficients in a subband into independently coded coefficient block and makes spatial random access to image segments of basis images possible. With the compact layered bitplane data representation and the simplified coding structure, *IBLDP* features reduced computational and implemental complexity and hence is more suitable for the efficient and interactive image rendering application.

- A rectangular subregion side-image random access mechanism is designed for cubic-panorama image dataset compression with wavelet-based coding schemes. Providing appropriate random access and selective decoding is a unique requirement for compressing image datasets. A specific hierarchical data structure is proposed to accommodate spatial image random access and maintain the coding performance at high levels. The corresponding bit-stream syntax supporting this data structure is formed to provide efficient selective decoding for interactive image rendering. Multi-level index tables are embedded in the bit-stream to easily facilitate the required spatial image random access.

- A scalable lifted wavelet-based coding scheme with cross-image displacement compensation is developed for cubic-panorama image dataset compression. It poses some challenges to develop a coding scheme for compressing image datasets because a more complicated balance among the coding efficiency, the spatial-resolution scalability and the random access and selective decoding flexibility needs to be made. This developed cubic-panorama image dataset compression scheme is built based on the scalable lifted wavelet transform framework enhanced with displacement compensation. It incorporates the new context-based arithmetic entropy coding approach and combines the specially designed random access mechanism. A suitable tradeoff among the compression efficiency, the spatial-resolution scalability and the desired random access flexibility is reached by adapting the coding scheme to cubic-panorama image dataset compression.

- A novel global displacement estimation and compensation approach with scaled block-depth estimation is developed for cubic-panorama image dataset compression. Most global displacement estimation algorithms deal with global image displacements associated with camera zoom, pan, tilt and rotation, avoiding the more challenging issue arising from camera translations that relate displacement compensation to the depth distribution of the scenes. Few papers deal with camera translations but limit the discussions to the scenes able to be approximated by a planar surface, confine the translation to the camera optical axis direction or assume the scene depth distribution determinable. In this thesis, based on a perspective projection model of six-parameter global displacements, a displacement-compensated image prediction algorithm is presented, taking into consideration all kinds of camera motions involving camera translations with parameter components perpendicular and parallel to the camera axis. A unique block-based scaled depth estimation technique is proposed. Superior coding performance over that of the comparable *BMA* is achieved with the proposed approach.

7.2 Future Research Work

Although some compression techniques and schemes have been developed for compressing cubic-panorama image datasets, some problems still remain to be investigated to work out more efficient and effective cubic-panorama image dataset compression schemes for image-based virtual environment navigation.

The developed coding scheme with global displacement estimation and compensation for cubic-panorama image dataset compression demonstrates superior coding performance. However, it is currently applied to the indoor image datasets which are captured in relatively stable conditions. For outdoor image datasets, this coding scheme needs to be improved to adapt to the unsteady movement of the camera in capturing the image sequences and the uneven changes of lighting conditions.

Random access and selective decoding are unique requirements for image dataset compression, and spatial scalable coding is extremely desirable for rendering images in steady quality. It presents some challenges to satisfy all the coding requirements while maintaining the high coding efficiency. Currently, a proper balance among these requirements is made in the proposed schemes. Optimized compression schemes in terms of compression efficiency under the constraint of providing the required spatially scalable coding and random access will be investigated and evaluated to provide more precise and optimized control over meeting all these requirements in the compression schemes.

Currently, in our work, compressing the spatially consistent representation of cubic-panorama image datasets, scalable lifted wavelet-based cubic-panorama image dataset coding and global displacement estimation and compensation for cubic-panorama image dataset compression are implemented separately in different compression schemes based on either block-based hybrid coding or wavelet-based coding. It is very necessary and quite valuable to combine all these methods in a single compression scheme, most likely a wavelet-based scheme, to further benefit from the superior features of these methods in practical applications of cubic-panorama image dataset compression. This presents a challenge to our future work on cubic-panorama image dataset compression.

Besides the above mentioned work, the quality of cubic-panorama image datasets needs to be enhanced for more precise image analysis required by image dataset compression schemes. Special treatment of the down side image is necessary to remove its negative influence on the coding results. All these suggest the potential future work on image dataset compression.

References

[1] M. D. Adams and F. Kossentini, "JasPer: A Software-Based JPEG 2000 Codec Implementation," *Proc. IEEE Int. Conf. Image Processing (ICIP'00),* Vancouver, BC, Canada, Sept. 2000, vol. 2, pp. 53-56.

[2] E. H. Adelson and J. Bergen, "The Plenoptic Function and the Elements of Early Vision," *Computer Models of Visual Processing,* pp. 3-20, MIT Press, Cambridge, MA, USA, 1991.

[3] D. G. Aliaga and I. Carlbom, "Plenoptic Stitching: A Scalable Method for Reconstructing 3D Interactive Walkthroughs," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'01),* Los Angeles, CA, USA, Aug. 2001, pp. 443-450.

[4] M. Antonini, M. Barlaud, P. Mathieu and I. Daubechies, "Image Coding Using Wavelet Transform," *IEEE Trans. Image Processing*, vol. 1, no. 2, pp. 205-220, April, 1992.

[5] M. Beermann and E. Dubois, "Acquisition Processing Chain for Dynamic Panoramic Image Sequences," *Proc. IEEE Int. Conf. Image Processing (ICIP'07)*, Sept. 2007, vol. V, pp. 217-220.

[6] D. Bradley, "Navire Cube Viewer," Navire Internal Technical Report, NAV-TR-2005-01, Aug. 2005.

[7] D. Bradley, A. Brunton, M. Fiala and G. Roth, "Image-based Navigation in Real Environments Using Panoramas," *Proc. IEEE Int. Workshop on Haptic Audio Visual Environments and their Applications (HAVE'05)*, Ottawa, ON, Canada, Oct. 2005, pp 57-59.

[8] J.-X. Chai, X. Tong, S.-C. Chan and H.-Y. Shum, "Plenoptic Sampling," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'00)*, New Orleans, LA, USA, 2000, pp. 307-318.

[9] S.-C. Chan and H.-Y. Shum, "A Spectral Analysis for Light Field Rendering," *Proc. IEEE Int. Conf. Image Processing (ICIP'00)*, Vancouver, BC, Canada, Sept. 2000, vol. 2, pp. 25-28.

[10] S. E. Chen, "QuickTime VR - An Image-Based Approach to Virtual Environment Navigation," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'95),* Aug. 1995, pp. 29-38.

[11] W.-C. Chen, J.-Y. Bouguet, M. H. Chu and R. Grzeszczuk, "Light Field Mapping: Efficient Representation and Hardware Rendering of Surface Light Fields," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'02),* San Antonio, Texas, USA, July 2002, pp. 447-456.

[12] P. Chen and J. W. Woods, "Bidirectional MC-EZBC with Lifting Implementation," *IEEE Trans. Circuits and Syst. for Video Technol.,* vol. 14, no. 10, pp. 1183-1194, Oct. 2004.

[13] Z. Chen, Z. Nie, X. Gu, L. Zhu and C. Wang, "Fast Global Motion Estimation Based on Iteration Least-Square Estimation with Sustained Symmetrical Structure," *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS'06),* Island of Kos, Greece, May 2006, pp. 4695-4698.

[14] S. -J. Choi and J. W. Woods, "Motion-Compensated 3-D Subband Coding of Video," *IEEE Trans. Image Processing,* vol. 8, no. 2, pp. 155-167, Feb. 1999.

[15] E. Dubois, "Frequency-domain Methods for Demosaicking of Bayer-sampled Color Images," *IEEE Signal Process. Lett.*, vol. 12, no. 12, pp. 847-850, Dec. 2005.

[16] E. Dubois, "Video Sampling and Interpolation," *Handbook of Image and Video Processing*, pp. 911-922, Elsevier Academic Press, Amsterdam, Netherlands, 2005.

[17] E. Dubois, "Filter Design for Adaptive Frequency-domain Bayer Demosaicking," *Proc. IEEE Int. Conf. Image Processing (ICIP '06)*, Atlanta, GA, USA, Oct. 2006, pp 2705-2708.

[18] F. Dufaux and J. Konrad, "Efficient, Robust, and Fast Global Motion Estimation for Video Coding," *IEEE Trans. Image Processing,* vol. 9, no. 3, pp. 497-501, Mar. 2000.

[19] A. N. Evans, Y. Guo and D. M. Monro, "Limited Motion Estimation Scheme for Multimedia Video Compression," *Proc. IEEE Int. Conf. Electronics, Circuits and Systems (ICECS'99),* Pafos, Cyprus, Sept. 1999, vol. 1, pp. 453-456.

[20] R. Ewerth, M. Schwalb, P. Tessmann and B. Freisleben, "Estimation of Arbitrary Camera Motion in MPEG Videos," *Proc. IEEE Int. Conf. Pattern Recognition (ICPR'04)*, Surrey, UK, Aug. 2004, vol. 1, pp. 512-515.

[21] M. Fiala and G. Roth, “Automatic Alignment and Graph Map Building of Panoramas,” *Proc. IEEE Int. Workshop on Haptic Audio Visual Environments and their Applications (HAVE’05)*, Ottawa, ON, Canada, Oct. 2005, pp 104-109.

[22] M. Flierl and B. Girod, “Video Coding with Motion-Compensated Lifted Wavelet Transforms,” *Signal Processing: Image Communication,* vol. 19, no. 7, pp. 561-575, Aug. 2004.

[23] B. Girod, C.-L. Chang, P. Ramanathan and X. Zhu, “Light Field Compression Using Disparity-Compensated Lifting,” *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP’03),* Hong Kong, China, Apr. 2003, pp. IV760-763.

[24] S. Gortler, R. Grzeszcczuk, R. Szeliski and M. F. Cohen, “The Lumigraph,” *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH’96),* New Orleans, LA, USA, Aug. 1996, pp. 43-52.

[25] N. Greene, “Environment Mapping and Other Applications of World Projections,” *IEEE Computer Graphics and Applications,* vol. 6, no. 11, pp. 21-29, Nov. 1986.

[26] Y. He, B. Feng, S. Yang and Y. Zhong, “Fast Global Motion Estimation for Global Motion Compensation Coding,” *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS’01)*, Sydney, Australia, May 2001, vol. 2, pp. 233-236.

[27] S.-T. Hsiang and J. W. Woods, “Invertible Three-Dimensional Analysis/Synthesis System for Video Coding with Half-Pixel-Accurate Motion Compensation,” *Proc. SPIE Visual Communications and Imaging Processing Conf.,* San Jose, CA, USA, Jan. 1999, vol. 3653, pp. 537-546.

[28] S.-T. Hsiang and J. W. Woods, “Embedded Video Coding Using Invertible Motion Compensated 3-D Subband/Wavelet Filter Bank,” Signal Processing: Image Communication, vol. 16, no. 8, pp. 705-724, May 2001.

[29] T. S. Huang and A. N. Netravali, “Motion and Structure from Feature Correspondences: A Review,” *Proc. IEEE*, vol. 82, no. 2, pp. 252-268, Feb. 1994.

[30] X. Huang and E. Dubois, “3D Reconstruction Based on a Hybrid Disparity Estimation Algorithm,” *IEEE Int. Conf. on Image Processing,* pp. 1025-1028, Oct. 2006.

[31] S. Ikeda, T. Sato and N. Yokoya, “High-Resolution Panoramic Movie Generation from Video Streams Acquired by an Omnidirectional Multi-Camera System,” *Proc. IEEE Int. Conf. Multisensor Fusion and Integration for Intelligent Systems (MFI’03)*, Aug. 2003, pp. 155-160.

[32] A. Islam and W. A. Pearlman, "An Embedded and Efficient Low-Complexity Hierarchical image Coder," *Proc. SPIE Conf. Visual Communications and Image Processing*, San Jose, CA, USA, Jan. 1999, vol. 3653, pp. 294-305.

[33] ISO/IEC JTC1/SC29/WG1, "Information Technology — Digital Compression and Coding of Continuous-tone Still Images: Requirements and Guidelines," ISO/IEC 10918-1 and ITU-T Rec. T.81 (JPEG), 1992.

[34] ISO/IEC JTC1/SC29/WG1, "Information Technology — JPEG 2000 Image Coding System, Part 1: Core Coding System," ISO/IEC 15444-1 and ITU-T Rec. T.800 (JPEG2000), Aug. 2000.

[35] ISO/IEC JTC1/SC29/WG11, "Information Technology — Coding of Moving Pictures and Associated Audio for Digital Storage Media at Up To About 1.5 Mbit/s, Part 2: Video," ISO/IEC 11172-2 (MPEG-1), Nov. 1992.

[36] ISO/IEC JTC1/SC29/WG11, "Information Technology — Generic Coding of Moving Pictures and Associated Audio Information, Part 2: Video," ISO/IEC 13818-2 and ITU-T Rec. H.262 (MPEG-2), Nov. 1994.

[37] ISO/IEC JTC1/SC29/WG11, "Information Technology — Coding of Audio-Visual Objects, Part 2: Visual," ISO/IEC 14496-2 (MPEG-4), Jan. 2000.

[38] ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, "H.264 /MPEG-4 AVC Reference Software," ISO/IEC and ITU JVT-X072, July 2007.

[39] ITU-T, "Video Codec for Audiovisual Services at p x 64 kbit/s," ITU-T Rec. H.261, Mar. 1993.

[40] ITU-T, "Video Coding for Low Bit Rate Communication," ITU-T Rec. H.263, Version 2, Jan. 1998.

[41] ITU-T and ISO/IEC, "Advanced Video Coding for Generic Audiovisual Services," ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC (H.264/AVC), 2003.

[42] K. Jiang and E. Dubois, "Motion-Oriented Coding Scheme for Compression of Concentric Mosaic Scene Representations," *Proc. IEEE Int. Workshop on Haptic Audio Visual Environments and their Applications*, Ottawa, ON, Canada, Oct. 2004, pp. 37-42.

[43] K. Jiang and E. Dubois, "Lifted Wavelet-Based Image Dataset Compression with Column Random Access for Image-Based Virtual Environment Navigation," *Proc. IEEE Int. Workshop on Haptic Audio Visual Environments and their Applications*, Ottawa, ON, Canada, Oct. 2005, pp.77-82.

[44] K. Jiang and E. Dubois, "Compression of Cubic-Panorama Datasets with Spatially Consistent Representation," *Proc. IEEE Int. Workshop on Haptic Audio Visual Environments and their Applications*, Ottawa, Ontario, Canada, November 2006, pp. 111-116.

[45] K. Jiang and E. Dubois, "Motion-Compensated Frame Prediction with Global Motion Estimation for Image Sequence Compression," *Proc. IEEE Canadian Conf. Electrical and Computer Engineering (CCECE'07)*, Vancouver, BC, Canada, Apr. 2007, pp. 1468-1471.

[46] S. B. Kang, "A Survey of Image-based Rendering Techniques," *Proc. SPIE Electronic Imaging Symp.,* San Jose, CA, USA, Jan. 1999, vol. 3641, pp. 2-16.

[47] F. Kangni and R. Laganiere, "Epipolar Geometry for the Rectification of Cubic Panoramas," *Proc. of the 3rd Canadian Conference on Computer and Robot Vision (CRV'06)*, June 2006, pp. 70-77.

[48] E.-T. Kim, "Robust 3-D Camera Motion Parameter Estimation with Applications to Video Coding," *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'06)*, Toulouse, France, May 2006 vol. II, pp. 261-264.

[49] J.C. Lagarias, J. A. Reeds, M. H. Wright and P. E. Wright, "Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions," *SIAM Journal of Optimization,* vol. 9 no. 1, pp. 112-147, Jan. 1998.

[50] D. T. Lee, "JPEG 2000: Retrospective and New Developments," *Proceedings of the IEEE,* vol. 93, no. 1, pp. 32-41, Jan. 2005.

[51] D. Le Gall and A. Tabatabai, "Sub-Band Coding of Digital Images Using Symmetric Short Kernel Filters and Arithmetic Coding Techniques," *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'88)*, New York, NY, USA, April 1988, pp. 761-764.

[52] B. J. Lei, C. Chang, E. A. Hendriks, "An Efficient Image-Based Telepresence System for Videoconferencing," *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 14, no. 3, pp. 335-347, Mar. 2004.

[53] M. Levoy and P. Hanrahan, "Light Field Rendering," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'96),* New Orleans, LA, USA, Aug. 1996, pp. 31-42.

[54] Z. C. Lin and H.-Y. Shum, "On the Number of Samples Needed in Light Field Rendering with Constant-Depth Assumption," *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR'00)*, Hilton Head Island, SC, USA, June 2000, vol. 1, pp. 588-595.

[55] L. Luo, Y. Wu, J. Li and Y.-Q. Zhang, "Compression of Concentric Mosaic Scenery with Alignment and 3D Wavelet Transform," *Proc. SPIE Image and Video Communications and Processing Symp.,* San Jose, CA, USA, Jan. 2000, vol. 3974, pp. 89-100.

[56] L. Luo, J. Li, S. Li, Z. Zhuang and Y.-Q. Zhang, "Motion Compensated Lifting Wavelet and Its Application in Video Coding," *Proc. IEEE Int. Conf. Multimedia and Expo (ICME'01)*, Aug. 2001, pp. 365-368.

[57] A. Luthra, G. J. Sullivan and T. Wiegand, "Introduction to the Special Issue on the H.264/AVC Video Coding Standard," *IEEE Trans. Circuits and Syst. for Video Technol.,* vol. 13, no. 7, pp. 557-559, July 2003.

[58] M. Magnor and B. Girod, "Hierarchical Coding of Light Fields with Disparity Maps," *Proc. IEEE Int. Conf. Image Processing (ICIP'99),* Kobe, Japan, Oct. 1999, vol. 3, pp. 334-338.

[59] M. Magnor and B. Girod, "Data Compression for Light-Field Rendering," *IEEE Trans. Circuits and Syst. for Video Technol.,* vol. 10, no. 3, pp. 338-343, Apr. 2000.

[60] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler and L. McMillan, "Image-Based Visual Hulls," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'00),* New Orleans, LA, USA, 2000, pp. 369-374.

[61] L. McMillan and G. Bishop, "Plenoptic Modeling: An Image-Based Rendering System," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'95),* Los Angeles, CA, USA, Aug. 1995, pp. 39-46.

[62] NAVIRE: http://www.site.uottawa.ca/research/viva/projects/ibr.

[63] J.-R. Ohm, "Three-Dimensional Subband Coding with Motion Compensation," *IEEE Trans. Image Processing,* vol. 3, no. 5, pp. 559-571, Sept. 1994.

[64] J.-R. Ohm, "Advances in Scalable Video Coding," *Proceedings of the IEEE,* vol. 93, no. 1, pp. 42-56, Jan. 2005.

[65] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer and T. Wedi, "Video Coding with H.264/AVC: Tools, Performance, and Complexity," *IEEE Circuits and Syst. Magazine,* vol. 4, no. 1, pp. 7-28, First Quarter 2004.

[66] G. Paul, C. Tillier, B. Pesquet-Popescu and H. Heijmans, "Motion Compensation and Scalability in Lifting-Based Video Coding," *Signal Processing: Image Communication,* vol. 19, no. 7, pp. 577-600, Aug. 2004.

[67] B. Pesquet-Popescu and V. Bottreau, “Three-Dimensional Lifting Schemes for Motion Compensated Video Compression,” *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP’01),* Salt Lake City, UT, USA, 2001, pp. 1793-1796.

[68] A. Puro, X. Chen and A. Luthra, “Video Coding Using the H.264/MPEG-4 AVC Compression Standard,” *Signal Processing: Image Communication,* vol. 19, pp. 793-849, 2004.

[69] A. Said and W. A. Pearlman, “A New Fast and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees,” *IEEE Trans. Circuits and Syst. for Video Technol.,* vol. 6, pp. 243-250, June 1996.

[70] A. Secker and D. Taubman, “Motion-Compensated Highly Scalable Video Compression Using an Adaptive 3D Wavelet Transform Based on Lifting,” *Proc. IEEE Int. Conf. Image Processing (ICIP’01),* Thessaloniki, Greece, 2001, vol. 2, pp. 1029-1032.

[71] S. M. Seitz and C. R. Dyer, “View Morphing,” *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH’96),* New Orleans, LA, USA, Aug. 1996, pp. 21-30.

[72] J. Shade, S. Gortler, L.-W. He and R. Szeliske, “Layered Depth Images,” *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH’98),* Orlando, FL, USA, July 1998, pp. 231-242.

[73] J. M. Shapiro, “Embedded Image Coding Using Zerotrees of Wavelet Coefficients,” *IEEE Trans. Signal Processing,* vol. 41, no. 12, pp. 3445-3462, Dec. 1993.

[74] H.-Y. Shum and L.-W. He, “Rendering with Concentric Mosaics,” *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH’99),* Los Angeles, CA, USA, Aug. 1999, pp. 299-306.

[75] H.-Y. Shum, K.-T. Ng and S.-C. Chan, “Virtual Reality Using the Concentric Mosaic: Construction, Rendering and Data Compression,” *Proc. IEEE Int. Conf. Image Processing (ICIP’00),* Sept. 2000, vol. 3, pp. 644-647.

[76] H.-Y. Shum, S. B. Kang and S.-C. Chan, “Survey of Imaged-Based Representations and Compression Techniques,” *IEEE Trans. Circuits and Syst. for Video Technol.,* vol. 13, no. 11, pp. 1020-1037, Nov. 2003.

[77] Y. Su, M.-T. Sun and V. Hsu, “Global Motion Estimation from Coarsely Sampled Motion Vector Field and the Applications,” *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 15, no. 2, pp. 232-242, Feb. 2005.

[78] G. Sullivan, "Efficient Scalar Quantization of Exponential and Laplacian Variables," IEEE Trans. Information Theory, vol. 42, no. 5, pp. 1365-1374, Sept. 1996.

[79] X. Sun and E. Dubois, "View Morphing nnd Interpolation through Triangulation," *Proc. SPIE Electronic Imaging Symp., Conf. on Image and Video Communications and Processing,* vol. 5685, pp. 513-521, Jan. 2005.

[80] X. Sun and E. Dubois, "A Matching-Based View Interpolation Scheme," *Proc. IEEE Int. Conf. Acoustics Speech Signal Processing*, pp. II 877-880, March 2005.

[81] W. Sweldens, "The Lifting Scheme: A Construction of Second Generation Wavelets," *SIAM J. Math Anal.*, vol. 29, no. 2, pp. 511-546, Mar. 1998.

[82] R. Szeliski and H. Shum, "Creating Full View Panoramic Image Mosaics and Environment Maps," *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH'97),* 1997, pp. 251-258.

[83] S.-C. Tai, Y.-Y. Chen and S.-F. Sheu, "Deblocking filter for Low Bit Rate MPEG-4 Video," *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 15, no. 6, pp. 733-741, June 2005.

[84] D. Taubman, "High Performance Scalable Image Compression with EBCOT," *IEEE Trans. Image Processing,* vol. 9, no. 7, pp. 1158-1170, July 2000.

[85] X. Tong and R. M. Gray , "Coding of Multi-View Images for Immersive Viewing," *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP'00),* Istanbul, Turkey, June 2000, vol. 4, pp. 1879-1882.

[86] X. Tong and R. M. Gray, "Interactive Rendering from Compressed Light Fields," *IEEE Trans. Circuits and Syst. for Video Technol.,* vol. 13, no. 11, pp. 1080-1091, Nov. 2003.

[87] P. N. Topiwala, "Time-Frequency Analysis, Wavelets and Filter Banks," *Wavelet Image and Video Compression*, pp. 33-59, Kluwer Academic Publishers, Boston, Massachusetts, USA, 1998.

[88] M. Uyttendaele, A. Criminisi, S. B. Kang, S. Winder, R. Szeliski and R. Hartley, "Image-Based Interactive Exploration of Real-World Environments," *IEEE Trans. Computer Graphics and Applications*, vol. 24, no. 3, pp. 52-63, May 2004.

[89] E. Vincent and R. Laganire, "Junction Matching and Fundamental Matrix Recovery in Widely Separated Views," *Proc. British Machine Vision Conf.*, pp. 77-86, Sept. 2004.

[90] C. Warrington, G. Roth and E. Dubois, “Markerless Augmented Reality for Cubic Panorama Sequences,” *Proc. IEEE/ACM Int. Symposium on Mixed and Augmented Reality (ISMAR’06)*, Oct. 2006, pp 255-256.

[91] J. Weng, T. S. Huang and N. Ahuja, “Motion and Structure from Two Perspective Views: Algorithms, Error Analysis, and Error Estimation,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 5, pp. 451-476, May 1989.

[92] J. J. Weng and T. S. Huang, “3-D Motion Analysis from Image Sequences Using Point Correspondences,” *Handbook of Pattern Recognition and Computer Vision*, pp. 395-441, World Scientific Publishing Co., 1993.

[93] T. Wiegand, G. J. Sullivan, G. Bjontegaard and A. Luthra, “Overview of the H.264/AVC Video Coding Standard,” *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 13, no. 7, pp. 560-576, July 2003.

[94] D. N. Wood, D. I. Azuma, K. Aldinger, B. Curless, T. Duchamp, D. H. Salesin and W. Stuetzle, “Surface Light Fields for 3D Photography,” *Proc. ACM Annu. Conf. Computer Graphics and Interactive Techniques (SIGGRAPH’00),* New Orleans, LA, USA, 2000, pp. 287-296.

[95] Y. Wu, L. Luo, J. Li and Y.-Q. Zhang, “Rendering of 3D-Wavelet Compressed Concentric Mosaic Scenery with Progressive Inverse Wavelet Synthesis (PIWS),” *Proc. SPIE Visual Communications and Image Processing Symp.,* Perth, Australia, June 2000, vol. 4067, pp. 31-43.

[96] Y. Wu, C. Zhang and J. Li, “Smart Rebinning for the Compression of Concentric Mosaic,” *IEEE Trans. on Multimedia,* vol. 4, no. 3, pp. 332-342, Sept. 2002.

[97] A. Zakhor and F. Lari, “Edge-Based 3-D Camera Motion Estimation with Application to Video Coding,” *IEEE Trans. Image Processing*, vol. 2, no. 4, pp. 481-498, Oct. 1993.

[98] C. Zhang and J. Li, “Compression of Lumigraph with Multiple Reference Frame (MRF) Prediction and Just-in-Time Rendering,” *IEEE Data Compression Conf.,* Snowbird, UT, USA, Mar. 2000, pp. 254-263.

[99] C. Zhang and J. Li, “Compression and Rendering of Concentric Mosaics with Reference Block Codec (RBC),” *Proc. SPIE Visual Communications and Image Processing Symp.,* Perth, Australia, June 2000, vol. 4067, pp. 43-54.

[100] C. Zhang and J. Li, “Interactive Browsing of 3D Environment over the Internet,” *Proc. SPIE Visual Communications and Image Processing Symp.,* San Jose, CA, USA, Jan. 2001, vol. 4310, pp. 509-520.

[101] C. Zhang and T. Chen, “Spectral Analysis for Sampling Image-Based Rendering Data,” *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 13, no. 11, pp. 1038-1050, Nov. 2003.

[102] C. Zhang and T. Chen, “A Survey on Image-Based Rendering – Representation, Sampling and Compression,” *Signal Processing: Image Communication,* vol. 19, pp. 1-28, Jan. 2004.

[103] Z. Zhang, F. Wang, G. Zhu, L. Xie and J. Gao, “A New Global Motion Estimation Algorithm,” *Proc. SPIE Image Analysis Techniques*, Nov. 2005, vol. 6044, pp. 604422 1-8.

[104] L. Zhang, D. Wang and A. Vincent, “Adaptive Reconstruction of Intermediate Views from Stereoscopic Images,” *IEEE Trans. Circuits and Syst. for Video Technol.*, vol. 16, pp. 102-113, Jan. 2006.

[105] J. Ziv and A. Lempel, “A Universal Algorithm for Sequential Data Compression,” *IEEE Trans. Information Theory*, vol. 23, no. 3, pp. 337-343, May 1977.