SURVEILLANCE VIDEO CODING WITH VEHICLE LIBRARY

Changyue Ma[1], *Dong Liu*[1]*, *Xiulian Peng*[2], *Feng Wu*[1]

[1]CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System,
University of Science and Technology of China, Hefei 230027, China
cyma@mail.ustc.edu.cn, {dongeliu,fengwu}@ustc.edu.cn
[2]Microsoft Research Asia, Beijing 100080, China, xipe@microsoft.com

ABSTRACT

Inter prediction in video coding is very efficient to remove temporal redundancy. However, due to the limitation of short-term references, inter prediction can work only within a very short time interval. In surveillance videos, we observe that there are always similar vehicles passing through one static camera, but the time intervals of similar vehicles are usually several seconds to minutes, exceeding the time interval that short-term references can handle. To solve this problem, we propose to build a vehicle library, and to put high-quality copies of the similar vehicles into the vehicle library. During encoding, vehicles are detected from the current frame, and for each vehicle we can retrieve similar vehicles from the vehicle library, and take the retrieved vehicle picture as additional references for inter prediction. Preliminary experimental results show that the proposed vehicle library based method achieves as high as 10.1% bit-rate saving for surveillance video coding, compared to HEVC anchor.

Index Terms— HEVC, Inter prediction, Surveillance video, Vehicle library.

1. INTRODUCTION

With the development of smart city, the amount of surveillance video data has shown exponential growth. Currently, surveillance videos are usually compressed by the video coding standards such as H.264/AVC [1], which are designed for general video coding. However, as there are some special characteristics in surveillance video, e.g. the camera is relatively static, directly applying general video coding methods on surveillance video cannot make full use of these characteristics. As a result, a lot of researches have focused on surveillance video coding to exploit the special characteristics.

Generally, the content in surveillance video can roughly be divided into foreground and background. For background, under the assumption of static camera, several background modeling based methods have been proposed [2]. For foreground, model based and object oriented methods [3, 4] are proposed at earlier years, which propose to segment the foreground objects and then encode them separately. Since the object segmentation is difficult in complex scenarios and the overhead bits for object representation are excessive, these methods have not been adopted practically. Later, based on the hybrid block-based coding framework, the authors in [5] propose to differentiate and process the background and foreground blocks separately, and modify the inter prediction methods for foreground objects, which efficiently improves the foreground coding efficiency.

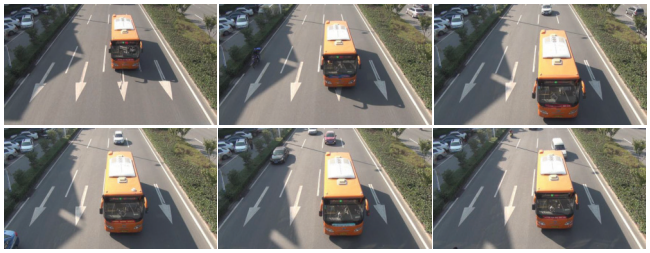

Fig. 1: Some frames selected from a one-hour surveillance video, showing similar vehicles that pass through the same camera at different timestamps.

In recent years, methods have been proposed to improve the foreground coding through removing the global redundancy in surveillance video, especially for vehicles [6, 7]. As the vehicles move and disappear from one camera to appear in another, the authors in [7] propose to build a 3D vehicle model library for surveillance video coding. The sketch part of the vehicle is predicted via vehicle library, whilst the texture part is predicted via short-term inter prediction. The proposed methods greatly improve vehicle coding efficiency, but there are some kinds of prior information required, including the 3D vehicle models, the camera parameters, the position and pose of vehicles, and so on [7], which are difficult to obtain or estimate in real applications.

In this paper, we also target vehicles in surveillance video coding, but we consider the global redundancy not in spatial

* Corresponding author. This work was supported by the National Key Research and Development Plan under Grant 2016YFC0801001, by the Natural Science Foundation of China (NSFC) under Grants 61390512, 61331017, and 61632001, and by the Fundamental Research Funds for the Central Universities under Grant WK3490000001.

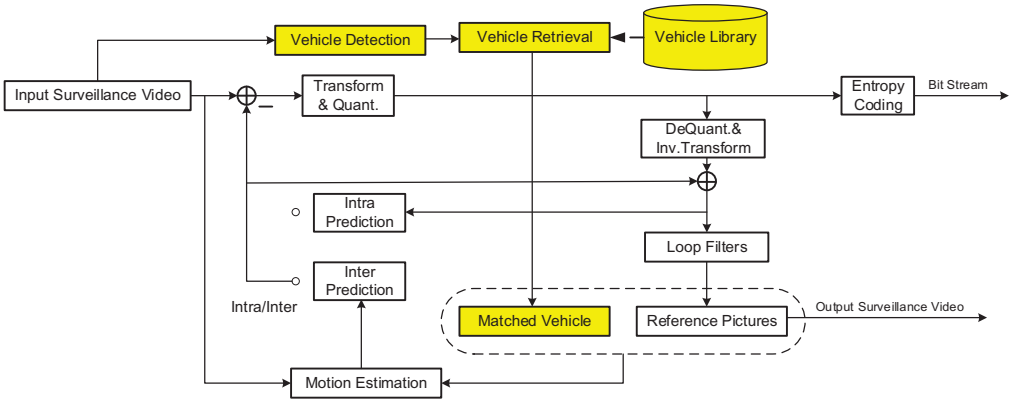

Fig. 2: Flowchart of our proposed vehicle library based surveillance video coding framework. The yellow blocks highlight the new modules in our framework compared with traditional video coding schemes (like HEVC).

dimension (i.e. across different cameras) but rather in temporal dimension (i.e. within the same camera). As shown in Fig. 1, there are always similar vehicles moving through one static camera, since the amount of vehicle models is limited. In order to remove the temporal global redundancy, we propose to build a 2D vehicle library for single camera surveillance video coding. Compared with [7], our proposed vehicle library based method has three advantages. First, the 2D vehicle pictures can be obtained directly from videos without the need of 3D models. Second, the 2D vehicle library consists of both sketch and texture information of vehicle. Third, the geometrical and illumination changes in single camera are much less than those across multiple cameras and thus easier to handle.

We have implemented a surveillance video coding scheme with vehicle library. The vehicle library is built offline by extracting the vehicles from the former part of the surveillance video shot by a single camera. Then during encoding, for each vehicle we retrieve similar vehicles from the library based on SIFT features [8], the retrieved vehicle is then used in inter prediction. Preliminary experimental results show that our proposed method achieves as high as 10.1% bit-rate saving for surveillance video coding, which demonstrates the potential of our proposed scheme.

The remainder of this paper is organized as follows. In Section 2, the details of vehicle library based video coding are presented. Experimental results and corresponding analyses are presented in Section 3, and Section 4 concludes this paper.

2. VEHICLE LIBRARY BASED VIDEO CODING

The flowchart of our proposed vehicle library based surveillance vehicle coding scheme is depicted in Fig. 2, where a 2D vehicle library is utilized at both encoder and decoder. In this paper, the vehicle library is built beforehand by extracting the vehicles from the former part of the surveillance video. Specifically, we adopt the SuBSENSE object detection method [9] to identify vehicles from every frame of the uncompressed video. As the vehicle shape can be irregular, we crop the bounding box of vehicle as rectangular picture and put it into the library. In the following subsections, we will discuss the details of the other parts of our proposed scheme.

2.1. Vehicle Retrieval

For the current frame to code, we first use the SuBSENSE method to segment vehicle out from background. After that, current vehicle retrieves similar vehicles from the library based on SIFT features [8]. Note that we crop the bounding box of vehicle in both the current frame and the vehicle library for simplicity. Therefore, we need to exclude the background pixels from the bounding boxes to make the vehicle matching more accurate.

Since the matching is based on SIFT features, to remove background SIFT features in the vehicle bounding boxes, we also perform SIFT extraction from a clean background picture which is built from the former part of the surveillance video. The SIFT features are extracted from both vehicle bounding box and collocated background area, and the matched background SIFT features between them are removed from the SIFT features of vehicle bounding box. Specifically, for each SIFT feature in the vehicle bounding box, it retrieves SIFT features from background area that is the neighborhood of current SIFT feature,

$$(xs_c - xs_b)^2 + (ys_c - ys_b)^2 \leq d^2 \tag{1}$$

where xs_c and ys_c are the coordinates of current SIFT feature in vehicle bounding box, xs_b and ys_b are the coordinates of its one neighboring SIFT feature in background area. d is the distance threshold, and is set to 5 in our implementation. If the minimum difference between current SIFT feature and its neighboring SIFT features satisfies (2), we regard the current SIFT feature as a background SIFT feature and it is removed out from vehicle bounding box,

$$D_{\min} \leq D_1 \tag{2}$$

(a) (b)

Fig. 3: Example results showing (a) the detected SIFT keypoints and (b) the remaining keypoints after removing background keypoints. Each keypoint is represented by a red dot.

Fig. 4: Example vehicle pictures in the built vehicle library.

where D_1 is the difference threshold and set to 1.1 in our implementation. It is worth noting that before computing the SIFT difference, all SIFT features are normalized. The example results of removing background SIFT features are shown in Fig. 3. It can be observed that the remaining SIFT features are almost all around the vehicle, as expected.

After removing the background SIFT features, we perform vehicle matching based on the remaining vehicle SIFT features. For current vehicle, it retrieves the most similar one vehicle from vehicle library based on the comparison of matched number of SIFT features. For one SIFT feature in current vehicle, we compute the differences between it and all the SIFT features in one vehicle that is in the vehicle library, if the differences satisfy the conditions (3) and (4), we regard that the current SIFT feature has found its matched SIFT feature.

$$d_1 \leq D_2 \tag{3}$$

$$d_1/d_2 \leq \alpha \tag{4}$$

where d_1 and d_2 are the minimum and second minimum distances, separately. D_2 is the threshold set to 0.4 and α is a constant set to 0.8. If multiple SIFT features in current vehicle match to the same SIFT feature in the in-library vehicle, the matched number is only added once. After computing all the matched SIFT numbers between current vehicle and the vehicles in library, we choose the in-library vehicle that has the most matched SIFT number as the final retrieved vehicle.

2.2. Using the Retrieved Vehicle for Inter Prediction

After retrieving the matched vehicle from the vehicle library for current vehicle, we utilize the matched vehicle picture for inter prediction. To this end, we put the matched vehicle picture onto a blank frame whose size is the same as normal reference frames. We also need to decide the position of the matched vehicle picture in the blank frame, as

$$x_0 = x_c + \frac{1}{n}\sum_{i=1}^{n}(xc_i - xv_i) \tag{5}$$

$$y_0 = y_c + \frac{1}{n}\sum_{i=1}^{n}(yc_i - yv_i) \tag{6}$$

where x_0 and y_0 are the positions to put the retrieved vehicle, x_c and y_c are the locations where the current vehicle is extracted. n is the number of matched SIFT keypoints between current vehicle and the matched vehicle, xc_i and yc_i are the positions of matched SIFT keypoint in current vehicle, xv_i and yv_i are the positions of the corresponding SIFT keypoint in matched vehicle. Then, the blank frame pasted with the matched vehicle is used together with normal reference frames for inter prediction.

Due to the matched vehicle used in inter prediction, some syntax elements about the matched vehicle need to be written into the bitstream. First, the number of vehicles that are extracted from current picture is encoded. Then, for each matched vehicle, its index in the vehicle library and its position to put onto the blank frame are encoded. These are frame-level syntax elements and are all encoded with fixed length coding. In our implementation, the bit length for vehicle number is 2, for vehicle index is 11, the bit lengths for vehicle position are 12 in both horizontal and vertical coordinates. Moreover, if the matched vehicle is chosen for inter prediction, the prediction unit (PU) encodes the corresponding index of the current vehicle within the current frame, the index is coded similar as reference frame index.

3. EXPERIMENTAL RESULTS

The proposed vehicle library based scheme is implemented into HM 16.7 software[1], and compared with HM anchor. Low delay B (LDB) configuration and QPs 27, 32, 37, 42 are used in experiments, and all the other parameters are set according to common test conditions. Especially, the number of short-term references is set to 4 in both our scheme and the anchor. A one-hour surveillance video, which is shot at Huangshan Road, Hefei, China, is adopted for test. The vehicle library is built by extracting vehicles from the former ten minutes

[1] https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-16.7/

Table 1: BD-rate of our scheme compared with HEVC, vehicle area only or entire sequence

Sequence	Vehicle Area	Entire Sequence
HSRClipA	−12.9%	−10.1%
HSRClipB	−11.5%	−3.6%
HSRClipC	−12.3%	−3.9%
HSRClipD	−4.5%	−0.9%
HSRClipE	−7.3%	−1.6%
HSRClipF	−7.0%	−1.1%
Average	**−9.2%**	**−3.5%**

Fig. 5: Example frames of the test sequences in our experiments.

of this video. Six test sequences are cropped from the remaining fifty minutes of the video. The video resolution is 1920×1080, and test sequences each lasts for 4 seconds. In Figs. 4 and 5, we show some example vehicles in our library and some frames of the test sequences. From these figures, it can be observed that the test sequences include similar vehicles but taken at different timestamps with various poses, which reflects the illumination and geometrical changes.

In order to evaluate our scheme, we compute BD-rate not only for entire sequence but also for vehicle area only. It is worth noting that the foreground objects are more meaningful than background in surveillance applications, therefore the quality of foreground objects is more important. Specifically, we calculate the mean-squared-error (MSE) of the vehicle area, which is detected by the SuBSENSE method, by including only the foreground pixels, and then calculate the corresponding PSNR. For bit rate, since we cannot obtain the accurate number of bits of only foreground, we calculate at the CTU level, if more than half of the area of a CTU is foreground, the CTU is included in counting the number of foreground bits.

The BD-rate reduction for vehicle area and entire sequence is shown in Table 1. It can be observed that our proposed vehicle library based method achieves as high as 12.9% BD-rate reduction for vehicle area, and 10.1% for entire sequence, respectively. Since our method is aimed at improving the vehicle coding efficiency, the BD-rate reduction for vehicle area is more important. Comparing Fig. 5 and Table 1, it can be observed that our proposed method achieves consistent improvement, even there are illumination and geometrical changes in the test sequences. The BD-rate reduction for entire sequence is partially determined by the relative area of vehicles in the video.

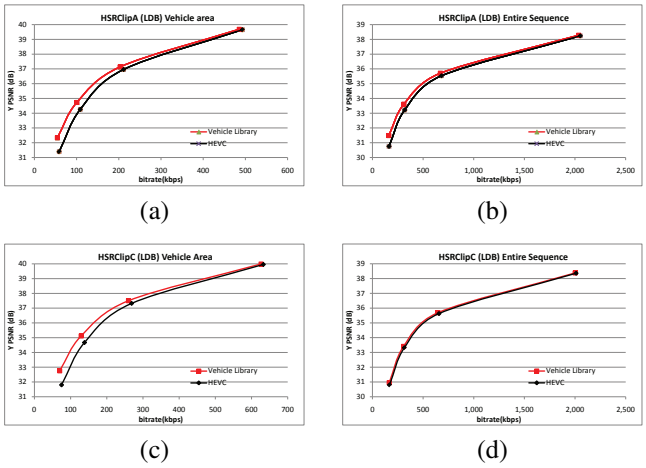

Fig. 6: RD curves of the sequences *HSRClipA* and *HSRClipC*: (a) (c) the vehicle area only, (b) (d) the entire sequence.

To further analyze the vehicle coding efficiency at different bit rates, the RD curves of sequences *HSRClipA* and *HSRClipC* are depicted in Fig. 6. It can be observed that at lower bit rates, the coding gain is higher. This is because the lower the bit rate is, the lower the vehicle reconstruction quality is in short-term reference pictures, and thus the higher probability that current vehicle uses the matched vehicle in library, which has high quality, for inter prediction. As a result, our proposed vehicle library based method is more suitable for low bit rate scenarios.

4. CONCLUSION

In this paper, we propose a vehicle library based surveillance video coding scheme, which is aimed at removing the temporal global redundancy among vehicles in single surveillance video. Preliminary experimental results demonstrate that our proposed method achieves as high as 10.1% BD-rate reduction, which verifies the proposed method. Compared with the 3D vehicle model library that is designed for multi-source surveillance video, our proposed method is easier to implement and promising for real applications.

In the future, we plan to improve our method in the following two aspects. First, we will investigate the vehicle retrieval method. Second, we will investigate geometrical transformation and illumination compensation methods to make our method more robust to the changes of vehicle.

5. REFERENCES

[1] Thomas Wiegand, Gary J Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[2] Xianguo Zhang, Yonghong Tian, Tiejun Huang, Siwei Dong, and Wen Gao, "Optimizing the hierarchical prediction and coding in HEVC for surveillance and conference videos with background modeling," *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4511–4526, 2014.

[3] William J Welsh, "Model-based coding of moving images at very low bit rates," in *Picture Coding Symposium (PCS)*, 1987.

[4] Divya Venkatraman and Anamitra Makur, "A compressive sensing approach to object-based surveillance video coding," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2009, pp. 3513–3516.

[5] Xianguo Zhang, Tiejun Huang, Yonghong Tian, and Wen Gao, "Background-modeling-based adaptive prediction for surveillance video coding," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 769–784, 2014.

[6] Jing Xiao, Yu Chen, Liang Liao, Jinhui Hu, and Ruimin Hu, "Global coding of multi-source surveillance video data," in *Data Compression Conference (DCC)*. IEEE, 2015, pp. 33–42.

[7] Jing Xiao, Ruimin Hu, Liang Liao, Yu Chen, Zhongyuan Wang, and Zixiang Xiong, "Knowledge-based coding of objects for multisource surveillance video data," *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp. 1691–1706, 2016.

[8] David G Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[9] Pierre-Luc St-Charles, Guillaume-Alexandre Bilodeau, and Robert Bergevin, "Subsense: A universal change detection method with local adaptive sensitivity," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 359–373, 2015.