Distributed Source Coding Architectures for Multi-view Images

(多視点画像の分散型符号化器設計法)

Mehrdad Panahpour Tehrani[†], Michael Droese[†], Toshiaki Fujii (member)[†],
and Masayuki Tanimoto (member)[†]

Abstract This paper discusses new methods for coding of multi-view images. Three distributed solutions are proposed based on a parent node and child node framework. A parent node encodes the whole image whereas a child node only partially. The proposed scheme allows independent encoding of each view. Experimental results show good performance of the proposed architectures at low bit-rates.

Key words: Distributed source coding, multi-view images, parent node, child node

1. Introduction

Multi-view images of a scene can be used for several applications ranging from free viewpoint television (FTV)¹⁾²⁾ to surveillance. Due to the enormous size of multi-view images, coding is one of the challenges to build such applications. Multi-view images are usually highly correlated in spatial domain and therefore spatial redundancy can be removed by encoding the information "differentially" with respect to an appropriate "reference". The disadvantage of this method is an extra overhead in communication between those nodes. In a scenario with limited processing and communication abilities this method is not preferable. Work by Slepian and Wolf³⁾ shows that even if the sources are encoded independently, they can be fully reconstructed under certain conditions. Distributed source coding of multi-view images is preferable if there is a major constraint on individual camera node performance (i.e., energy, which is consumed by sensing and communication operations). Kusuma et al⁴⁾ presented a distributed source coding method for sensor networks. Zhu et al⁵⁾ proposed a distributed compression method for a large number of cameras using Wyner-Ziv⁶⁾ coding, where the parent

nodes (PNs) and children nodes (CNs) are interspersed amongst one another. The geometry is reconstructed from the PNs and used to render an estimate for each CN's view.

This letter proposes a coding method without inter-node communication for multi-view images; similar to the distributed source coding method proposed in⁷⁾ for intelligent transportation system (ITS). To perform the decoding task disparity estimation is employed to compensate the scene geometry. In addition, three source coding architectures using the distributed fashion with and without PNs are proposed and evaluated considering: communication amount, symmetry in communication and decoding quality, and robustness of joint decoder.

The remainder of the paper is organized as follows: Section 2 describes the three proposed coding architectures. In Section 3 the coding scheme using CNs is explained in detail. Section 4 shows the experimental results in comparison with individually coded images using JPEG compression and finally Section 5 with the conclusion of this paper.

2. Coding Architectures

The cameras in the multi-view system are grouped into correlated clusters. Each cluster of nodes is coded independently. The cluster size corresponds to the maximum allowable disparity, respectively the maximum distance of a camera pair. In **Fig. 1** different coding architectures without inter-node communication can be

Reported in SPIE/VCIP on Jan. 20, 2004

Received March 31, 2004; Revised July 9, 2004; Accepted August 9, 2004

[†]Department of Information Electronics, Graduate School of Engineering, Nagoya University

(Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan

TEL: +81-52-789-3163)

Fig. 1 Distributed source coding architectures: (a) PPP, (b) PCP, (c) CPC, (d) CCC.

seen. Fig.1(a) shows the individual coding of multi-view images. Therefore, all nodes are PNs and the coding architecture is named PPP (PN,...,PN,...,PN). The following are the three proposed architectures. Fig.1(b) demonstrates coding with two PNs and therefore is called PCP (PN,...,CN,...,PN). Fig.1(c) illustrates a distributed source coding including just one PN, which is located in the middle of the cluster. This architecture is called CPC (CN,...,PN,...,CN). Another architecture avoids any PNs in a cluster, which makes the decoding of CN independent from PN as shown in Fig.1(d). This coding architecture is called CCC (CN,...,CN,...,CN). The arrows in Fig.1 show the node to be used to decode a CN at the joint decoder. In CPC, the PN in the middle provides side information to decode each CN in the cluster. CNs in PCP are decoded using virtual-PNs (vPN), which are generated by the two PNs at the outer border of a cluster. In CCC, each CN is decoded by using its neighbor.

3. Coding Method

Before describing the encoding/decoding algorithms it is essential to note the variables used throughout the following sections. " n " describes the number of pixels in a block to be encoded. " D " stands for the pre-determined maximum allowable gray level difference bound that is imposed on the multi-view image coding. In other words, a pixel in a specific location

Fig. 2 Distributed source coder in general.

in an image, is at most $D/2$ gray level values more or less than the corresponded location pixel in another image. The " D " corresponds to intra-pixel correlation at encoder side, however it represents the inter-block correlation at decoder side. "Maximum disparity" stands for the number of pixels required to find all correspondences in a stereo setup. It also defines the size of a cluster. **Fig. 2** shows a block diagram of the proposed distributed source coder in general (i.e., with and without PN). Although CNs and PNs are set apart, in practice they are arranged one after another as shown in Fig. 1. Due to no inter-node communication amongst cameras, the views are encoded independently at each node. The encoded data is transmitted to the joint decoder. At the joint decoder the side information to perform the joint decoding is provided by the scene geometry, which is obtained by an area-based matching method⁽⁸⁾⁽⁹⁾.

3.1 Encoding

In the encoding algorithm, the PN images are encoded using JPEG. As shown in **Fig. 3** the encoding of a CN works as follows: The captured image is sub-sampled according to the block size " n " and forms the so called "syndrome image" (i.e., not to be done in PCP architecture). The remaining pixels (i.e., all pixels in a block in PCP architecture) build an image, called "coset image". In other words, no syndrome pixels is present in the CNs of PCP architecture. However, the CNs in CPC and CCC architectures have syndrome pixels. Due to high correlation among pixels of a block, the coset image is further encoded by applying a module-

Fig. 3 Encoding method of CNs.

operation based on " D " to the pixel values. Therefore, the intra-pixel correlation has been taken into account at the encoder side. JPEG encoding of the syndrome and coset images follows. Note that the JPEG coding is applied to coset image, which has $\log_2(D)$ bit per pixel.

3.2 Decoding

After receiving the encoded data at the joint decoder, the PN images are reconstructed by JPEG decompression. The CN image reconstruction is also done by JPEG decompression of the syndrome and coset images, respectively and combining them into one image. Due to the module-operation applied to the coset image, decoding is necessary. This is done by using side information from the estimated scene geometry. This scene geometry is obtained by using a block-based matching method [6], [7], as mentioned above. The geometry compensation of the PCP architecture is using a disparity estimation with regularization on the neighbourhood disparities, which introduces smoothing of the disparities. In CPC and CCC architectures, the correspondence problem is solved by using block matching of the syndrome and coset pixels of a PN and a CN. The reconstructed CN image is postprocessed with a median filter to reduce salt and pepper noise. Therefore all CNs are reconstructed under similar conditions.

For the proposed architectures the decoding is explained in the following. CPC: For each pixel of a block in a CN, the corresponding pixels at the PN are found by using the PN pixels and the syndrome and coset pixels of CN. PCP: A vPN image is generated in the location of each CN as shown in Fig.1(b). The CN image is decoded by using the vPN's pixels at the same location as the coset pixels. CCC: To decode the image the correspondence is obtained by using the syndrome and coset pixels of two CNs.

The values of the coset image are decoded by applying an "inverse" module-operation, which is not unique. Therefore, that solution is chosen which minimizes the distance to the corresponding pixel of the other image (i.e., syndrome pixels of CN, PN or vPN). Therefore, the inter-block correlation has been taken into account at decoder side.

4. Experiment

In this experiment, the data set consists of 25 views with 640x480 pixels per view. The camera interval is 5mm and the distance from camera to object measures about 30cm. For this experiment a maximum disparity of 120 pixels was allowed. In our case this is equal

Fig. 4 Experimental results " D " =16, " n " =4, Max. disparity=120 pixels: (a) Rate-PSNR curve of three coding architecture and individual JPEG compression of each node (b) PCP coding of a CN at 0.1bpp, reconstruction of a CN with 30mm base line located in between two PNs at 31.26dB.

to a stereo setup with a camera baseline of 30mm. All experiments are carried out on luminance component only. Reconstruction quality of a CN is measured in term of peak-signal-to-noise-ratio (PSNR) and bit-rates are expressed as bit per pixel (bpp).

Fig. 4 shows PSNR curves vs. bit-rates (a) as well as the sample reconstructed image of a CN in PCP with 30mm distance (b). The individually encoded images using JPEG (PPP) are compared with the decoded image of the three distributed source coding architectures (CPC, CCC, and PCP). The comparison is done with the nearest (5mm distance between the camera nodes) and the maximum allowable case (30mm). The maximum gray level difference " D " has been chosen 16, and the block size " n " 4 pixels. Due to geometry compensation at the joint decoder, the proposed distributed source coder performs in all three architectures significantly better than PPP in the low bit-rates. The performance gain is up to 4dB in PSNR. At higher bit-rates, the PPP JPEG coder tends to be more efficient, whereas the reconstruction quality of the decoding is limited by miss-detection of correspondences in geometry estimation. Furthermore, the distance between the CN and the node, which provides the side information, influences the performance gain of the coding method. For all disparities PCP performs better than other ar-

Table 1

Performance comparison of 25 nodes coding using the three coding architectures and individual JPEG compression of each node.

Coding method	Communication amount	Communication symmetry	Decoding symmetry	Robustness
PPP	High	Symmetrical	Symmetrical	High
CPC	Medium	Asymmetrical	Asymmetrical	Medium
PCP	Low	Asymmetrical	Asymmetrical	Medium
CCC	Medium	Symmetrical	Symmetrical	High

chitectures due to the fact that no syndrome is present. In the other architectures (CPC and CCC), each CN is decoded by using a PN or a CN directly, that gives a good quality for large disparities. Due to less side information the CCC architecture has almost the same performance in all disparities.

In the case of PPP and CCC, the communication amount between camera nodes and joint decoder and the decoded image quality at the joint decoder are symmetrical. In other cases, neither the communication, nor the decoded image qualities are symmetrical. In case of architectures with PN the reconstruction relies on error free transmission. If any data from the PN is lost, the decoding will certainly fail. However, the decoding can be performed using a neighboring PN with the result in reducing the image quality. If no PN is involved, the decoding can be survived by employing any neighboring CN in the cluster. **Table 1** summarizes the performance of the 4 cases in Fig.1, considering communication amount, symmetry in communication and decoding quality, and their robustness in case of transmission errors.

5. Conclusion

Three distributed architectures for multi-view image coding are proposed. The CCC architecture has the most optimal performance among all other architectures, however its decoding performance can be improved. The PCP and CPC architectures are interesting according to their high compression rate at low bit-rates and small disparities. Furthermore, the simplicity of this distributed source coding method allows an easy implementation, even if processing and communication capabilities are limited.

In the future research, a perfect model of the scene should be used to compare the architectures independent from miss-detection of correspondence search, and also the optimization of maximum gray level difference and bit-rate should be considered to obtain the maximum decoding PSNR result. In addition, the proposed architecture can be extended to two-dimensional and dynamic multi-view images.

[References]

- 1) P. Na Bangchang, T. Fujii, and M. Tanimoto, "Experimental system of free viewpoint television", Proceedings of SPIE on Stereoscopic Displays and Virtual Reality System, 5006-66, pp. 554-563 (Jan. 2003)
- 2) T. Fujii, and M. Tanimoto, "Experimental system of free viewpoint television", Proceedings of SPIE on Stereoscopic Displays and Virtual Reality System, 5006-66, pp. 554-563 (Jan. 2004)
- 3) D. Slepian and J.K. Wolf, "Noiseless Coding of Correlated Information Sources," IEEE Transaction on Information Theory, IT-19, pp. 471-480 (July 1973)
- 4) J. Kusuma, L. Doherty, K. Ramchandran, 'Distributed Compression for Sensor Networks', IEEE Signal Processing Society Conference ICIP (2001)
- 5) X. Zhu, A. Aaron and B. Girod, 'Distributed Compression For Large Camera Arrays', Proc. IEEE Workshop on Statistical Signal Processing, SSP-2003, St Louis, Missouri, (Sept. 2003)
- 6) A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," IEEE Transactions on Information Theory, IT-22, 1, pp. 1710 (Jan. 1976)
- 7) M. P. Tehrani, T. Fujii, M. Tanimoto, "A Distributed Source coding for ITS", FIT 2002, pp.389-390 (Sept. 2002)
- 8) A. Nakanishi, T. Fujii, T. Kimoto, and M. Tanimoto, "Ray-space data interpolation for view image generation", IDW'01, pp. 1385-1388 (2001)
- 9) A. Nakanishi, T. Fujii, T. Kimoto, and M. Tanimoto, "Ray-space data interpolation by adaptive filtering using locus of corresponding points on epipolar plane image", The journal of the institute of Image information and Television Engineers (ITE), 56, 8, pp. 1321-1327 (Aug. 2002)

Mehrdad Panahpour Tehrani received the B.E. and M.E. degrees in Electrical Engineering from Tehran Polytechnics University, 1997 and Tarbiat Modarres University-Tehran, Iran 2000, respectively. Currently, he is a candidate for a doctor of engineering at Nagoya University, Japan. His research interests include 3D image processing, and communications.

Michael Droese received the Dipl.-Ing. degree in Computer Engineering from the Technical University of Berlin, Germany, in 2003. Currently he is a Visiting Researcher at Nagoya University, Japan. His research interests include 3D image and video processing as well as vision and computer graphics.

Toshiaki Fujii received the B.E., M.E., and Dr. Eng. degrees in electrical engineering from the University of Tokyo, Japan, in 1990, 1992, and 1995, respectively. He is currently an Associate Professor in the Graduate School of Engineering, Nagoya University, Japan. His research interests include 3D image processing and 3D visual communications.

Masayuki Tanimoto received the B.E., M.E. and Dr.E. degrees in electronic engineering from the University of Tokyo, Japan, in 1970, 1972 and 1976, respectively. He has been a Professor of the Department of Information Electronics, Nagoya University, Japan. His current research interests include image coding, multidimensional signal processing, video processing, 3D and HD images, and multimedia systems.