Journal of Southeast University (English Edition) Vol. 26, No. 3, pp. 384 – 388 Sept. 2010 ISSN 1003—7985

Perceptual video coding method based on JND and AR model

Wang Chong Zhao Li Zou Cairong

(School of Information Science and Engineering, Southeast University, Nanjing 210096, China)

Abstract: In order to achieve better perceptual coding quality while using fewer bits, a novel perceptual video coding method based on the just-noticeable-distortion (JND) model and the auto-regressive (AR) model is explored. First, a new texture segmentation method exploiting the JND profile is devised to detect and classify texture regions in video scenes. In this step, a spatial-temporal JND model is proposed and the JND energy of every micro-block unit is computed and compared with the threshold. Secondly, in order to effectively remove temporal redundancies while preserving high visual quality, an AR model is applied to synthesize the texture regions. All the parameters of the AR model are obtained by the least-squares method and each pixel in the texture region is generated as a linear combination of pixels taken from the closest forward and backward reference frames. Finally, the proposed method is compared with the H. 264/AVC video coding system to demonstrate the performance. Various sequences with different types of texture regions are used in the experiment and the results show that the proposed method can reduce the bit-rate by 15% to 58% while maintaining good perceptual quality.

Key words: perceptual video coding; texture synthesis; just-noticeable-distortion; AR model

In the last two decades, image and video compression techniques have been developed greatly. The state-of-the-art JPEG2000[1] and MPEG-4 AVC/H. 264[2] greatly outperform their predecessors in terms of coding efficiency. All these methods attempt to remove spatial-temporal statistical redundancy of the visual signal for compression. Unfortunately, a common problem is that the statistical redundancy among pixels is considered as the only adversary of compression, with perceptual redundancy being totally ignored. That is to say, although the rate distortion for the previous video compression standards is broadly adopted, it does not completely reflect the particularity of human vision. Essentially, compression schemes and vision systems face a similar problem, that is, how to represent visual objects in efficient and effective ways.

To further improve the coding efficiency, much pioneering work[3-5] has been done by exploiting the human visual system (HVS) limitations to develop an encoding system targeted at the perception criterion rather than statistical fidelity. In these works, some texture regions in video scenes, such as flowers, grass, water and sand, which are not sensitive or important to HVS, are first segmented and then reconstructed by synthesizing.

Received 2010-04-28.

Biography: Wang Chong (1982—), male, doctor, lecturer, wangchong219@163.com.

Foundation item: The National Natural Science Foundation of China (No. 60472058, 60975017).

Citation: Wang Chong, Zhao Li, Zou Cairong. Perceptual video coding method based on JND and AR model[J]. Journal of Southeast University (English Edition), 2010, 26(3): 384 – 388.

A coding scheme integrating the texture analyzer and the synthesizer to the traditional hybrid coding framework was introduced in Ref. [3], in which a multi-resolution quad tree of scalable color descriptor is applied to segment an image by a series of splitting and merging processes. After the segmentation, the affine model was used to synthesize the segmented textured regions, rather than encoding them by traditional methods. This approach achieves good performance for rigid objects. However, it seems to have poor performance for non-rigid objects. To overcome such a limitation, a texture synthesis algorithm based on graph cut was proposed in Ref. [4], which exhibits good performance for non-rigid objects. Nevertheless, this approach has no superiority for rigid objects compared with Ref. [3]. Both these two algorithms utilize statistical color information to perform image segmentation. In Ref. [5], a simple edge detector was first utilized to detect structural blocks, and then the textural blocks were selected from the remaining blocks. The detected texture blocks were then passed by the corresponding blocks, and were pointed by their motion vectors in the reference frame.

All these methods[3-5] just take statistical characteristics of color or edge information into account and ignore the HVS when performing image segmentation. It is very desirable to segment texture regions by exploiting HVS limitations and image contents. As for texture synthesis, both methods in Ref. [3] and Ref. [4] are not robust enough to achieve good performance for various texture regions. The synthesis algorithm in Ref. [5] just calculated one block from its reference frame.

To tackle the problems mentioned above and fully employ the perceptual redundancy, this paper proposes a perceptual image segmentation algorithm and then synthesizes the segmented regions by an auto-regressive (AR) model, which is very robust for various kinds of texture regions.

In this paper, we classify input sequences into texture frames and non-texture frames. I-frames and P-frames are defined as non-texture frames, which are encoded by traditional methods. B-frames are defined as texture frames, in which the proposed image segmentation and texture synthesis are performed. The perceptual image segmentation mainly employs the just-noticeable-distortion (JND) to detect and segment the texture regions. It is pointed out that human eyes cannot sense any changes below the JND threshold [6]. In this paper, the JND is used to guide the segmentation process of texture regions.

In the proposed AR texture synthesis algorithm, each pixel is synthesized by a linear combination of pixels in the temporal neighborhoods in its adjacent frames. AR has shown remarkable progress and has been adopted for different applications, such as image compression and video processing. In Ref. [7], Wu et al. presented a piecewise 2D AR for predictive image coding, and in Ref. [8], Tugnait proposed a texture synthesis using the asymmetric 2-D non-

causal AR model.

1 Texture Region Segmentation

1.1 Spatial JND model

The perceptual redundancy in the spatial domain is mainly based on the sensitivity of the HVS due to luminance contrast and the spatial masking effect[9-10]. Various computational JND models have been developed. In Refs. [9-10], the JND models were built in the spatial (pixel) domain. To incorporate the CSF into the JND model, some other models were proposed in sub-band, DCT, and wavelet domains [11-14]. In this paper, we use the spatial JND model proposed in Ref. [9]. Chou and Li[9] found that the spatial JND threshold can be modeled as a function of luminance contrast and spatial masking,

$$\mathrm{SJND}(x, y) = \max\{f_1(bg(x, y), mg(x, y)), f_2(bg(x, y))\} \tag{1}$$

where $f_1(bg(x, y), mg(x, y))$ and $f_2(bg(x, y))$ are functions to estimate the spatial masking and luminance contrast, respectively. The quantity $f_1(bg(x, y), mg(x, y))$ is defined as

$$f_1(bg(x, y), mg(x, y)) = mg(x, y) \times \alpha(bg(x, y)) + \beta(bg(x, y)) \tag{2}$$

where $mg(x, y)$ is the maximum weighted average of luminance differences derived by calculating the weighted average of luminance changes around position (x, y) in four directions.

$$mg(x, y) = \max_{k=1,2,3,4}\{|\mathrm{grad}_k(x, y)|\} \tag{3}$$

where

$$\mathrm{grad}_k(x, y) = \frac{1}{16}\sum_{i=1}^{5}\sum_{j=1}^{5} p(x-3+i, y-3+i)G_k(i, j) \tag{4}$$

The operators G_k are defined in Fig. 1. The quantities $\alpha(bg(x, y))$ and $\beta(bg(x, y))$ in Eq. (2) depend on the background luminance and specify the linear relationship between the visibility threshold and the luminance difference (or luminance contrast around the point of coordinate (x, y)); hence, they model the spatial masking[15]. These quantities are expressed as

$$\left.\begin{aligned}\alpha(bg(x, y)) &= bg(x, y) \times 0.0001 + 0.115\\ \beta(bg(x, y)) &= \mu - bg(x, y) \times 0.01\end{aligned}\right\} \tag{5}$$

0	0	0	0	0
1	3	8	3	1
0	0	0	0	0
−1	−3	−8	−3	−1
0	0	0	0	0

(a)

0	0	1	0	0
0	8	3	0	0
1	3	0	−3	−1
0	0	−3	−8	0
0	0	−1	0	0

(b)

0	0	1	0	0
0	0	3	8	0
−1	−3	0	3	1
0	−8	−3	0	0
0	0	−1	0	0

(c)

0	1	0	−1	0
0	3	0	−3	0
0	8	0	−8	0
0	3	0	−3	0
0	1	0	−1	0

(d)

Fig. 1 Definition of G_k. (a) G_1; (b) G_2; (c) G_3; (d) G_4

In Eq. (5), μ is the slope of the function at a higher background luminance level. $bg(x, y)$ is the average background luminance calculated by a weighted low-pass filter $\boldsymbol{B}$ (see Fig. 2).

$$bg(x, y) = \frac{1}{32}\sum_{i=1}^{5}\sum_{j=1}^{5} p(x-3+i, y-3+i) \times \boldsymbol{B}(i, j) \tag{6}$$

1	1	1	1	1
1	2	2	2	1
1	2	0	2	1
1	2	2	2	1
1	1	1	1	1

Fig. 2 Matrix $\boldsymbol{B}$ for weighted low-pass filtering

The function $f_2(bg(x, y))$ computes the visibility threshold from the luminance contrast as

$$f_2(bg(x, y)) = \begin{cases} T_0\left(1 - \left(\dfrac{bg(x, y)}{127}\right)^{1/2}\right) + \varepsilon & bg(x, y) \leqslant 127\\ \gamma(bg(x, y) - 127) & bg(x, y) > 127\end{cases} \tag{7}$$

where T_0 is the visibility threshold when the background luminance level is 0 and ε denotes the minimum visibility threshold. γ is the visibility threshold when the background luminance level reaches the maximum. This function shows that the visibility threshold has a square root relationship with low background luminance and a linear relationship with higher background luminance.

1.2 Temporal JND model

In addition to the spatial masking effect, the temporal masking effect should also be considered to build the spatial-temporal JND (STJND) model for video signals. A greater inter-frame luminance difference usually results in a greater temporal masking effect. Based on Ref. [16], the temporal JND is defined as

$$\mathrm{TJND}(x, y, t) = \begin{cases} \max\left(\tau, \dfrac{H}{2}\exp\left(-\dfrac{0.15}{2\pi}(\Delta(x, y, t) + 255)\right)\right) + \tau & \\ \qquad \Delta(x, y, t) \leqslant 0 & \\ \max\left(\tau, \dfrac{L}{2}\exp\left(-\dfrac{0.15}{2\pi}(255 - \Delta(x, y, t))\right)\right) + \tau & \\ \qquad \Delta(x, y, t) > 0 & \end{cases} \tag{8}$$

where H and L are model parameters. The value $\tau = 0.8$ is based on the conclusion in Ref. [16] stating that the scale factor should be reduced to 0.8 when $\Delta(x, y, t) < 5$, in order to minimize the allowable distortion in stationary regions. The quantity $\Delta(x, y, t)$ denotes the average luminance difference between frame t and the previous frame $t-1$.

$$\Delta(x, y, t) = \frac{p(x, y, t) - p(x, y, t-1)}{2} + \frac{bg(x, y, t) - bg(x, y, t-1)}{2} \tag{9}$$

1.3 STJND-based segmentation algorithm

Our STJND is then defined as

$$\mathrm{STJND}(x, y, t) = \mathrm{SJND}(x, y)\,\mathrm{TJND}(x, y, t) \quad (10)$$

The STJND model exploits the HVS visual sensitivity to luminance contrast, as well as the spatial and temporal masking effects. The STJND model provides the visibility threshold of each pixel of an image by assuming that the pixel is perceived at the highest visual acuity.

Since an accurate segmentation is of great importance for the subsequent synthesis process, we adopt the STJND profile to detect and segment the texture regions from each texture frame. Compared with other JND models, the STJND takes full consideration of the luminance adaptation, texture masking and their overlapping effects.

We apply the following principle to detect texture regions in this paper. If one pixel has a relatively greater JND value, it is of less importance to human viewers, and, consequently, it can be classified as a candidate pixel within texture regions. In order to be compatible with the existing video coding frameworks[1-2], the texture region segmentation is performed in the unit of a macro-block (MB). The JND energy of one MB is defined as

$$\mathrm{JND}_{\mathrm{MB}} = \frac{1}{256}\sum_{i=0}^{15}\sum_{j=0}^{15}\mathrm{JND}(i, j) \quad (11)$$

where $\mathrm{JND}(i, j)$ represents the JND value located at (x, y). If one MB has a JND energy higher than a given threshold, it is defined to be a candidate texture MB; otherwise, it is a non-texture MB. The threshold of the JND energy is defined as the average of all the MBs' JND energy in a texture frame,

$$\mathrm{JND}_{\mathrm{threshold}} = \frac{1}{\mathrm{total\_MB}}\sum_{\mathrm{MB}}\mathrm{JND}_{\mathrm{MB}} \quad (12)$$

One segmentation result by the aforementioned method is depicted in Fig. 3(b), from which we can observe several isolated MBs in the detected texture regions. To remove the isolated MBs, an iterative row and column scanning algorithm is devised. We first scan the texture MBs row by row, and then scan the texture MBs column by column. The row scanner removes horizontally isolated MBs, which have no adjacent texture MBs in the same row. And then the column scanner removes the vertically isolated texture MBs in the same way. The row scanner and column scanner are repeated until there are no isolated texture MBs in the detected texture regions. The result is shown in Fig. 3(c), where isolated MBs are removed. To achieve better segmentation results, only the largest connected texture region is selected, while others are ignored. The final detected texture region is depicted in Fig. 3(d).

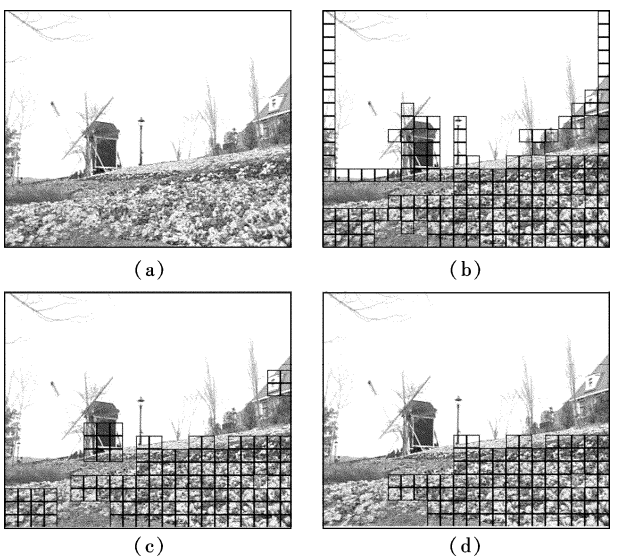


Fig. 3 Original picture and JND segmentation results at different stages. (a) Original picture; (b) JND selected result with isolated MBs; (c) Row and column scanner result without isolated MBs; (d) Final detected region

2 AR-Based Texture Synthesizer

In the proposed AR-based synthesizer, each pixel in the texture region is generated as a linear combination of pixels taken from the closest forward and backward reference frames. In our scheme, the synthesized pixel $\tilde{p}_t(m, n)$ is interpolated as

$$\tilde{p}_t(m, n) = \sum_{k=0}^{L-1}\sum_{l=0}^{L-1}\hat{p}_{t-1}(\hat{m}, \hat{n})\,W_{\mathrm{f}}(k, l) + \sum_{u=0}^{L-1}\sum_{v=0}^{L-1}\hat{p}_{t+1}(\breve{m}, \breve{n})\,W_{\mathrm{b}}(u, v) \quad (13)$$

with $\hat{m} = m - L/2 + k + \boldsymbol{V}_{x,\mathrm{f}}$, $\hat{n} = n - L/2 + l + \boldsymbol{V}_{y,\mathrm{f}}$, $\breve{m} = m - L/2 + u + \boldsymbol{V}_{x,\mathrm{b}}$, and $\breve{n} = n - L/2 + v + \boldsymbol{V}_{y,\mathrm{b}}$. Here $\boldsymbol{V}_{x,\mathrm{t}}$ and $\boldsymbol{V}_{y,\mathrm{f}}$ represent the forward motion vectors in the horizontal and vertical directions; $\boldsymbol{V}_{x,\mathrm{b}}$ and $\boldsymbol{V}_{y,\mathrm{b}}$ represent the backward motion vectors in the horizontal and vertical directions; $\hat{p}_{t-1}(\hat{m}, \hat{n})$ and $\hat{p}_{t+1}(\breve{m}, \breve{n})$ represent the corresponding reconstructed pixels along the motion trajectory in the forward and backward reference frames; $W_{\mathrm{f}}(k, l)$ and $W_{\mathrm{b}}(u, v)$ represent AR parameters pointing to the forward and backward reference frames; L represents the window size of the AR model. An example of the AR model with $L = 3$ is illustrated in Fig. 4.

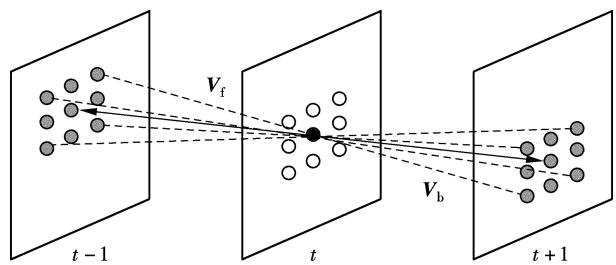


Fig. 4 Example of the AR model with $L = 3$

In contrast to the non-texture MBs, the texture ones have no motion information in the encoder or the decoder. For a better perceptual result, the direct mode is employed to find corresponding forward and backward blocks for the texture block in the current texture frame. When the co-located MB is intra mode, the direct mode is simply replaced by the spatial direct mode.

AR parameters $W_{\mathrm{f}}(k, l)$ and $W_{\mathrm{b}}(u, v)$ are computed by minimizing the sum of square error e between the original pixel values and the synthesized pixel values in the texture region.

$$e = \sum_{(m, n)\in\text{texture\_region}}(p_t(m, n) - \tilde{p}_t(m, n))^2 \quad (14)$$

where $p_t(m, n)$ represents the original pixel value located at (m, n). Substituting (13) into (14), we obtain

$$e = \sum_{(m,n)\in \text{texture\_region}} \left[\sum_{k=0}^{L-1}\sum_{l=0}^{L-1} \hat{p}_{t-1}(\hat{m}, \hat{n}) W_f(k, l) + \sum_{u=0}^{L-1}\sum_{v=0}^{L-1} \hat{p}_{t+1}(\breve{m}, \breve{n}) W_b(u, v) - p_t(m, n) \right]^2 \quad (15)$$

According to the least-squares method, AR parameters can be derived by setting

$$\frac{\partial e}{\partial W_f(k, l)} = 0, \quad \frac{\partial e}{\partial W_b(u, v)} = 0 \quad (16)$$

In this paper, each texture frame has unique AR parameters, which are written into the bit-stream and sent to the decoder. At the decoder side, AR parameters are decoded and the same synthesis is performed.

3 Experimental Results and Analysis

The proposed segmentation and synthesis methods are integrated into the H. 264/AVC reference software JM10. 1. Our experiments are conducted with the GOP structure (e. g., IBBBP) and rate distortion optimization is enabled. The quantization parameters QP are set to 30, 32, 34 and 36, respectively. To validate the performance of the proposed method, three standard sequences (mobile, coastguard and flower garden), which are full of rigid textures, non-rigid textures, and detailed textures, are tested.

Tab. 1 shows the bit-rate savings on sequences under different QPs. For sequence mobile, bit-rate savings ranging from 15. 55% to 19. 86% are achieved, and for coastguard, the bit-rate savings range from 17. 58% to 23. 18%. While higher bit-rate savings of more than 50% are achieved for the sequence flower garden, in which nearly half of the regions are segmented as texture regions.

Tab. 1 Bit-rate savings on sequences under different QPs %

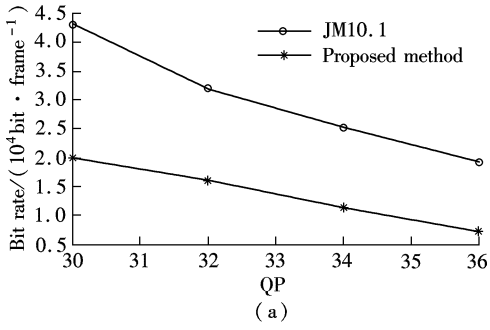
Video sequence	Average bits saving			
	QP = 30	QP = 32	QP = 34	QP = 36
Mobile	15. 55	16. 49	17. 58	19. 86
Coastguard	17. 58	20. 99	21. 89	23. 18
Flower garden	55. 00	55. 89	57. 11	58. 42

Fig. 5 shows the 19th reconstructed frames in flower garden by the proposed methods and the anchor reference of JM10. 1. The bits spending on Fig. 5(a) and Fig. 5(b) are 37 664 bits and 18 448 bits, respectively. And the peak signal noise ratios (PSNR) of these two pictures compared with the original one are 36. 76 dB and 31. 35 dB, respectively. However, it is very difficult to distinguish Fig. 5(a) and Fig. 5(b) in terms of visual quality. That is because human viewers pay less attention to the detected regions (flower regions) in these two pictures. Consequently, the synthesized texture region in Fig. 5(b), which exhibits similar semantics rather than pixel-by-pixel fidelity as in Fig. 5(a), achieves satisfactory results.

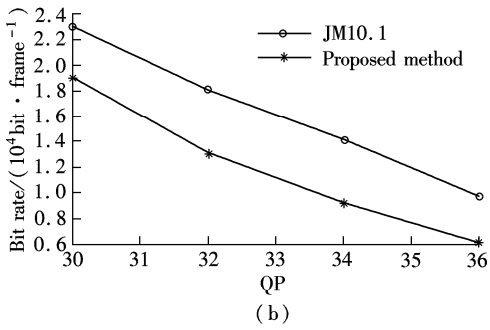
Fig. 6 depicts the bit rates spending on texture frames within the sequences mobile and coastguard by the proposed method and the anchor reference of JM10. 1. It can be easily observed that the proposed method can effectively reduce the bit rates while maintaining the same perceptual quality.

(a) (b)

Fig. 5 The 19th reconstructed texture frame of flower garden. (a) JM10. 1; (b) Proposed method



(a)



(b)

Fig. 6 Bit-rate spending on texture frames by different methods. (a) Mobile; (b) Coastguard

4 Conclusion

Considering the texture regions in most video scenes, a perceptual segmentation algorithm and an AR-based synthesis method are proposed in this paper. The proposed methods are integrated into the H. 264/AVC reference codec JM10. 1. The texture frame is first divided into texture regions and non-texture regions by the JND-based segmentation, and then an AR-based synthesizer is performed on the texture regions. Experimental results verify that the proposed methods can effectively reduce the bit rates while maintaining good perceptual quality.

References

[1] ISO/IEC 15444-1 Team. Our new standard [EB/OL]. (2000-03-11) [2010-04-20]. http://www. jpeg. org/jpeg2000/.

[2] Wikipedia. H. 264/MPEG-4 AVC [EB/OL]. (2003-11-25) [2010-04-20]. http://en. wikipedia. org/wiki/H. 264/.

[3] Ndjiki-Nya P, Wiegand T. Video coding using texture analysis and synthesis [C]//*Proceedings of Picture Coding*. Saint-Malo, France, 2003: 489 - 497.

[4] Zhang Y, Ji X, Zhao D, et al. Video coding by texture analysis and synthesis using graph cut [C]//*Proceedings of Pacific-Rim Multimedia Conference*. Heidelberg, Germany,

2006: 582 - 589.
[5] Zhu C, Sun X, Wu F, et al. Video coding with spatio-temporal texture synthesis[C]//*Proceedings of the IEEE International Conference on Multimedia and Expo*. Beijing, China, 2007: 112 - 115.
[6] Yang X K, Ling W S, Lu Z K. Just noticeable distortion model and its applications in video coding [J]. *Signal Processing*: *Image Communication*, 2005, **20**(7): 662 - 680.
[7] Wu X, Barthel K U, Zhang W. Piecewise 2D auto-regression for predictive image coding [C]//*Proceedings of the IEEE International Conference on Image Processing*. Chicago, USA, 1998: 901 - 904.
[8] Tugnait J K. Texture synthesis using asymmetric 2-D non-causal AR models [C]//*Proceedings of the IEEE International Conference on Image Processing*. South Lake Tahoe, USA, 1993: 71 - 75.
[9] Chou C H, Li Y C. A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile [J]. *IEEE Transactions on Circuits System and Video Technology*, 1995, **5**(12): 467 - 476.
[10] Yang X, Lin W, Lu Z, et al. Motion-compensated residue preprocessing in video coding based on just-noticeable-distortion profile [J]. *IEEE Transactions on Circuits System and Video Technology*, 2005, **15**(6): 742 - 752.
[11] Safranek R J, Johnston J D. A perceptually tuned subband image coder with image dependent quantization and post-quantization data compression [C]//*Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*. Glasgow Scotland, UK, 1989: 1945 - 1948.
[12] Watson A B, Yang G Y, Solomon J A, et al. Visibility of wavelet quantization noise [J]. *IEEE Transactions on Image Processing*, 1997, **6**(8): 1164 - 1175.
[13] Watson A B. Perceptual optimization of DCT color quantization matrices [C]//*Proceedings of IEEE International Conference on Image Processing*. Austin, USA, 1994: 100 - 104.
[14] Lubin J. A visual system discrimination model for imaging system design and evaluation [J]. *Vision Models for Target Detection and Recognition*, 1995, **6**(15): 245 - 283.
[15] Netravali A N, Prasada B. Adaptive quantization of picture signals using spatial masking [J]. *IEEE Transactions on Signal Processing*, 1977, **65**(4): 536 - 548.
[16] Chou C H, Chen C W. A perceptually optimized 3-D subband codec for video communication over wireless channels [J]. *IEEE Transactions on Circuits System and Video Technology*, 1996, **6**(4): 143 - 156.

基于 JND 和 AR 模型的感知视频编码方法

王 翀 赵 力 邹采荣

(东南大学信息科学与工程学院,南京 210096)

摘要:为了达到减少比特数同时保持画面质量的目的,提出了一种基于最小可视失真(JND)和自回归(AR)模型的感知视频编码方法. 首先,设计了基于 JND 的纹理分割算法,建立了空时 JND 模型,以 MB 为基本单元,通过计算其 JND 能量并与阈值做比较,用以分割出视频序列中的纹理区域. 然后,开发了 AR 模型来合成纹理区,在使用最小二乘法计算出 AR 模型的参数后,用相邻的前后参考帧对应像素的线性插值来生成重构像素. 最后,为了检验所提方法的效果,将其与 H.264/AVC 视频编码系统做比较,用不同的视频序列实验来验证所提方法的有效性. 实验结果显示,对于具有不同纹理特点的实验序列,所提方法可以在保持感知质量的同时将比特率减少 15% ~58%.

关键词:感知视频编码;纹理合成;最小可视失真;AR 模型

中图分类号:TN911.73