Visual Comput (2007) 23: 891–896
DOI 10.1007/s00371-007-0145-4

ORIGINAL ARTICLE

Feng Dai
Yanfei Shen
Yongdong Zhang
Shouxun Lin

Selection of the most efficient tile size in tile-based cylinder panoramic video coding and transmission

Published online: 9 June 2007

F. Dai (✉) · Y. Shen · Y. Zhang · S. Lin
Institute of Computing Technology
Chinese Academy of Sciences
Beijing 100080, P.R. China
fdai@ict.ac.cn

F. Dai
Graduate University of Chinese Academy of Sciences, Beijing 100080, P.R. China

Abstract Panoramic videos are a 360 degree representation of a certain scene. The users can navigate interactively through the scene and change their view angles. Panoramic videos are often high-resolution and consume a significant amount of bandwidth for transmission. To resolve the problem, tile-based panoramic video coding and transmission is applied in some systems. With tile-based panoramic video coding and transmission, only the tiles involved with the perspective view are transmitted and decoded. Different tile sizes will bring different transmission bit rates for same video quality. In this paper, a two path coding method with H.264/AVC for cylinder panoramic video based on a hyperbolic model is proposed. With this method, the most efficient tile size can be selected and users can build the same quality perspective view with the smallest transmission bit rate.

Keywords Cylinder panoramic video · Most efficient tile size

1 Introduction

Today, with the increasing processing power of CPUs and graphics adapters, more complex data can be processed by computers. Additionally, the Internet has helped us to connect and communicate with others anywhere in the world. The increasing bandwidth of the Internet provides the possibility of transmitting larger amounts of bits than ever before. These technological advances make it is possible for people to demand better user experience in interactive applications such as virtual walkthroughs and computer games.

In computer graphics literature, the methods for scene representation are often classified into two categories. The first one represents the scene by classical 3-D computer graphics, called geometry-based modeling. The other represents the scene by real images or videos, called image-based modeling. Geometry-based modeling provides a higher degree of interactivity in general than is found in image-based modeling, but it suffers from the expensive cost in both modeling and rendering. The image-based approach can avoid these drawbacks and brings the users a natural feeling in the real scene.

Panoramic configuration is one example of image-based representations. Providing interactions such as zoom and rotation gives the effect of "looking around" to the users. We can obtain panoramic videos using a multi-camera system or single camera system.

Panoramic videos can be thought of as a 360 degree representation of a certain scene. The field of view of a panoramic video can be 360 degrees in the vertical and horizontal direction while that of conventional videos are usually 60 ∼ 70 degrees. The users can navigate interactively through the scene and change their view angle with a special panoramic video player. Cylinder projection is one of the most popular projections in applications. Cylinder panoramic videos are often 360 degrees in the horizontal direction and limited in the vertical view direction. The users can also change their view direction in the horizontal direction.

Because of the large field of view and large dataset, the resolution of panoramic videos is usually very large. For example, the resolution of panoramic video acquired by

the Telemmersion System is up to 2400 × 1200 [4]. Compared with conventional videos, a large amount of transmission bandwidth is needed.

Tile-based panoramic video coding is a popular method in panoramic video coding. It divides the panorama into tiles. Each tile is compressed and decompressed individually. Only the tiles involved with the current perspective view need to be transmitted. An appropriate portion of the panorama inside the tiles is used to render the perspective view, which we call corresponding areas.

In tile-based panoramic video coding and transmission, the selection of tile size is a problem. An appropriate tile size could reduce a large amount of data transmission while the same quality perspective view is built. Up to now, researchers have worked on the problem but did not obtain a satisfying result [7]. In this paper, we propose a two path coding method based on a hyperbolic model to find the most efficient tile size for cylinder panoramic video coding, which brings the smallest data transmission to build the perspective view with similar quality.

The rest of paper is organized as follows. Section 2 briefly reviews and analyzes the tile-based data coding and transmission for cylinder panoramic videos. A two path coding method based on a hyperbolic model is presented to select the most efficient tile size in Sect. 3. In Sect. 4, experimental results are reported. We conclude the paper in Sect. 5.

2 Tile-based cylinder panoramic video coding and transmission

As the resolution of a panoramic video is very large, transmitting the entire panoramic video is very often time-consuming. Fortunately, building the perspective view of a given view direction does not need all of the data in the panoramic video frame. In order to avoid transmitting the entire high-resolution image to users, in QuickTime VR system [1] the tile-based coding and transmission for panoramic videos is initially proposed. It allows rotation and zooming within a photorealistic 360 degree panoramic view. Many researchers also adopt the tile-based coding and transmission in their systems [2, 3, 5, 6]. In [5], the authors divided the panorama with a resolution of 2048 × 768 into six tiles. In [6], the authors thought the ideal tile size was 64 × 480 or 352 × 480 to a panorama of 3520 × 480. Actually, in the mentioned research, no work was done on the impact of tile size to the efficiency of compression. Yamazawa et al. evaluated the different results of divided panoramic video using H.264/AVC with different tile sizes in [7], and the result do not show different bit rates among numbers of tile. We thought why they got the result is that the all the chosen tile sizes are too large (the smallest is 1024 × 1024). In our experiments the results show that if two tile sizes are both large enough, the difference between them is too small to be observed.

Tile-based panoramic video coding and transmission divides the high-resolution panorama into tiles. Each tile is compressed and decompressed individually. The server transmits the tiles involved with the perspective view. The corresponding areas of the panorama inside the tiles are used to render the perspective view.

Because of the limited field of view in the vertical, the vertical view angle is usually fixed. Only the horizontal view angle is changed. Therefore we only need to divide the panorama into tiles in the horizontal direction and not in the vertical direction as shown in Fig. 1.

3 Selection of the most efficient tile size

3.1 Relation between tile size and compression efficiency

Firstly we discuss the relation between tile size and compression efficiency for panoramic video. The division of the panorama will bring more tile boundaries and more macroblocks located on the boundaries. Because each tile must be encoded and transmitted individually, they cannot refer to each other. During intraprediction, the valid intraprediction modes for macroblocks located on the boundaries are less than those inside the tiles. During interprediction, the motion vectors for macroblocks located on the boundaries have more possibilities to point to the outside of the slice than the ones inside the tile. So for the same panoramic video, the smaller tile size, the larger the number of macroblocks on the tile boundary, and the lower the compression efficiency.

3.2 Relation between tile size and involved macroblocks

When the field of novel perspective view is given, the different tile sizes will bring the different number of involved

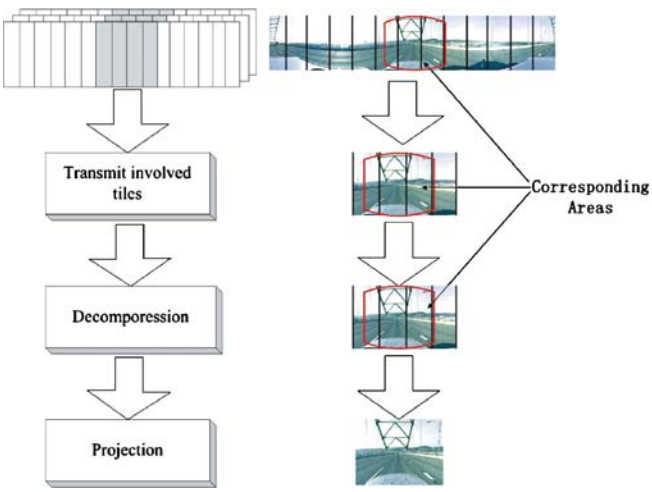


Fig. 1. Process of building a perspective view from a cylinder panoramic video

tiles and involved macroblocks. Smaller tile size will bring more involved macroblocks and larger tile size will bring less involved macroblocks. As shown in Fig. 2, when the panorama is divided into 13 tiles, four tiles are involved and need to be transmitted to users. If the tile number increases to 26, seven tiles in the panorama are involved and transmitted. From Fig. 2 we can see that the macroblocks in the shadow need to transmit when the tiles number is 13 and do not need to transmit to users when the tiles number is 26.

When the field of the perspective view is known, the shape and size of corresponding areas is determined, but the different position of corresponding areas brings the different number of involved tiles. In Fig. 3, the corresponding areas involve three tiles. If the view direction moves towards the left a little, four tiles will be involved.

Furthermore, when the tile size is a and the width of corresponding areas is M in macroblock units, the width of involved tiles may be $\lceil \frac{M}{a} \rceil$ or $\lceil \frac{M}{a} \rceil + 1$ depending on the position of the corresponding areas. We can obtain the average width of involved tiles using the following Eq. 1, in which $\lceil\rceil$ is the ceil function:

$$\begin{aligned} W_a &= \left(\left(\left\lceil \frac{M}{a} \right\rceil + 1 \right) \cdot a \right) \cdot \frac{a'}{a} + \left(\left\lceil \frac{M}{a} \right\rceil \cdot a \right) \cdot \frac{a''}{a} \\ &= \left(\left\lceil \frac{M}{a} \right\rceil + 1 \right) \cdot a \cdot \frac{M - \left(\left(\left\lceil \frac{M}{a} \right\rceil - 1 \right) \cdot a \right)}{a} \\ &\quad + \left\lceil \frac{M}{a} \right\rceil \cdot a \cdot \frac{a - \left(M - \left(\left\lceil \frac{M}{a} \right\rceil - 1 \right) \cdot a \right) \right)}{a} \\ &= M + a \,. \end{aligned} \tag{1}$$

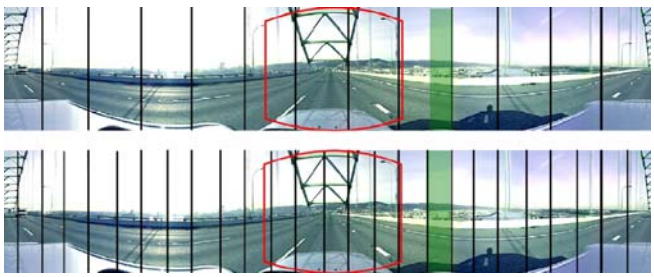


Fig. 2. Involved tiles with different tile sizes

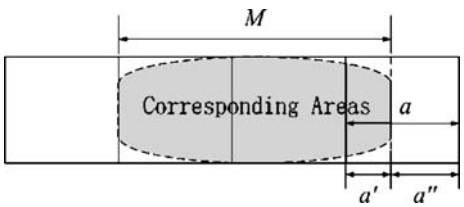


Fig. 3. Illustration of M, a' , a'' and a

3.3 Transmission bit rate and the most efficient tile size

From the discussion above, we know that smaller tile size brings fewer involved macroblocks and reduces the number of transmission macroblocks. But on the other hand, smaller tile size will brings less compression efficiency. More bits have to be consumed in each macroblock to get the same quality.

We define transmission bit rate as the sum of bit rates of all the involved tiles, which must be transmitted to the users. Suppose the bit rate is uniform everywhere in the frame, when the field of view is known, the average transmission bit rate could be calculated as:

$$r_a = \frac{N_a}{N} \cdot R_a = \frac{W_a}{W} \cdot R_a \tag{2}$$

where r_a represents the average transmission bit rate, i.e., sum of bit rate of involved tiles, N_a represents the number of involved macroblocks when tile size is a, N represents the number of all macroblocks in the panoramic video frame, R_a represents the bit rate of panoramic video when tile size is a, W represents the width of panoramic frame in macroblock units, and W_a represents the width of involved tiles in macroblock units. Our aim is to find the most efficient tile size a_{me}, which minimizes r_a.

In the process of coding, quantization parameter (QP) is an important parameter. Large quantization parameter brings high compression and low visual quality; small quantization parameter brings low compression and high visual quality. We can select the parameter according to the application and bandwidth. The most direct and accurate way to find the most efficient tile size for a given quality is full coding, which encodes the panoramic video sequence with an appropriate quantization parameter employing different tile sizes. Then, calculate and compare all the values of the average transmission bit rate obtained by Eq. 2. The tile size which minimizes the average transmission bit rate is the most efficient tile size. Table 1 shows the result of average transmission bit rate with different tile sizes encoding a sequence *bridge* whose resolution is 1920×352. In the experiment, QP is 28 and the field of perspective view is set to 60 degrees, i.e., $M = 20$.

From Table 1 we can clearly see that when the tile size is two, the average transmission bit rate is lowest. R_a is the bit rate obtained in the experiment by encoding the sequence with H.264/AVC. More results show that the average transmission bit rate always falls before the most efficient tile size then increases with the increment of tile size.

3.4 Two path coding method based on hyperbolic model

Full coding is the most direct way to find the most efficient tile size, but it is too complicated to use in applications. We propose a two path coding method based on a hyperbolic model to get the most efficient tile size.

Table 1. Average transmission bit rate for different tile sizes

Tile size a	R_a(kps)	r_a(kps)
1	1128.51	197.49
2	1063.04	194.89
3	1041.77	199.67
4	1030.95	206.19
5	1023.48	213.28
6	1021.11	221.24
...	...	...
40	1003.98	501.99

First we define λ_a as the ratio of R_a to R_0:

$$\lambda_a = \frac{R_a}{R_0} \tag{3}$$

where R_a is the bit rate when the tile size is a, and R_0 is the bit rate when the panorama is not divided. The term λ_a is a float value which is always more than one. From Eq. 2 and Eq. 3, we obtain Eq. 4:

$$r_a = \frac{W_a}{W} \cdot R_a = \frac{W_a}{W} \cdot \lambda_a \cdot R_0 \,. \tag{4}$$

For different tile sizes, W and R_0 are same. So we define a variable P_a:

$$P_a = W_a \cdot \lambda_a = (M + a) \cdot \lambda_a \,. \tag{5}$$

Now, we need only to find the tile size which minimizes P_a. Further, we define ω_a as the increment ratio of R_a to R_0:

$$\omega_a = \frac{R_a - R_0}{R_0} = \lambda_a - 1. \tag{6}$$

Experiments are conducted to obtain the relation between ω_a and the tile size a. We encoded a sequence bridge employing different tile sizes. For each tile size, four quantization parameters are used. For every tile size and quantization parameter, the corresponding increment ratio ω_a is calculated using Eq. 6. Table 2 is the result of increment ratio ω_a. Figure 4a shows the relation between ω_a and tile number and Fig. 4b shows the relation between ω_a and tile size a.

From Fig. 4 we can clearly see that ω_a is directly proportional to the tile number. Because tile size is an inverse measure of tile number, the value of ω_a varies inversely with the tile size as a shown in Fig. 4b. We repeat the experiment with different frame sizes and sequences. The same results are obtained that indicate ω_a is directly proportional to the tile number and is an inverse measure of tile size.

Because ω_a is an inverse measure of tile size, we build a hyperbolic model to express the relation between ω_a and tile size a as shown in Eq. 7. C is a constant for different a. When a equals W, ω_a is zero:

Table 2. Average transmission bit rate for different tile sizes

Tile number	Tile size a	ω_a $QP=28$	$QP=32$	$QP=36$	$QP=40$
1	120	0.00%	0.00%	0.00%	0.00%
3	40	0.25%	0.41%	0.52%	0.83%
4	30	0.39%	0.50%	0.87%	1.47%
6	20	0.62%	0.95%	1.42%	2.15%
8	15	0.78%	1.11%	1.87%	3.03%
10	12	0.80%	1.17%	2.06%	3.20%
12	10	1.21%	1.83%	2.89%	4.67%
15	8	1.50%	2.18%	3.61%	5.80%
20	6	1.86%	2.81%	4.70%	7.51%
30	4	2.92%	4.59%	7.38%	11.73%
40	3	3.94%	6.16%	9.96%	15.65%
60	2	5.96%	9.33%	14.94%	23.37%
120	1	11.96%	18.47%	29.38%	45.56%

$$\omega_a = \frac{C}{a} - \frac{C}{W}. \tag{7}$$

From Eq. 7, we obtain Eq. 8:

$$C = \frac{\omega_a \cdot a \cdot W}{W - a}. \tag{8}$$

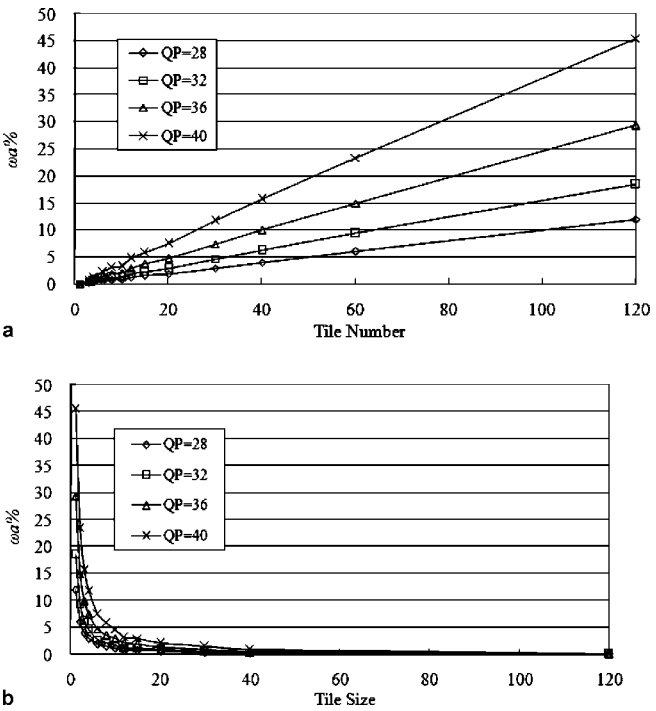


Fig. 4. **a** Relation between ω_a% and tile number. **b** Relation between ω_a% and tile size a

Based on Eq. 5, Eq. 6 and Eq. 7, we obtain Eq. 9:

$$P_a = (M+a)\cdot(1+\omega_a) = (M+a)\cdot\left(1+\frac{C}{a}-\frac{C}{W}\right). \quad (9)$$

The derivative of P_a equals zero, when P_a is minimum. Because a is an integer, we get the most efficient tile size a_{me} as Eq. 10:

$$P'_a = 0 \Rightarrow a_{\text{me}} = \left\lfloor \sqrt{\frac{CM}{1-\frac{C}{W}}} \right\rfloor. \quad (10)$$

Based on the discussion above, we propose an efficient two path panoramic video coding to encode the panoramic video with the most efficient tile size:

1. Encode the panoramic video without dividing and obtain R_0.
2. Encode the panoramic video with a tile size a_{pre}; the bit rate $R_{a_{\text{pre}}}$ is obtained.
3. Calculate ω_a using Eq. 6.
4. Calculate C using Eq. 8.
5. Using Eq. 10, calculate the most efficient tile size a_{me}.

Tile size a_{pre} is a preliminary tile size for encoding and in the experiment it was set to three. Actually, it could be set to other values and not affect the selection of the most efficient tile size.

4 Experimental results

Four cylinder panoramic video sequences as shown in Fig. 5 were tested in the experiment. All the sequences are from Immersive Media [4]. We set the field of perspective view as 60 degrees. For 1920×352 the width of corresponding areas is 320 pixels, that is, $M = 20$ MBs (macroblocks). For 2880×512, $M = 30$ MBs. The term a_{pre} is three in the experiment.

From Table 3 we can see that only two results with asterisks are different from the results obtained by full coding. The right ratio of results is 94%. This is a satisfying result. We can also notice that all of the most efficient tile sizes are small and no result is more than six. For a given sequence, the most efficient tile size increases with M and QP. More experimental results show that even when the M and QP are large enough, the most efficient tile size is always less than 10. In the experiment, we changed a_{pre} from 1 to 10 and the same results were obtained.

5 Conclusion

We proposed a two path coding method based on a hyperbolic model to select the most efficient tile size in cylinder panoramic video coding. This proposed method can obtain the most efficient tile size with much less complexity than full coding. The result obtained by two path coding method is almost the same as with full coding. When the most efficient tile size is used in the tile-based panoramic video coding, a large amount of transmission bit rate is saved. This is very useful to the application of the cylinder panoramic videos in the Internet.

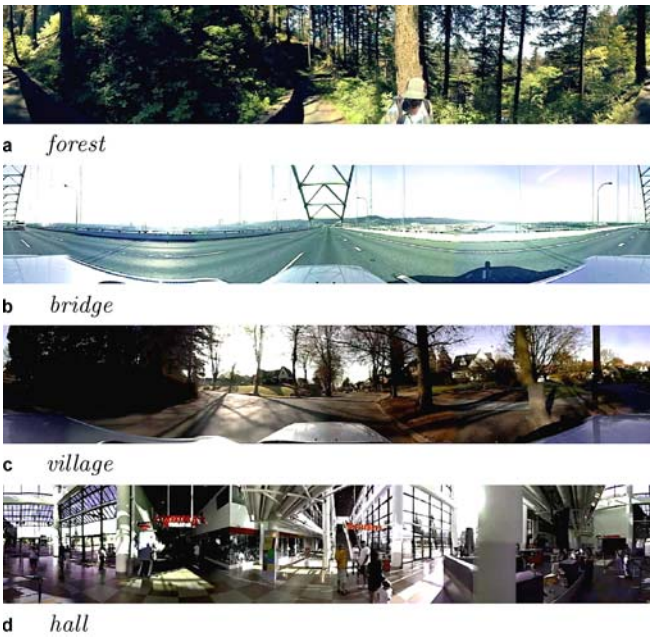


Fig. 5. Four cylinder panoramic video sequences

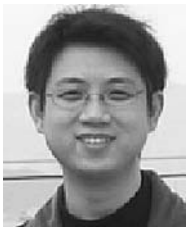
Table 3. Results of most efficient tile size selection

Sequence	QP	Most efficient tile size a_{me}			
		1920×352		2880×528	
		Proposed	Full	Proposed	Full
Forest	28	2	2	2	2
	32	2	2	3	3
	36	2	2	3	3
	40	3	3	4	4
Bridge	28	2	2	2	2
	32	2	2	3	3
	36	2	2	3	3
	40	3	3	4	4
Village	28	2	2	3	3
	32	2	2	3	3
	36	3	3	4*	3*
	40	3	3	4	4
Hall	28	3	3	3	3
	32	3	3	4	4
	36	4	4	5*	6*
	36	4	4	6	6

Acknowledgement This work is supported by Beijing Science and Technology Planning Program of China (D0106008040291).

References

1. Chen, S.E.: QuickTime VR: an image-based approach to virtual environment navigation. In: SIGGRAPH 95 Conference Proceedings, pp. 29–38 (1995)
2. Grunheit, C., Smolic, A., Wiegand, T.: Efficient representation and interactive streaming of high-resolution panoramic views. In: International Conference on Image Processing, vol. 3, pp. 24–28 (2002)
3. Heymann, S., Smolic, A., Miller, K., Guo, Y., Rurainski, J., Eisert, P., Wiegand, T.: Representation, coding, and interactive rendering of high-resolution panoramic images and video using MPEG-4. The 2nd Panoramic Photogrammetry Workshop, Berlin, Germany, February (2005)
4. Immersive Media: Telemmersion System. http://www.immersivemedia.com. Cited (2006)
5. Ng, K.T., Chan, S.C., Shum, H.Y.: Data compression and transmission aspects of panoramic videos. IEEE Trans. Circuits Syst. Video Technol. **15**(1), 82–95 (2005)
6. Pintaric, T., Neumann, U., Rizzo A.: Immersive panoramic video. Proceedings of the 8th ACM International Conference on Multimedia, pp. 493–494, October (2000)
7. Yamazawa, K., Kitaura, R., Habe, H., Kimata, H., Nomura, T.: Evaluation result of divided omni-directional video using AVC (EE1). In: ISO/IEC JTC 1/SC 29/WG 11 MPEG2003/M10416, Waikaloa, HI, December (2003)




FENG DAI born in 1979, is currently working toward his Ph.D. degree in the Multimedia Computing Group, Virtual Reality Lab, Institute of Computing Technology, Chinese Academy of Sciences. He received the B.S. degree in Computer Science and Technology from Tsinghua University, Beijing, China, in 2002. His research interests include video coding and transmission, video processing, and image-based rendering.

YANFEI SHEN born in 1976, received his B.S. and M.S. degrees in Computer Science from the Key Laboratory of Multimedia and Network Communication, Wuhan University, China, in 1999 and in 2002. He is currently working as an Associate Researcher in the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His research interests include video coding technology, video processing, digital TV, and associated VLSI architecture.

YONGDONG ZHANG born in 1973, received his Ph.D. degree in Electronic Engineering from Tianjing University, Tianjing, China, in 2002. He is an Associate Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His research interests are in the field of video coding and transcoding, video analysis and retrieval, and universal media access.

SHOUXUN LIN born in 1948, received his Ph.D. degree from Beijing University of Technology, Beijing, China, in 1998. Since 1995, he has been Associate Professor and Professor (in 2000) with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His research interests include multimedia processing and comparison, video coding, video analysis, multimedia indexing, statistical machine translation, and evaluation of computer human interaction.