SURVEILLANCE VIDEO CODING WITH DYNAMIC TEXTURAL BACKGROUND DETECTION

Kun Yang, Fangdong Chen, Dong Liu, Zhibo Chen, Weiping Li

CAS Key Laboratory of Technology in Geo-spatial Information Processing and Application System
University of Science and Technology of China, Hefei 230027, China
Email : chenzhibo@ustc.edu.cn

ABSTRACT

Texture scenes like flickering flames, swaying tree branches, flowing water exhibit a complex stochastic motion character. It presents a great challenge to compress these dynamic texture efficiently even with the state-of-the-art video encoder. Furthermore, these contents only contain a little helpful information in surveillance analysis. In this paper, we propose an approach for compressing the dynamic textures in the surveillance video. In the proposed scheme, the dynamic texture contents are detected by the histogram of motion direction (HMD) algorithm, and then removed at the encoder, these dynamic texture contents will be restored at the decoder directly. Objective and subjective results are presented, demonstrating that the proposed approach provides about 8.7% bitrate saving with visually plausible dynamic textures in comparison with High Efficiency Video Coding (HEVC).

Index Terms— dynamic texture, detection, HVS, surveillance video, HEVC

1. INTRODUCTION

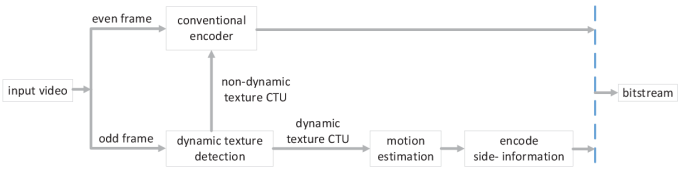
The concept of dynamic textures was first proposed in [1] by extending the static texture in conventional image from the spatial domain to temporal domain. Obviously, dynamic textures exhibit high frequency information in the spatial domain and stochastic motion in the temporal domain. These video contents consume large amounts of bits even when compressed by state-of-the-art technique, like HEVC. It is observed that the dynamic texture region will be split into small Coding Units (CU) which add more overheads. Moreover the residual of these regions usually has lots of high frequency energy, and it is difficult to encode them by entropy coding efficiently.

However, natural dynamic textures usually contain little semantic information, especially in the surveillance video, which usually belongs to the background. According to the Human Visual System (HVS) [2], human eyes are less sensitive to these higher spatial frequencies, and this property can be used in the design of a novel perceptive dynamic texture coding technique. At the same time, some recent dynamic texture synthesis works [3],[4] can generate visually similar dynamic textures given some dynamic texture frames as input. These works can be exploited for video coding application to remove the visual perception redundancy at the dynamic texture region.

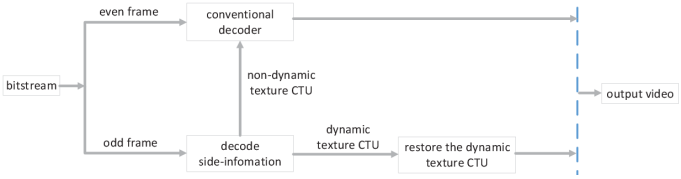
In [5], it is assumed that the viewers prefer semantic meaning of the texture than the specific details therein. And this assumption is adopted to a closed-loop, content-based approach, where the detail-irrelevant textures are analyzed at the encoder, and synthesized at the decoder guided by side information. However, the texture detector and synthesizer focus on the texture content given by a moving camera. [6] proposed to classify the regions into textural or structural region based on two motion models. And then textural and structural blocks are selectively removed at the encoder and recovered by a patched-based texture synthesis algorithm at the decoder. [7] presented a dynamic texture prediction (DTP) algorithm which is to use frames stored in the reference picture buffer for learning and then predict the next picture to be encoded. The following work [8] also used the synthesized frames for prediction to achieve bitrate saving under the conventional rate-distortion criterion in H.264. Beyond that, [8] proposed to synthesize the dynamic textures by a linear dynamic model.

In this work, we propose a novel surveillance video coding scheme based on the HVS property and dynamic texture detection/synthesis technique. Different from the above mentioned works, we focus on surveillance video where the camera is almost static. Dynamic texture content in surveillance is usually irrelevant to viewers. Therefore, we detect dynamic textures at the encoder and send some side information to the decoder, and then these contents will be restored guided by the side information directly instead of being decoded by conventional methods at the decoder. The proposed coding scheme can be integrated into the HEVC easily.

This work was supported by the National Key Research and Development Plan under Grant 2016YFC0801001, by the National Program on Key Basic Research Projects (973 Program) under Grant 2015CB351803, by the Natural Science Foundation of China (NSFC) under Grants 61571413, 61390514, and 61632001, and Intel ICRI MNC.



(a)



(b)

Fig. 1. Flowchart of the proposed scheme. (a) Encoder, (b) Decoder.

The remainder of this paper is organized as follows: In Section II, the overview of the proposed approach is presented. In Section III, we describe the dynamic texture detection and synthesis technique in detail. In Section IV, objective and subjective experiment results are presented to evaluate the proposed coding scheme. Finally, we conclude in Section V.

2. CODING SCHEME OVERVIEW

In surveillance applications, the dynamic texture contents are usually unimportant to surveillance analysis, such as object detection, object tracking, activity analysis, etc. In order to compress these contents efficiently at a high visual quality, we propose a coding scheme based on the dynamic texture detection and synthesis.

In our proposed coding scheme, the surveillance videos are processed with a group of pictures (GOP) as a unit. We begin to detect the dynamic texture regions in every frame from the second GOP. And the dynamic texture detection is performed on the CTU (64x64) level. The coding scheme can be applied to random access or low delay coding structure.

The encoder framework of the proposed scheme is depicted in Fig.1(a). For every even frame in the GOP, it is compressed by the conventional techniques to avoid the error accumulation. For the odd frames, we detect the dynamic texture CTU and only side-information such as CTU types, motion parameters, are encoded to guide the restoration at the decoder. The decoding flowchart is depicted in Fig.1(b). Side information is decoded for odd frames to restore the dynamic texture CTU at the decoder, while the even frame is decoded by conventional decoding techniques in HEVC.

3. DYNAMIC TEXTURE DETECTION AND SYNTHESIS

The dynamic texture detection algorithm identifies the dynamic texture CTUs, and generates a mask. This module has to avoid mistaking the foreground, such as pedestrian and vehicle, for dynamic texture. And the dynamic texture synthesis algorithm should restore the detected region at the decoder while keep them coherent to both spatial and temporal surrounding contents.

Algorithm 1 Dynamic Texture Detection Algorithm

Input: current CTU at time t C_t, the co-located CTU at time t-1 C_{t-1} and t-4 C_{t-4}

Output: the background block mask M_{back}, the foreground block mask M_{fore} and the dynamic texture block mask M_{dyn}

1: Initialize M_{back}, M_{fore} and M_{dyn}
2: Detect the foreground and the background using the difference value between C_t and C_{t-1}, C_t and C_{t-4}
3: Split the C_t into non-overlapped 2x2 cells, and estimate the motion field
4: Split the C_t into non-overlapped 8x8 blocks, and compute the histogram of motion direction for each block
5: Determine whether the C_t is a dynamic texture CTU by the statistics distribution of histogram
6: Refine the M_{dyn} by the spatial and temporal information
7: Output M_{back}, M_{fore} and M_{dyn}

3.1. Dynamic texture detection

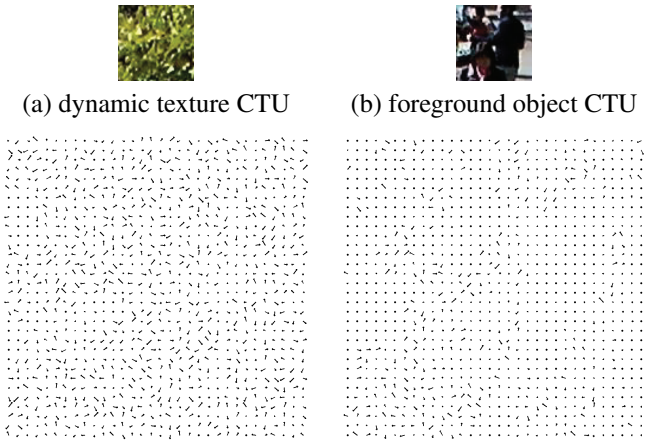
Many researchers [9], [10], [11] have previously worked on techniques for characterizing dynamic textures based on different statistics. However, their applications are mostly focused on the texture recognition and classification. In this paper, we propose a novel detection algorithm based on the histogram of motion direction. The pseudo code for the detection process is presented in the Algorithm 1.

3.1.1. Background/foreground detection

To suppress the interference of the background and the foreground, we detect the background and the foreground region first. A simple detection algorithm based on the difference value between current frame and the last frame, current frame and the last fourth frame is performed for every frame. After this preprocessing, all the CTUs will be classified into 3 categories roughly: background, foreground and dynamic texture candidates. As a result, three masks indicating different type of the CTU in every frame will be generated. This preprocessing can also educe the complexity in next step.

3.1.2. Dynamic texture region identification

For the dynamic texture candidates, we need to identify whether these regions are dynamic texture regions further. The dynamic textures typically feature local random motion activity in both spatial and temporal domain. The motion direction and displacement of one small block are usually independent to the surroundings, as we can see in the Fig.2(a) and Fig.2(c). However, the motion of the foreground object are usually consistent to the surrounding motion in spatial or temporal domain, as shown in Fig.2(b) and Fig.2(d).



(a) dynamic texture CTU (b) foreground object CTU

(c) motion field of (a) (d) motion field of (b)

Fig. 2. The motion field of different CTUs.

We propose a novel histogram of motion direction (HMD) algorithm to capture the motion feature mentioned above. Firstly, the CTU is split into non-overlapped 2x2 cells. Then, we do motion estimation for every cell in the dynamic texture candidates. Compared with the dense optical flow or the pixel-wise motion estimation, the proposed approach is more robust to the noise. As a result of this process, we get a motion field for every dynamic texture candidate on the cell level. In this paper, we focus on the stochastic character of the motion vector direction. Because the displacement of different dynamic texture content varies widely, however, the motion vector direction always ranges from 0 to 360 degree. We divide 0-360 degree into 8 bins, which means every bin consists of 45 degree angle. To capture the homogeneous random motion within the CTU, we split the CTU into non-overlapped 8*8 blocks, that is, each block consists of 16 cells. After that, we compute the motion vector direction of each cell in the block and classify it to the appropriate bin. At last, we get a histogram of motion vector direction. The flowchart of HMD is shown as Fig.3.

The proposed analysis method captures dynamic texture characteristic by the motion vector direction histogram. The underlying hypothesis of the detection algorithm is that the motion vector direction of dynamic texture is consistent with random distribution. Hence, the histogram of motion direction is denser, and the current CTU is more likely to be a dynamic texture content. We use the sparsity of each histogram as a metric proposed by [12] to evaluate the random characteristic of the distribution:

$$SI_H = \frac{\sqrt{n} - \frac{\|Y\|_1}{\|Y\|_2}}{\sqrt{n} - 1}$$

where n is the number the motion vectors, Y is the number of motion vector direction at each bins. SI_H ranges from 0 to 1, with higher values corresponding to sparser distribution. If SI_H is smaller than the threshold, the current CTU is

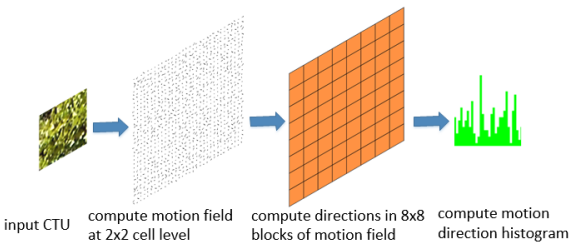


Fig. 3. Generating the histogram of motion direction.

dynamic texture content.

3.1.3. spatial-temporal refinement

In view of the consistency of the object motion, the dynamic texture region also has a strong consistency in temporal and spatial domain. If the surrounding CTU of the current CTU in same frame or its co-located CTU in the last frame is dynamic texture CTU, the current CTU is more likely to be a dynamic texture CTU too. So, to detect the dynamic texture CTU more accurately, we refine the detection result by the surrounding CTUs.

1	2	1
2		2
1	2	1

Fig. 4. Weights of eight neighbors.

For every CTU which is not a dynamic texture CTU, we take into account its 8 neighbors to determine whether it is a dynamic texture CTU again. Firstly, we assign different weight to the neighbor according to the distance, as shown in Fig.4. And then, we calculate a score S:

$$S = \sum_{b_i \in N_{ei}} D \cdot W$$

where N_{ei} stands for 8 neighbors of current CTU, W is the weight of the neighbor. And if current CTU is dynamic texture CTU, D=1, else D=0. If S is greater than 4 or the co-located CTU is dynamic texture CTU, we relax the threshold to identify whether the current CTU is a dynamic texture again. At last, we get a dynamic texture mask for every frame as shown in Fig.5, in which the red block stands for the dynamic texture region.

3.2. Dynamic texture synthesis

With the side information decoded from the bitstream, we propose a simple method to restore the dynamic texture CTU. For the dynamic texture synthesis, we must obey the following principles: to keep temporal consistency to avoid flickering artifacts and to preserve spatial smoothness to avoid blocking effect.

Every local dynamic texture sample is assumed to be predictable from temporal neighboring sample within a limited

(a) (b)

Fig. 5. The results of the dynamic texture detection: (a) detected by proposed algorithm, (b) drawn by hand perceptually.

spatial range in short time, but independent to the rest. To restore the dynamic texture CTU at the decoder with a low complexity, we adopt a method based on motion estimation and motion compensation within a limited range. The dynamic texture CTU in the odd frame finds the most similar block by sub-pixel motion estimation at the encoder. And then these blocks can be reconstructed by motion compensation at the decoder. Therefore, only the dynamic texture CTU flag and motion vectors need to be transmitted to the decoder, which reduce the decoding complexity significantly.

4. EXPERIMENTS

Experiments have been conducted to evaluate the performance of the proposed approach compared with the state of art codec. Since our proposed approach focuses on surveillance video, the test datasets used in this work comprise 10 sequences captured from real campus camera. All the sequence is at a resolution of 1920x1080. And these sequences cover different surveillance scenes, including rainy and fine weather, bright and dusky lightness.

4.1. Objective Experiment

The HEVC test model HM-12.0 is adopted as the anchor. And all the sequences are encoded under the low delay coding configurations in the experiment. Table 1 illustrates the overall results of bitrate saving and time changing by comparing the proposed approach with the anchor. It reveals that the proposed coding scheme reduces 10.6%, 7.6%, 3.4%, 1.0% bit rate at four different QP, and it provides 8.7% bits reduction on average. But its performance loses on some sequences at QP=37, this is because the side information of our method cost more than the simple mode (like skip mode) at QP=37. Meanwhile, our experimental results show that the proposed approach reduce the encoding and decoding complexity, it saves about 4% at the encoder and 3% at the decoder.

4.2. Subjective Experiment

The pixel-wise distortion measure (e.g. PSNR, MSE) used in hybrid video codecs like HEVC may not be an adequate coding distortion criterion for high frequency content. Although many studies have been conducted to propose a new quality metrics that correlate with the HVS, it remains a challenging

Table 1. RESULTS OF OBJECTIVE EXPERIMENTS

Sequences	Bitrate Saving				Coding Time	
	QP=22	QP=27	QP=32	QP=37	Enc	Dec
JingChunRoadNorthAfternoon	-14.1%	-7.9%	-2.0%	1.6%	86%	96%
JingChunRoadNorthRain0	-14.9%	-11.5%	-5.6%	-1.9%	92%	94%
JingChunRoadNorthRain1	-12.0%	-9.3%	-4.3%	-1.5%	96%	94%
JingChunCrossRoadRain	-4.9%	-3.6%	-0.8%	0.8%	103%	100%
WeiMingLakeEastNoon	-12.1%	-10.0%	-6.5%	-3.6%	97%	95%
WeiMingLakeWestMorning1	-7.1%	-5.5%	-3.2%	-1.6%	101%	102%
WeiMingLakeWestMorning2	-5.4%	-4.1%	-2.3%	-1.1%	102%	101%
WeiMingLakeWestNoon0	-10.6%	-6.0%	-1.3%	0.2%	99%	105%
WeiMingLakeWestNoon1	-13.1%	-9.4%	-4.1%	-1.6%	89%	91%
WeiMingLakeWestNoon2	-11.9%	-8.7%	-4.1%	-1.5%	91%	95%
OVERALL	-10.6%	-7.6%	-3.4%	-1.0%	96%	97%

Table 2. RESULTS OF SUBJECTIVE EXPERIMENTS

Sequences	Subject										AVE
	1	2	3	4	5	6	7	8	9	10	
JingChunRoadNorthAfternoon	-1	-1	-1	0	0	-1	-1	0	0	0	-0.5
JingChunRoadNorthRain1	-1	2	-1	-1	0	-1	0	0	2	0	0
JingChunCrossRoadRain	0	0	1	0	0	0	0	0	0	0	0.1
WeiMingLakeWestMorning1	-1	1	-1	0	-1	0	-1	-1	-1	-1	-0.6
WeiMingLakeWestMorning2	0	1	0	0	-1	-1	-2	-1	0	-1	-0.5
WeiMingLakeWestNoon0	0	1	0	0	0	0	0	0	0	0	0.1
WeiMingLakeWestNoon2	0	0	0	-1	0	-1	-1	0	-1	0	-0.4
AVE	-0.43	0.57	-0.29	-0.29	-0.29	-0.57	-0.71	-0.29	0	-0.29	-0.26

task. Hence, subjective experiments have to be performed to assess the quality of the synthesis.

We choose the the stimulus-comparison (SC) method to do the subjective experiment under the ITU-R BT.2021-1 standard[13]. In the subjective experiment, every subject will watch a pair of video, the original video and synthesized video. And then they are asked to compare the video and select one score they preferred. There are 5 discrete difference levels (-2, -1, 0, 1, 2: better, slightly better, the same, slightly worse, worse, '-' indicate the anchor is better)for rating the video difference. 7 sequences are selected from above dataset, other 3 sequences are used for training. All the test sequences are compressed at 4 different QP: 22, 27, 32, 37. 10 volunteers (5 men and 5 women) participated in the experiment. And the result at QP 22 is shown in Table 2.

We can see that the average is -0.26, which indicates that the subjects generally think synthesized video is very similar to anchor. Moreover, the standard deviation is 0.63, it reveals that the subjects hold different opinions about the quality of synthesized video. That is, it is difficult to identify the difference between the anchor and the synthesized video. Overall, the proposed approach can generate the dynamic texture content at a high visual quality.

5. CONCLUSIONS

In this paper, we propose a novel coding scheme for the surveillance video in which the dynamic texture contents are skipped at the encoder and restored at the decoder. The proposed approach has been implemented into the HM-12.0. And the result showes that it can achieve 8.7% bitrate reduction on the surveillance video. The subjective experiment showes that the decoded sequences are almost exactly the same with the input sequences.

6. REFERENCES

[1] Gianfranco Doretto, Alessandro Chiuso, Ying Nian Wu, and Stefano Soatto, "Dynamic textures," *International Journal of Computer Vision*, vol. 51, no. 2, pp. 91–109, 2003.

[2] Christian J Van den Branden Lambrecht and Olivier Verscheure, "Perceptual quality measure using a spatiotemporal model of the human visual system," in *Electronic Imaging: Science & Technology*. International Society for Optics and Photonics, 1996, pp. 450–461.

[3] Jianwen Xie, Song-Chun Zhu, and Ying Nian Wu, "Synthesizing dynamic textures and sounds by spatial-temporal generative convnet," *arXiv preprint arXiv:1606.00972*, 2016.

[4] Xinge You, Weigang Guo, Shujian Yu, Kan Li, José C Príncipe, and Dacheng Tao, "Kernel learning for dynamic texture synthesis," *IEEE Transactions on Image Processing*, vol. 25, no. 10, pp. 4782–4795, 2016.

[5] Patrick Ndjiki-Nya, Tobias Hinz, Aljoscha Smolic, and Thomas Wiegand, "A generic and automatic content-based approach for improved h. 264/mpeg4-avc video coding," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*. IEEE, 2005, vol. 2, pp. II–874.

[6] Chunbo Zhu, Xiaoyan Sun, Feng Wu, and Houqiang Li, "Video coding with spatio-temporal texture synthesis and edge-based inpainting," in *Multimedia and Expo, 2008 IEEE International Conference on*. IEEE, 2008, pp. 813–816.

[7] Aleksandar Stojanovic and Philipp Kosse, "Extended dynamic texture prediction for h. 264/avc inter coding," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 2045–2048.

[8] Johannes Balle, Aleksandar Stojanovic, and Jens-Rainer Ohm, "Models for static and dynamic texture synthesis in image and video compression," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 7, pp. 1353–1365, 2011.

[9] Guoying Zhao and Matti Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, 2007.

[10] Avinash Ravichandran, Rizwan Chaudhry, and René Vidal, "View-invariant dynamic texture recognition using a bag of dynamical systems," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 1651–1657.

[11] Konstantinos G Derpanis and Richard P Wildes, "Dynamic texture recognition based on distributions of spacetime oriented structure," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 191–198.

[12] Michael W Spratling, "Image segmentation using a sparse coding model of cortical area v1," 2013, vol. 22, pp. 1631–1643, IEEE.

[13] IT Union, "Subjective methods for the assessment of stereoscopic 3dtv systems," *Recommendation ITU-R BT*, vol. 2021, 2015.