INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC 1/SC 29/WG 2
MPEG TECHNICAL REQUIREMENTS

ISO/IEC JTC 1/SC 29/WG 2 M56679
Online – April 2021

Title: **[VCM] Partial transmission of SIFT features with compressed video**

Source: **WG 2 MPEG Technical requirements**
Author(s): **Sławomir Maćkowiak, Marek Domański, Dominik Cywiński, Jakub Szekiełda**
Poznań University of Technology, Poznań, Poland

Status: **Input**

Abstract

In Video Coding for Machines, parallel transmission of features and video will be likely a frequent application scenario. Unfortunately, such transmission is inevitably related to redundancy, as at least part of features may be extracted from the decoded video at the receiver. On the other side, the document M56678 demonstrates that strong compression yields losses of many features. In document M56678, SIFT keypoints were considered. Here, we are looking for the information about data that cannot be retrieved from the decoded video and needs to be transmitted as side information.

The research results of this paper are provided in the context of SIFT features and video compression using HEVC and VVC codecs.

1. Introduction

In Video Coding for Machines, high-fidelity extraction of visual features from decoded video is of paramount importance. Unfortunately, strong compression yields substantial losses and deterioration of the features that can be extracted from compressed video. In document M56678 [11], SIFT keypoints were considered. It is demonstrated that low-bitrate coding strongly reduces the count of SIFT keypoints that can be extracted from video at the receiver. Moreover, the parameters of the preserved SIFT keypoints are often modified, fortunately mostly be moderate errors.

Here, in this contribution, we study hybrid transmission of lossy-compressed video and some features (treated possibly by lossless coding). We continue to study the problem in the context of SIFT keypoints.

A possibility is to eliminate data from the feature stream if that data can be reconstructed from features retrieved from the video stream. Therefore, only data corresponding to those keypoints that cannot be retrieved from the decoded video, needs to be transmitted as side information to the video stream. The full set of keypoints can be reconstructed in the receiver, based on the keypoints reconstructed at the decoder side from the video stream and the corresponding corrections sent to the decoder by the feature stream. The goal of tis contribution is to asses the abovementioned possibility. Therefore, for the decoded video, we provide the results of an experiment aimed at estimation of the sets of keypoints in the relevant categories:

- keypoints with the same location as in original video (maximum shift by one sampling period either horizontally or vertically) – *"same",*
- keypoints moved by limited number of sampling periods (+- 3) – *"moved",*
- keypoints lost due to compression – *"missed",*
- new keypoints extracted from decoded video that were absent in the original uncompressed video – *"new".*

The SIFT feature detector [3] and HEVC and VVC video codecs [9,10] were used in this study. The PoznańCarpark and PoznańStrret HD sequences were used as a test material.

SIFT feature ~~Detector~~ detector/extractor ~~SIFT~~ used: Python + OpenCV versions 4.3.0.

2. Video codec configuration

The parameters of the **HEVC encoder** are as follows:
HM software: Encoder Version [16.20] (including RExt)[Linux][GCC 9.2.1][64 bit]

Real Format	: 1920x1088 25Hz
Internal Format	: 1920x1088 25Hz
Profile	: main
CU size / depth / total-depth	: 64 / 4 / 4
RQT trans. size (min / max)	: 4 / 32
Max RQT depth inter	: 3
Max RQT depth intra	: 3
Min PCM size	: 8
Motion search range	: 384
Intra period	: 32
Decoding refresh type	: 1
QP	: from 17 to 47
GOP size	: 16
Input bit depth	: (Y:8, C:8)
MSB-extended bit depth	: (Y:8, C:8)
Internal bit depth	: (Y:8, C:8)
PCM sample bit depth	: (Y:8, C:8)
Intra reference smoothing	: Enabled
Input ChromaFormatIDC	= 4:2:0
Output (internal) ChromaFormatIDC	= 4:2:0

The following encoder tool parameters were set:
TOOL CFG: IBD:0 HAD:1 RDQ:1 RDQTS:1 RDpenalty:0 LQP:0 SQP:0 ASR:1 MinSearchWindow:96 RestrictMESampling:0 FEN:1 ECU:0 FDM:1 CFM:0 ESD:0 RQT:1 TransformSkip:1 TransformSkipFast:1 TransformSkipLog2MaxSize:2 Slice: M=0 SliceSegment: M=0 CIP:0 SAO:1 PCM:0 TransQuantBypassEnabled:0 WPP:0 WPB:0 PME:2 WaveFrontSynchro:0 WaveFrontSubstreams:1 ScalingList:0 TMVPMode:1 AQpS:0 SignBitHidingFlag:1 RecalQP:0

The parameters of the **VVC encoder** are as follows:
VVCSoftware: VTM Encoder Version 11.0 [Linux][GCC 9.3.0][64 bit] [SIMD=AVX2]

Real Format	: 1920x1088 25Hz
Internal Format	: 1920x1088 25Hz
Profile	: main_10
CTU size / min CU size	: 128 / 4
Motion search range	: 384
Intra period	: 32
Decoding refresh type	: 1
DRAP period	: 0
QP	: from 17 to 47
GOP size	: 32
Input bit depth	: (Y:8, C:8)
MSB-extended bit depth	: (Y:8, C:8)
Internal bit depth	: (Y:10, C:10)
Intra reference smoothing	: Enabled
Input ChromaFormatIDC	= 4:2:0
Output (internal) ChromaFormatIDC	= 4:2:0

The following encoder tool parameters were set:
TOOL CFG: IBD:1 HAD:1 RDQ:1 RDQTS:1 RDpenalty:0 LQP:0 SQP:0 ASR:1 MinSearchWindow:96 RestrictMESampling:0 FEN:1 ECU:0 FDM:1 ESD:0 TransformSkip:1 TransformSkipFast:1 TransformSkipLog2MaxSize:5 ChromaTS:1 BDPCM:0 Tiles: 1x1 Slices: 1 MCTS:0 SAO:1 ALF:1 CCALF:1 WPP:0 WPB:0 PME:2 WaveFrontSynchro:0 WaveFrontSubstreams:1 ScalingList:0 TMVPMode:1 DQ:1 SignBitHidingFlag:0 RecalQP:0
TOOL CFG: LFNST:1 MMVD:1 Affine:1 AffineType:1 PROF:1 SbTMVP:1 DualITree:1 IMV:1 BIO:1 LMChroma:1 HorCollocatedChroma:1 VerCollocatedChroma:0 MTS: 1(intra) 0(inter) SBT:1 ISP:1 SMVD:1 CompositeLTReference:0 Bcw:1 BcwFast:1 LADF:0 CIIP:1 Geo:1 AllowDisFracMMVD:1 AffineAmvr:1 AffineAmvrEncOpt:1 DMVR:1 MmvdDisNum:6 JointCbCr:1 ACT:0 PLT:0 IBC:0 HashME:0 WrapAround:0 VirtualBoundariesEnabledFlag:0 VirtualBoundariesPresentInSPSFlag:1 vertical virtual boundaries:[ ] horizontal virtual boundaries:[ ] Reshape:1 (Signal:SDR Opt:0 CSoffset:6) MRL:1 MIP:1 EncDbOpt:0
FAST TOOL CFG: LCTUFast:1 FastMrg:1 PBIntraFast:1 IMV4PelFast:1 MTSMaxCand: 4(intra) 4(inter) ISPFast:0 FastLFNST:0 AMaxBT:1 E0023FastEnc:1 ContentBasedFastQtbt:0 UseNonLinearAlfLuma:1 UseNonLinearAlfChroma:1 MaxNumAlfAlternativesChroma:8 FastMIP:0 FastLocalDualTree:1 NumSplitThreads:1 NumWppThreads:1+0 EnsureWppBitEqual:0 RPR:0 TemporalFilter:1

3. The course of the experiment

The purpose of the experiment was to determine the number of keypoints belonging to one of the defined categories. The categories are defined in the following section.

The frames of the PoznańStreet and PoznańCarpark sequences were encoded at 1920x1088 resolution using both HEVC and VVC encoders for QP=17, 22, 27, 32, 37, 42 and 47 quantization factors and then decoded. SIFT technique was used to determine the characteristic points. At the same time, the SIFT feature stream for the uncompressed sequence images was determined. It was ensured in the SIFT algorithm that all possible feature points would be determined. The number of layers in an octave was left the original equal to 3. We left the sigma parameter at the default value, i.e. 1.6.

The results were independently performed for each sequence, encoder and QP / bitrate parameter. The results for the image were averaged after 250 frames of the sequence.

The number of keypoints are defined in 4 categories:

- *same* - keypoints consistent in position between the keypoints of the decoded image and the keypoints of the original image. A point is the same if its position does not exceed one sampling point in one direction. If the position changes by one point in both directions in the sampling grid then the point is not the same.
- *moved* - keypoints shifted in the decoded image but within the boundary of the 3 x 3 sampling window around the location of the keypoints in the original image,
- *missed* - keypoints lost in the decoded image but which were present in the original image, i.e. outside the range of the 3 x 3 window around the location of the keypoint in the original image,
- *new* - keypoints present in the decoded image that have no corresponding keypoint in the original image.

The number of keypoints in categories are presented in Figures 3.1-3.4. The sum in the graphs represents the summed scores of the key categories: 'same', 'moved' and 'new'. If we aggregated the number of 'same', 'moved' 'missed' keypoints, we would get the number of keypoints determined for the original image.

4. Experimental results

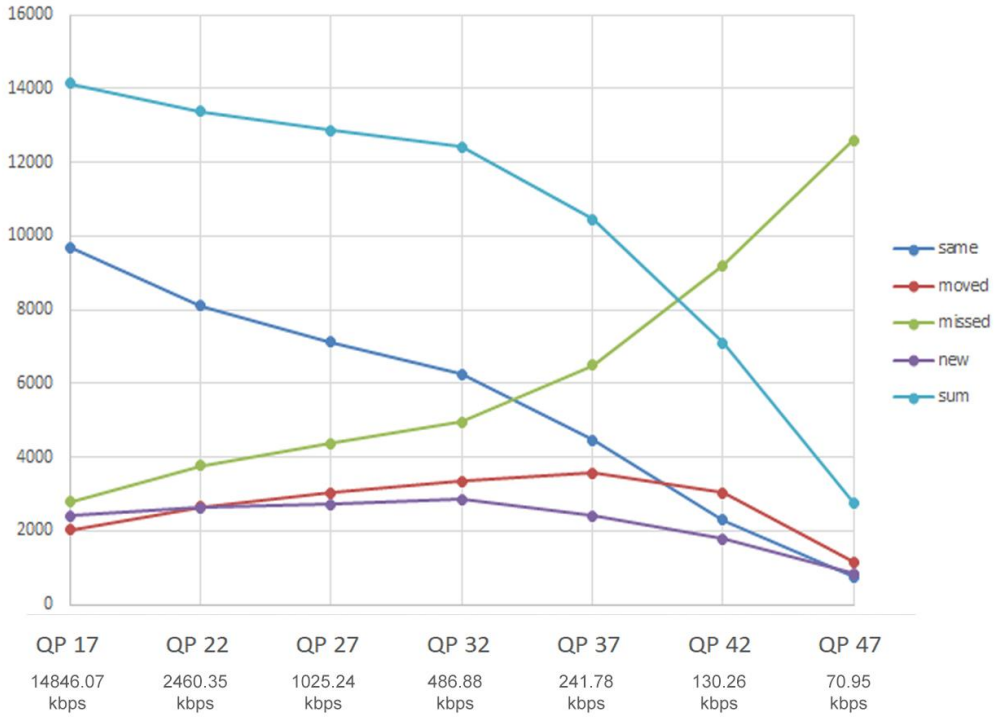

Fig. 4.1 Counts of keypoints in the categories (HEVC encoding, PoznańCarpark sequence).

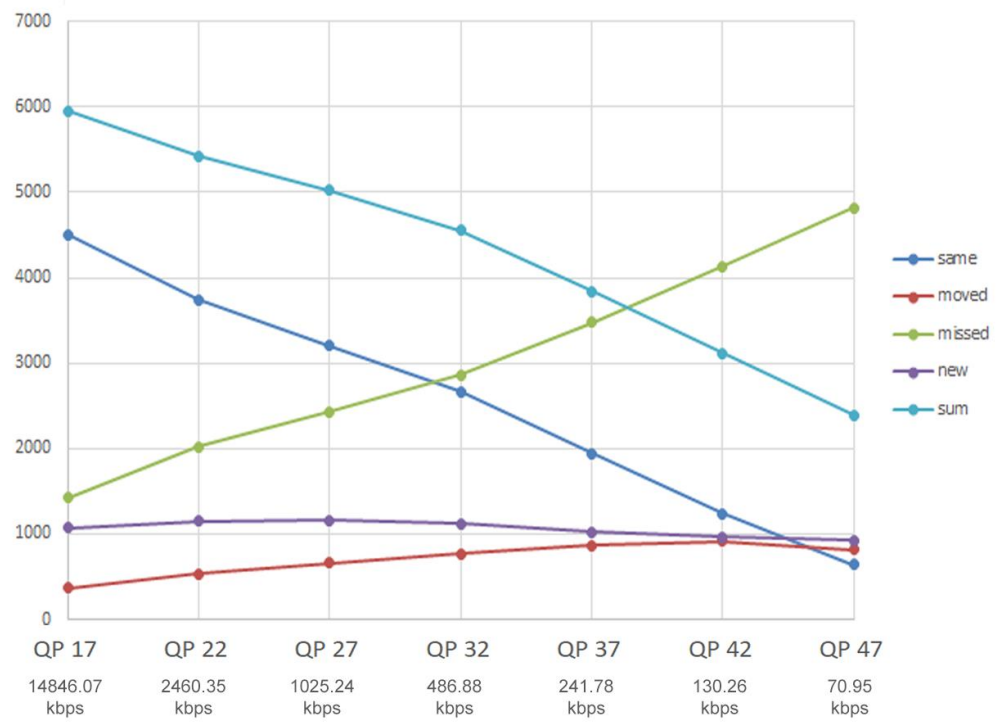

Fig. 4.2 Counts of keypoints in the categories (HEVC encoding, PoznańStreet seq~~.~~)uence).

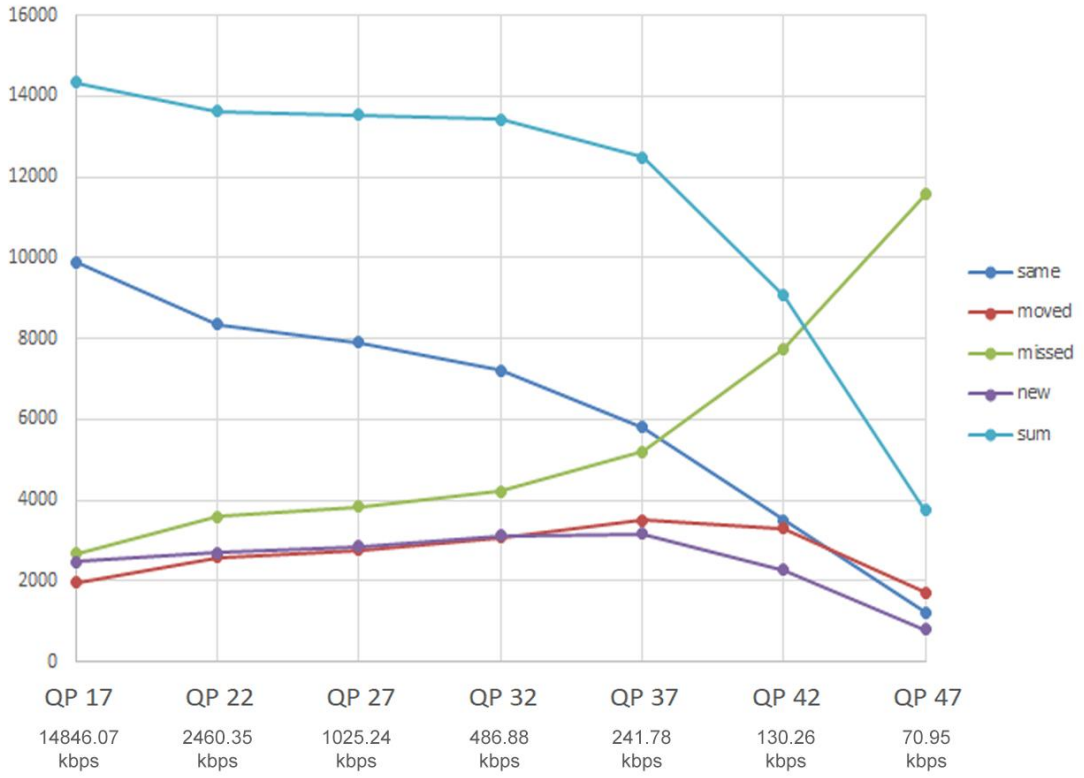

Fig. 4.3 Counts of keypoints in the categories (VVC encoding, PoznańCarpark sequence).

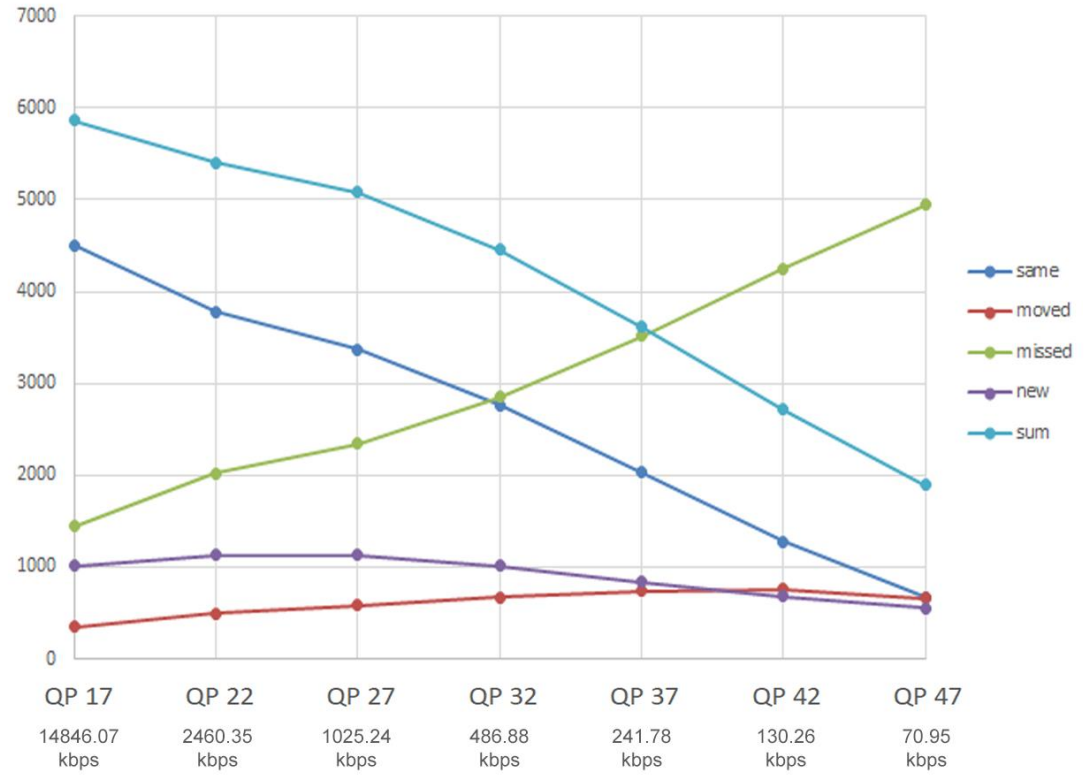

Fig. 4.4 Counts of keypoints in the categories (VVC encoding, PoznańStreet sequence).

Corrections have to be sent for keypoints whose positions are shifted, but still close to the position of the keypoint in the original image (category '*moved*'). In the case of keypoints outside the 3 x 3 points window ('*missed*' category), the entire set of keypoints from the original image in this category will have to be sent. Therefore, the counts in these categories should be paid attention to first.

The results are very comparable for HEVC and VVC compression techniques. The averaged number of keypoints per image in the '*moved*' category for the PoznanCarpark sequence ranges from 11.5 to 21 percent of the number of keypoints for the uncompressed image (QP value from 17 to 32, bitrate above 0.5Mbps). For the PoznanStreet sequence, this number is between 4.7 and 10.3 percent.

The number of keypoints in the '*missed*' category for the PoznanCarpark sequence ranges from 15.6 to 28 percent of the number of key points for an uncompressed image (QP value from 17 to 32, bitrate above 0.5Mbps). For the PoznanStreet sequence this number is between 19.3 and 39 percent.

5. Conclusions

In this contribution, we propose partial transmission of features together with compressed video. Such hybrid transmission is demonstrated for SIFT keypoints and HEVC/VVC-compressed video. Such a hybrid transmission of compressed video and a part of features appears as an interesting solution for prospective Video Coding for Machines.

6. Acknowledgment

This work was supported by Project 0314/050.

7. References

[1] "Use cases and requirements for Video Coding for Machines," Doc. ISO/IEC JTC1/SC29/WG11 N19506, June 2020.
[2] "Draft Evaluation Framework for Video Coding for Machines," Doc. ISO/IEC JTC1/SC29/WG11 N19507, June 2020.
[3] Lowe D. G., Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, 60(2), 2004, pp91-110.
[4] ISO/IEC Int. Standard 23008-2: 2015 "High efficiency coding and media delivery in heterogeneous environment – Part 2: High efficiency video coding" and ITU-T Rec. H.265 (V3) (2015), „High efficiency video coding".
[5] G. J. Sullivan, J. Ohm, W. J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard", in IEEE Transactions on Circuits Systems for Video Technology, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.
[6] M. Domański, T. Grajek, K. Klimaszewski, M. Kurc, O. Stankiewicz, J. Stankowski, K. Wegner, "Poznań multiview video test sequences and camera parameters", ISO/IEC JTC1/SC29/WG11 MPEG Doc. M17050, Xian, China, Oct. 2009.
[7] Rec. ITU-T H.265 | ISO/IEC 23008-2 High efficiency video coding.
[8] Rec. ITU-T H.266 | ISO/IEC 23090-3 Versatile video coding.
[9] https://vcgit.hhi.fraunhofer.de/jct-vc/HM/-/tree/HM-16.20.
[10] https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tree/VTM-11.0.
[11] S. Maćkowiak, M. Domański, J. Stankowski, D. Cywiński, J. Szekiełda, "Influence of HEVC and VVC coding on the SIFT characteristic points extracted from the received video," ISO/IEC JTC1/SC29/WG2 MPEG Doc. M56678, April 2021, Online.