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 M56678
Online – April 2021

Title: **[VCM] Influence of HEVC and VVC coding on the SIFT characteristic points extracted from the received video**
Source: **WG 2 MPEG Technical requirements**
Author(s): **Slawomir Maćkowiak, Marek Domański, Jakub Stankowski, Dominik Cywiński, Jakub Szekiełda**
Poznań University of Technology, Poznań, Poland
Status: **Input**

Abstract

The purpose of this contribution is to investigate the influence of HEVC and VVC compression on the number and parameters of the SIFT keypoints extracted from the decoded video. The experimental results provide the relation between quantization parameter and bitrate and the number and parameters of the keypoints retrieved from the decoded video. For the keypoints preserved in the decoded video, the modifications of the repective parameters are also provided as functions of quantization parameter and bitrate.

1. Introduction

Prospective technology of Video Coding for Machines can make more efficient use of the information that is currently sent to the decoder in both the encoded video stream and the feature stream. Assuming that some of the information is still redundant we look for ways to eliminate its dual transmission. We are looking at the feature stream and want to explore the properties of parameter changes for different video content and different encoders. Knowing the statistics of these parameters will allow us to propose mechanisms to reduce the duplicate information between the video stream and the feature stream which together will achieve higher efficiency in VCM scenarios. We use HEVC and VVC video encoders, HD video sequences, and SIFT features.

2. Video encoder 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 influence of HEVC and VVC compression on the ability to determine the SIFT keypoints in the decoded video

3.1 Influence of HEVC and VVC compression rates on the number of SIFT keypoints in decoded video

The aim of the experiments was to check how the HEVC and VVC video encoding techniques influence on the determination of SIFT keypoints in the image. 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 (Fig.2.1). 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 uncompressed frames of the PoznańStreet and PoznańCarpark sequences are used as a reference for comparison. Results were accumulated and averaged for 250 frames of sequence. The block diagram illustrating the experiment is presented on Fig. 3.1.

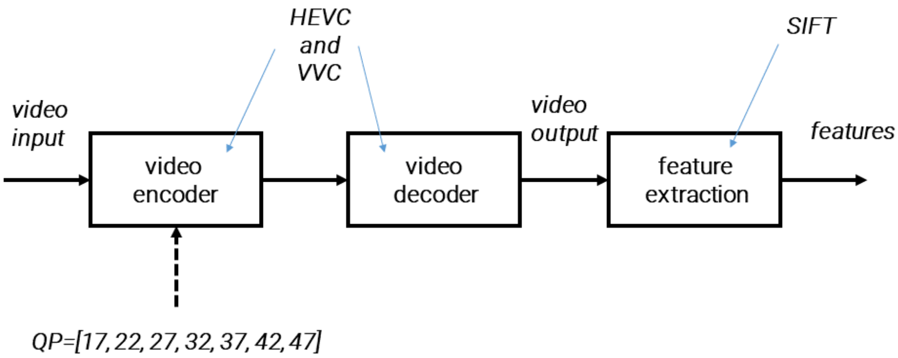

Fig. 3.1. Block diagram illustrating the experiment

Figures 3.2-3.5 depict the counts of SIFT keypoints versus the quantization parameter (QP) and bitrate. The counts of SIFT keypoints mean the counts of the keypoints extracted from the decoded video. The counts of the SIFT keypoints extracted from uncompressed sequences are also shown as horizontal lines on the top of the plots.

The SIFT features [3] were extracted using the SIFT feature detector/extractor from OpenCV version 4.3.0. and Python.

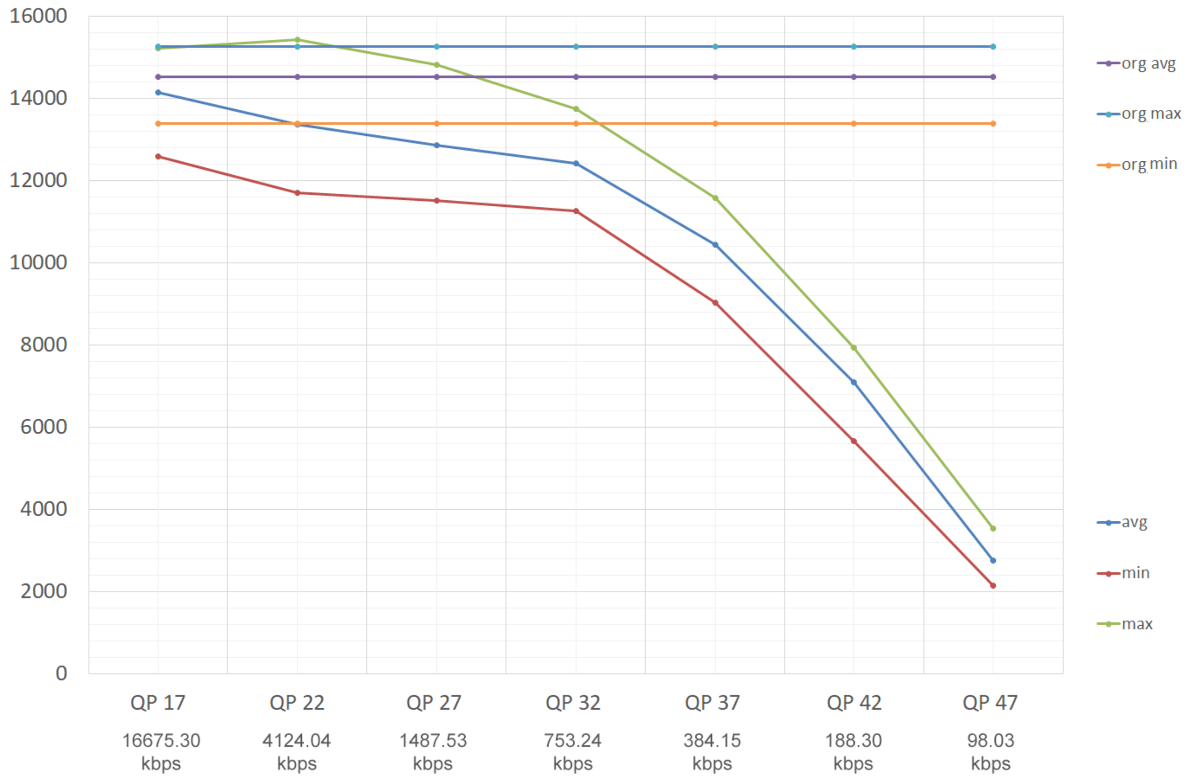

Fig. 3.2 The counts of SIFT keypoints extracted from the decoded (HEVC encoding, PoznańCarpark sequence). The values represent the minimum, the average and the maximum count of SIFT keypoints per frame.

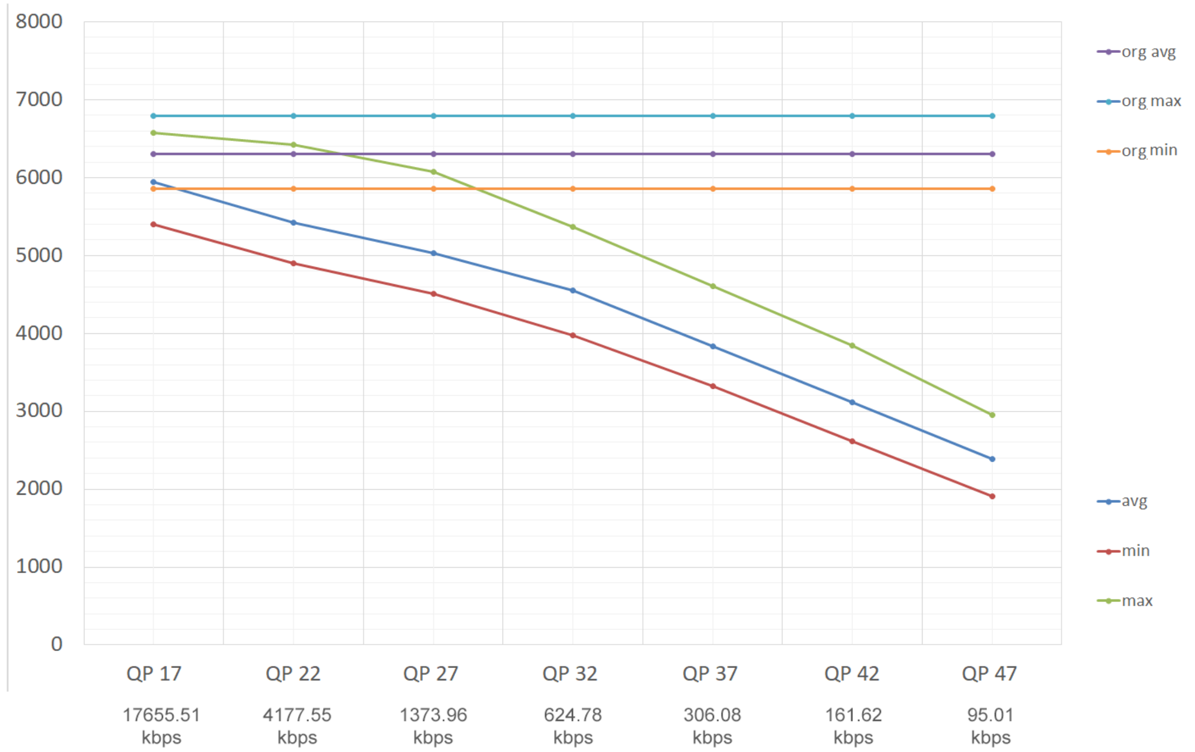

Fig. 3.3 The counts of SIFT keypoints extracted from the decoded (HEVC encoding, PoznańStreet sequence). The values represent the minimum, the average and the maximum count of SIFT keypoints per frame.

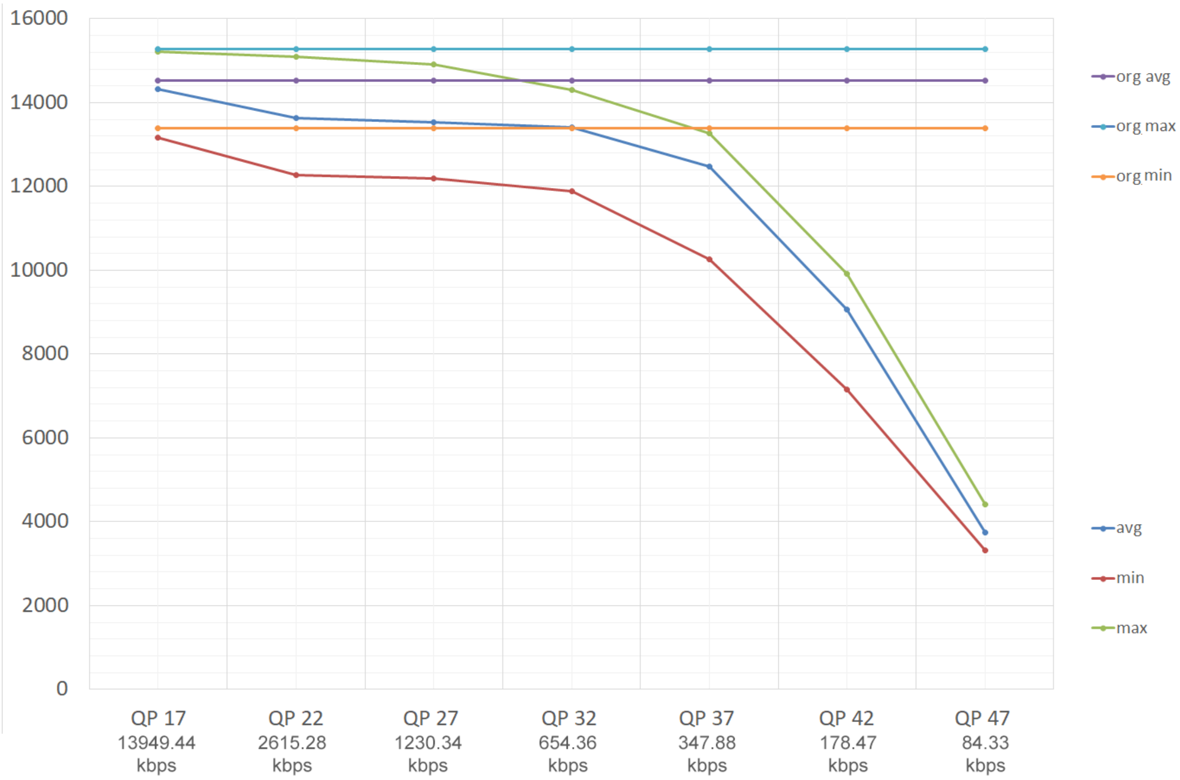

Fig. 3.4 The counts of SIFT keypoints extracted from the decoded (VVC encoding, PoznańCarpark sequence). The values represent the minimum, the average and the maximum count of SIFT keypoints per frame.

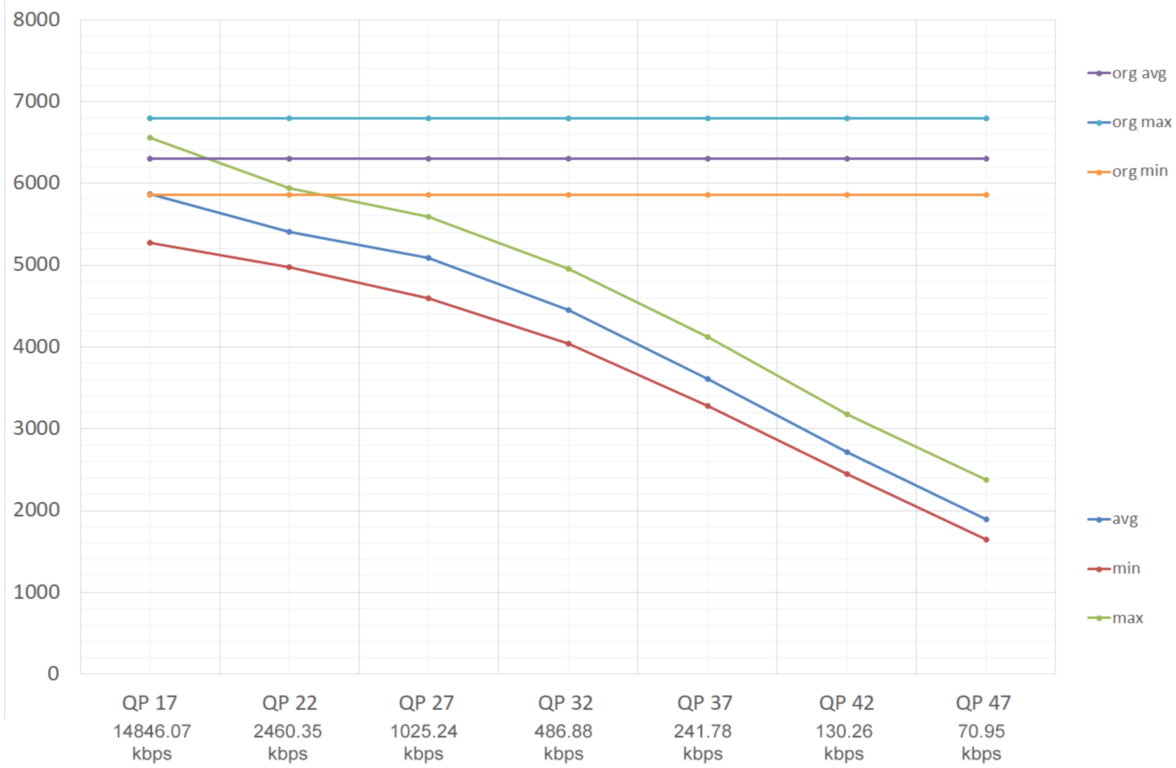

Fig. 3.5 Influence of the compression technique and quantization factor on the number of SIFT keypoints in the decoded image (VVC encoding, PoznańStreet sequence).

The general conclusion is as follows: as the QP increases, the number of feature points on the decoded image side decreases. However, from the graphs shown, we can see that roughly up to the value of QP=32, the number of points remains fairly constant and then rapidly decreases. The deviation of the minimum and maximum values of the number of points decreases towards the higher value of the quantization coefficient. This observation remains valid for both HEVC and VVC compression.

3.2 The count of different SIFT keypoints between the uncompressed image and the decoded image

Another study aimed to see how many different keypoints between the uncompressed image and the decoded image after compression are extracted. The distance in point position was used as a criterion for similarity. A point is similar 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. The block diagram illustrating the experiment is presented on Fig. 2.6.

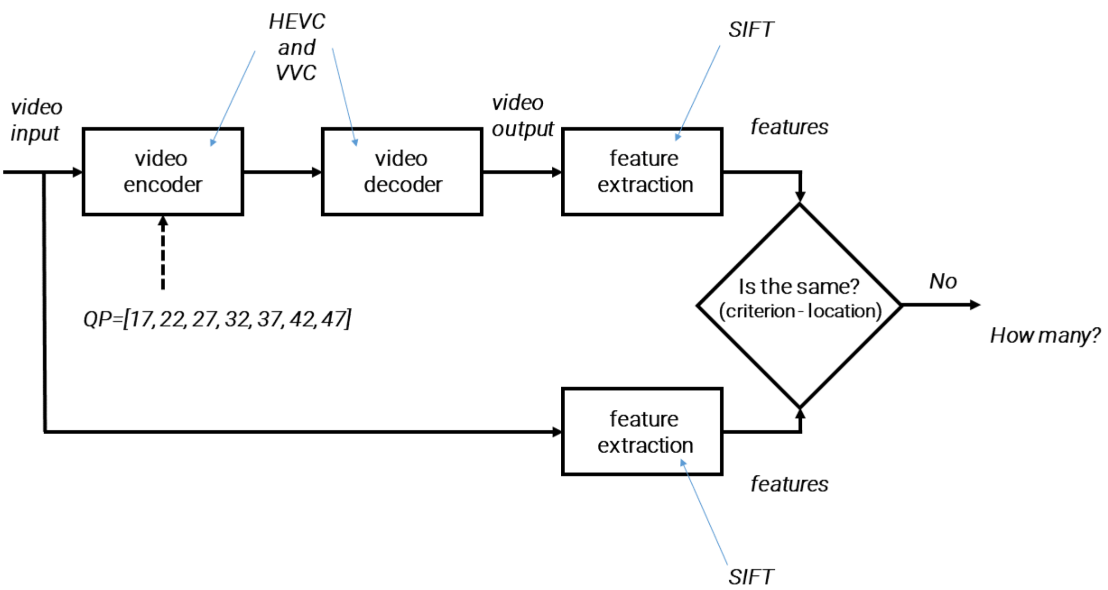

Fig. 3.6. Block diagram illustrating the experiment.

Figures 3.7 through 3.10 show the averaged statistics of the count of the original SIFT keypoints that are absent or different in the decoded video (in the sense of their locations). The counts are plotted separately for different picture types. The differences in the values of the keypoint features are not considered.

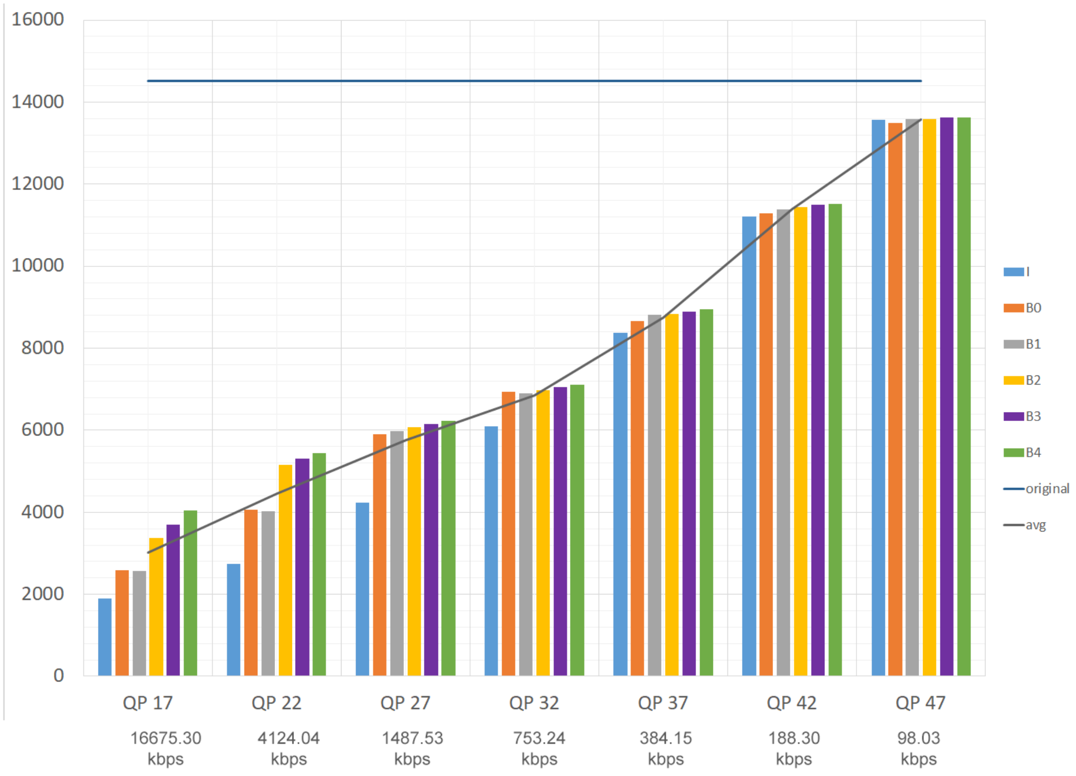

Fig. 3.7 The average count of SIFT keypoints that are not preserved in decoded video, i.e. that has vanished or are different in the sense of their location by coding type and quantization factor (HEVC encoding, PoznańCarpark sequence).

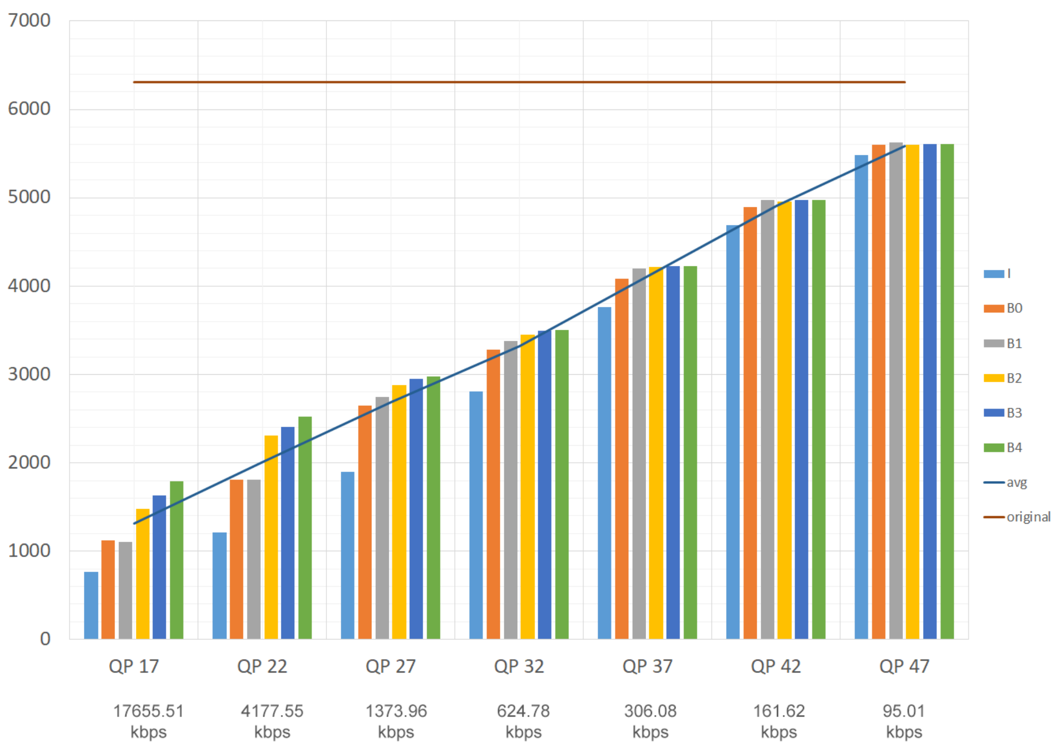

Fig. 3.8 The average count of SIFT keypoints that are not preserved in decoded video, i.e. that has vanished or are different in the sense of their location by coding type and quantization factor (HEVC encoding, PoznańStreet sequence).

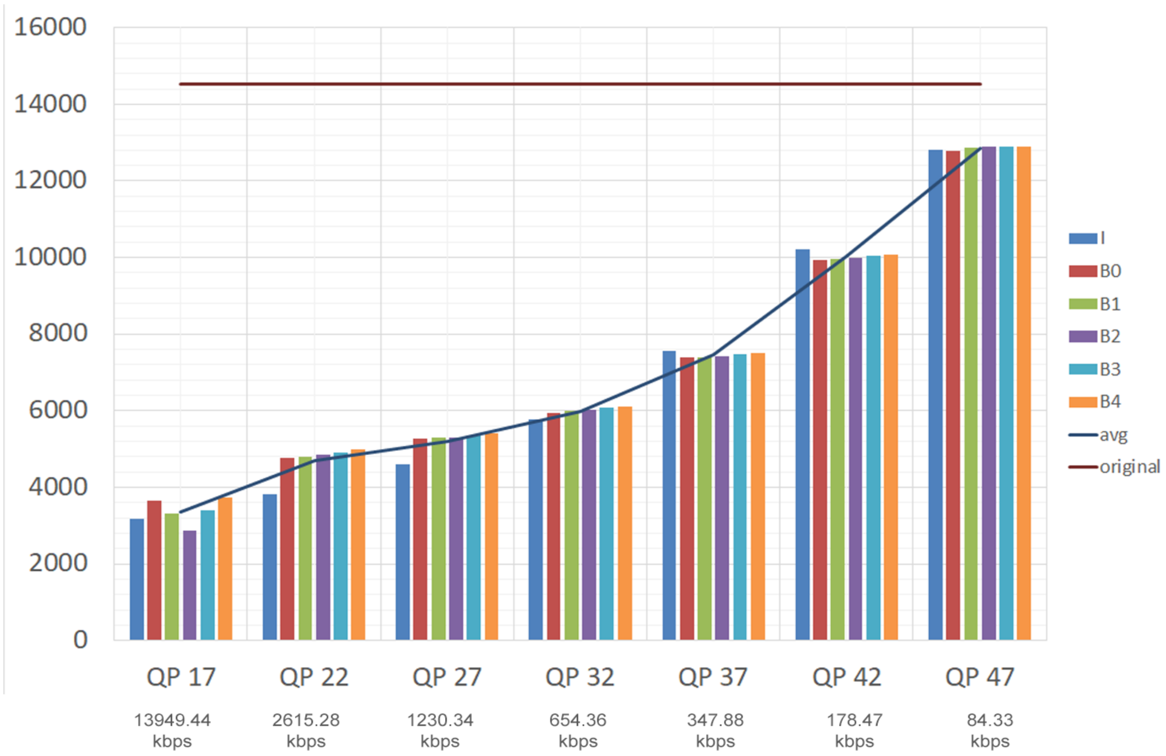

Fig. 3.9 The average count of SIFT keypoints that are not preserved in decoded video, i.e. that has vanished or are different in the sense of their location. by coding type and quantization factor (VVC encoding, PoznańCarpark sequence).

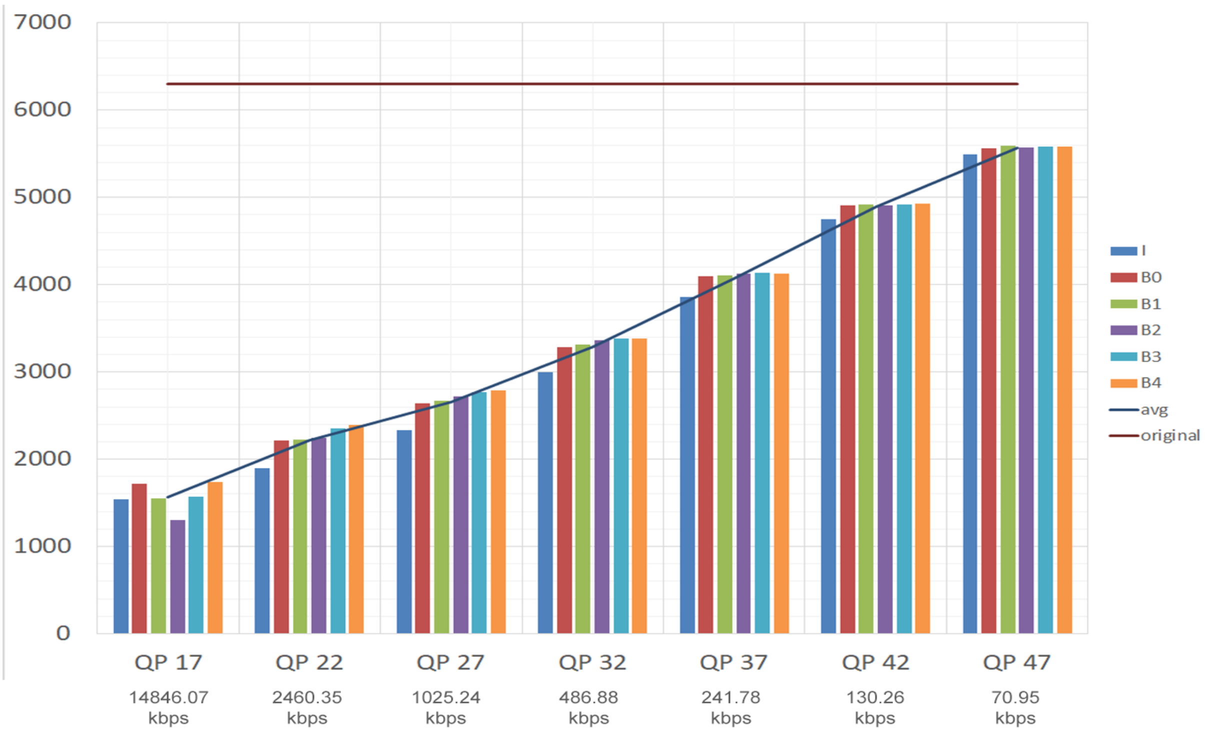

Fig. 3.10 The average count of SIFT keypoints that are not preserved in decoded video, i.e. that has vanished or are different in the sense of their location by coding type and quantization factor (VVC encoding, PoznańStreet sequence).

The analysis of the results shows that for quantization parameter values up to QP=32, there are significantly fewer different keypoints for intra-image coded (Intra) frames. This is independent of the image content and the codec type.

4. A study of the differences in SIFT keypoint parameters for compliant keypoint positions

This experiment has a goal to show the differences in SIFT keypoint parameters for the designated keypoints that are consistent between the original image and the decoded image after compression. In the experiment, we do not analyze the content of keypoint descriptors but only the parameters related to keypoint detection.

A SIFT keypoint is a circular image region with an orientation. It is described by a geometric frame of four parameters: the keypoint center coordinates *x* and *y*, its scale (the radius of the region), and its orientation (an angle expressed in radians). The SIFT detector uses as keypoints image structures which resemble “blobs”. By searching for blobs at multiple scales and positions, the SIFT detector is invariant (or, more accurately, covariant) to translation, rotations, and re scaling of the image.

For each keypoint with position *x*,*y* we will use from the SIFT algorithm the strength of the technique's response to the presence of a corner, and the dominant orientation based on the distribution of quantize gradients of the point directions (SIFT additionally performs Gaussian filtering to reduce the influence of gradients from the boundary of the region of interest). So we have three parameters ‘Response’, ‘Orientation’, and ‘Size’. The block diagram illustrating the experiment is presented on Fig. 3.1.

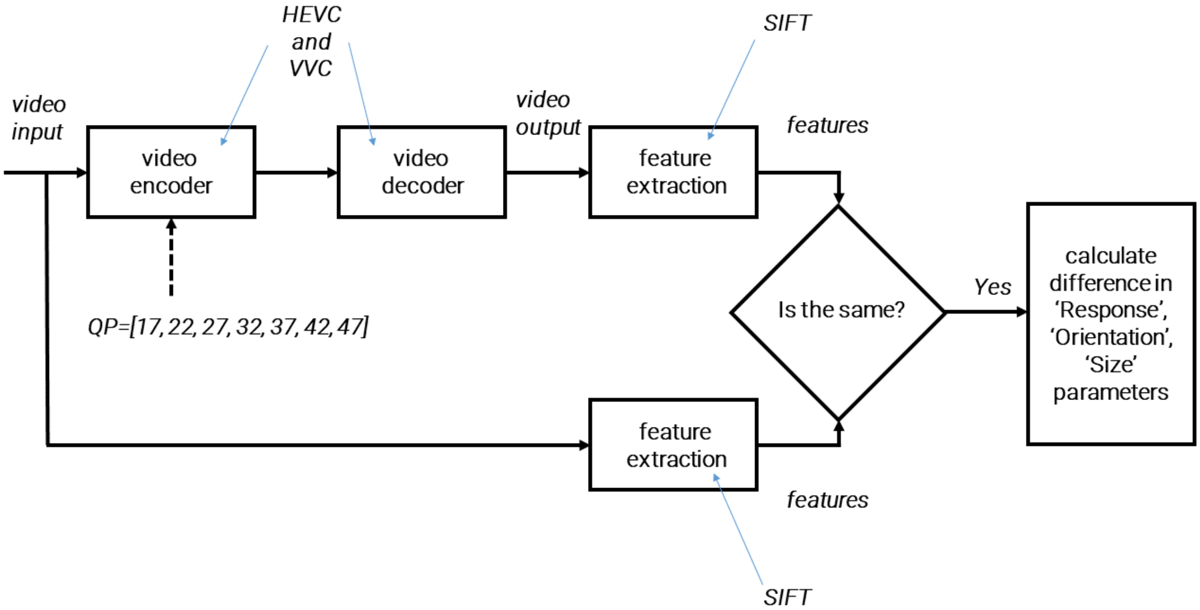

Fig. 4.1. Block diagram illustrating the experiment

Figures 4.2 through 4.13 show the cumulative distribution of errors the keypoint parameters ‘Response’, ‘Orientation’ and ‘Size’ for different QP / bitrate.

a) ‘Response’ parameter

The 'Response' parameter is a very important parameter because it determines the stability of the feature point with changes in image resolution. It is keypoint detector response on the keypoint (that is, strength of the keypoint), the response by which the most strong keypoints have been selected. This value is dependent on the value of the trace and the Hessian determinant. The Hessian matrix is the square matrix of the second partial derivatives of a function with real values twice differentiable at the location of the feature point.

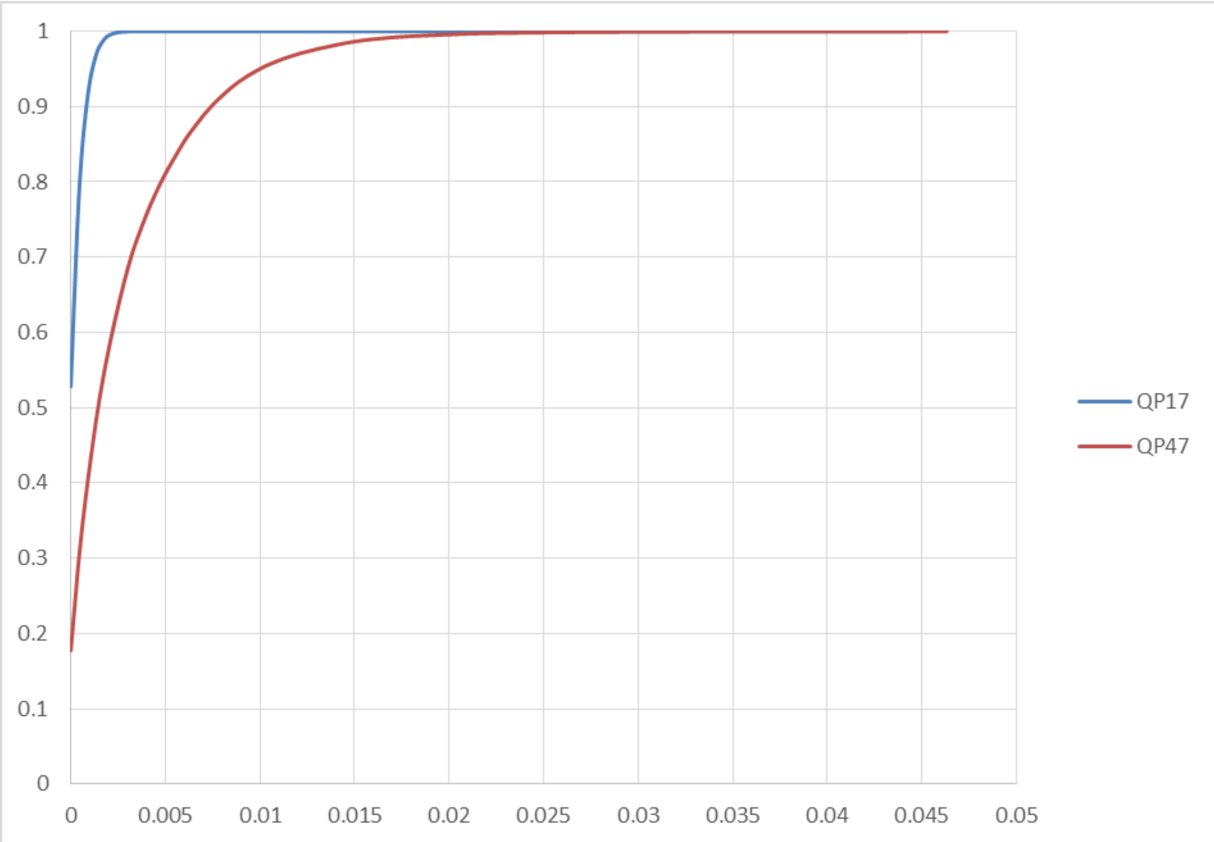

Fig. 4.2 Cumulative distribution of errors for the keypoint ‘Response’ parameter (HEVC encoding, PoznańCarpark sequence).

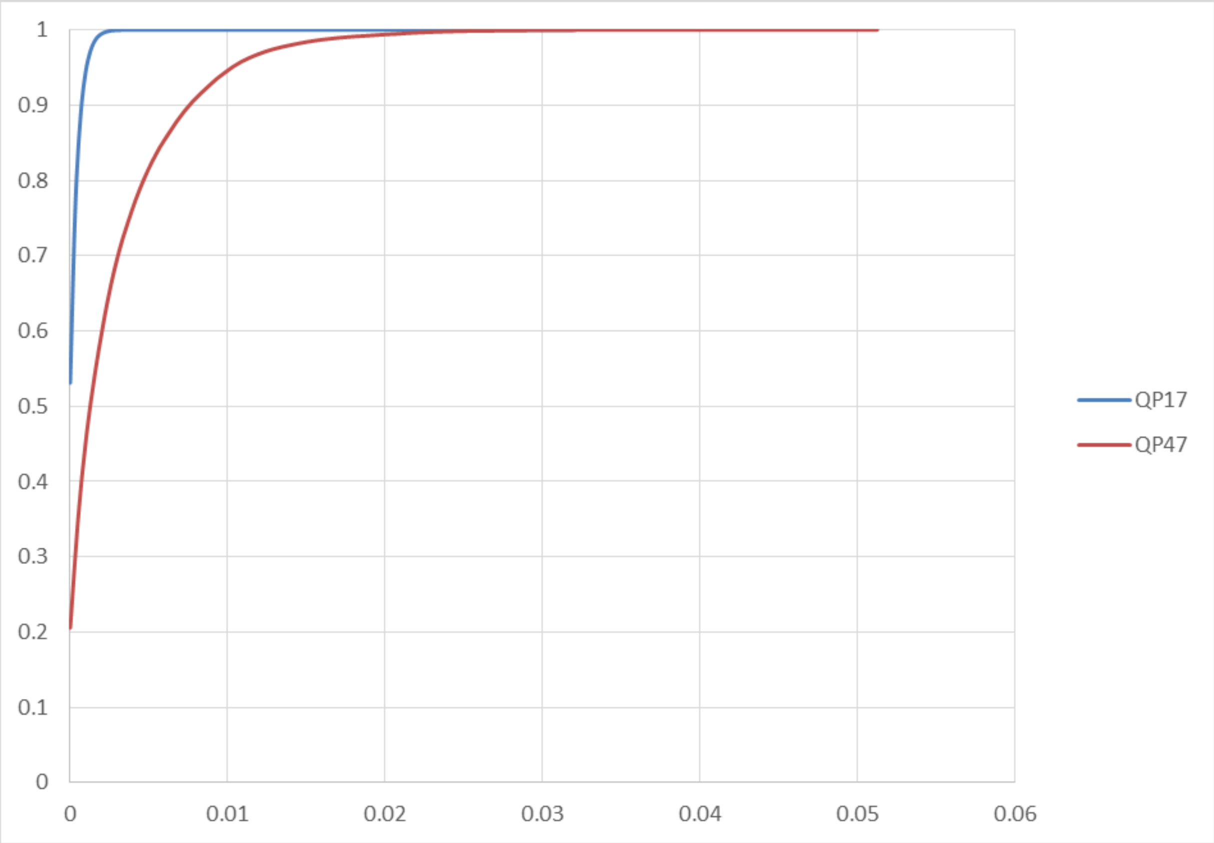

Fig. 4.3 Cumulative distribution of errors for the keypoint ‘Response’ parameter (HEVC encoding, PoznańStreet sequence).

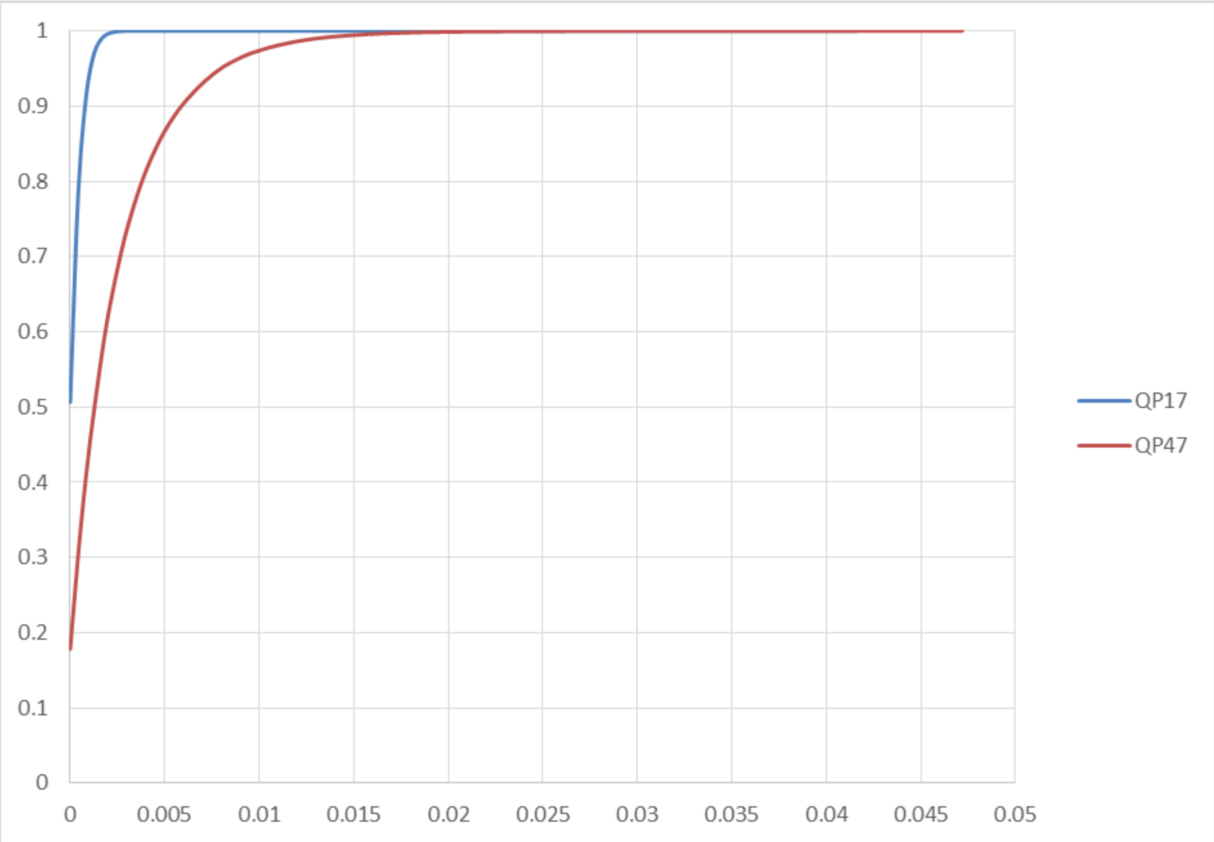

Fig. 4.4 Cumulative distribution of errors for the keypoint ‘Response’ parameter (VVC encoding, PoznańCarpark sequence).

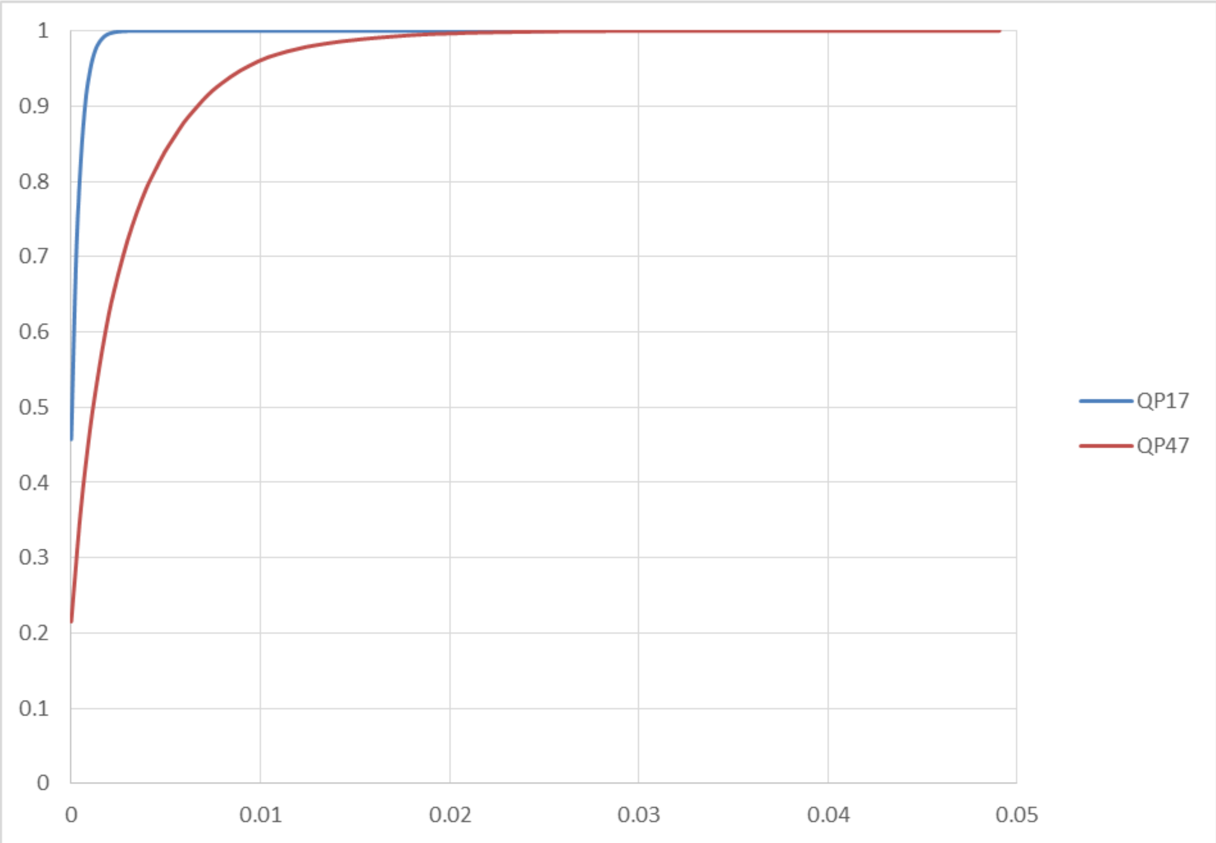

Fig. 4.5 Cumulative distribution of errors for the keypoint ‘Response’ parameter (VVC encoding, PoznańStreet sequence).

As can be seen from the results, the error distribution between the keypoint parameter determined for the original image and the keypoint parameter determined for the decoded image does not depend on the content and coding technique. Quantitatively, the error is dependent on the QP / bitrate parameter.

b) ‘Orientation’ parameter

The keypoint orientation is also determined from the local image appearance and is covariant to image rotations. Depending on the symmetry of the keypoint appearance, determining the orientation can be ambiguous. The orientation parameter is used to match keypoints more reliably, but from the point of view of comparing points in our experiment, could be a cause of possible single larger errors, since the SIFT algorithm can generate several instances of keypoints at the same position that differ in the orientation value. Therefore, keypoints for the original and decoded images whose orientations do not match within +/- 5 degrees are not treated as compliant keypoints.

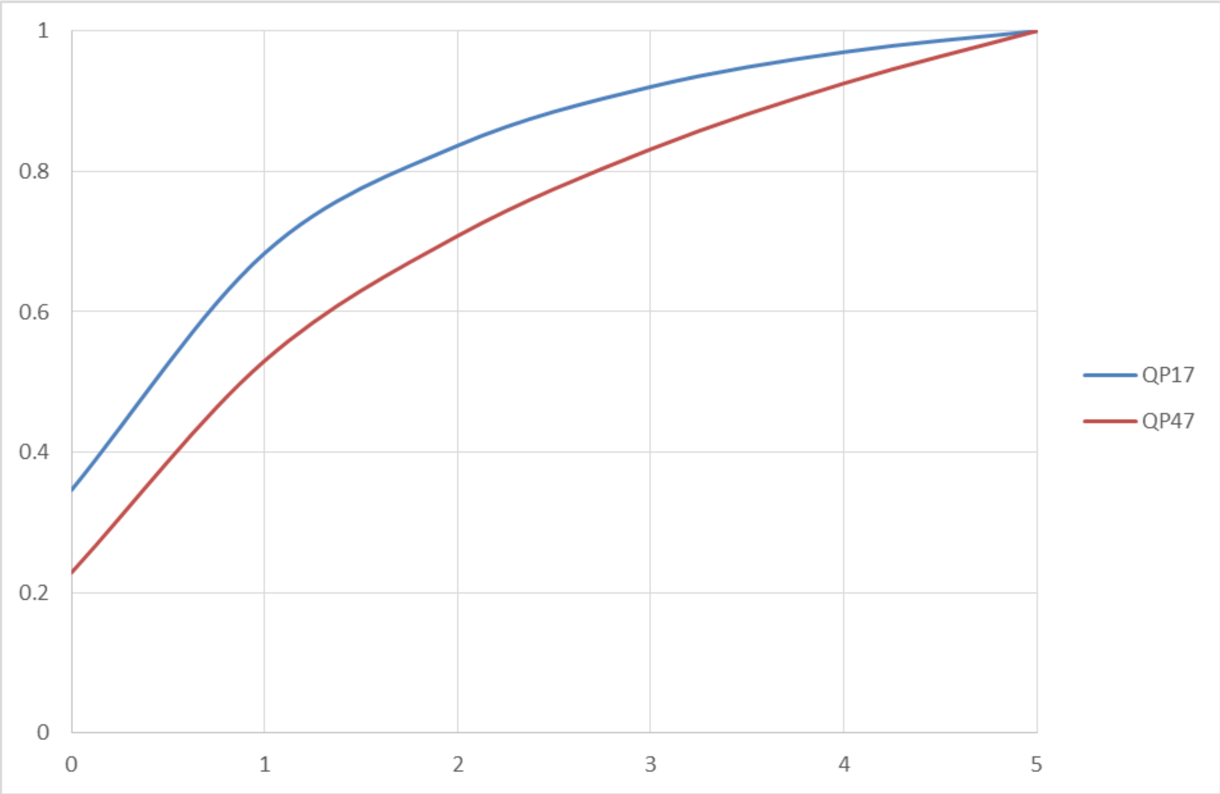

Fig. 4.6 Cumulative distribution of errors for the keypoint ‘Orientation’ parameter (HEVC encoding, PoznańCarpark sequence).

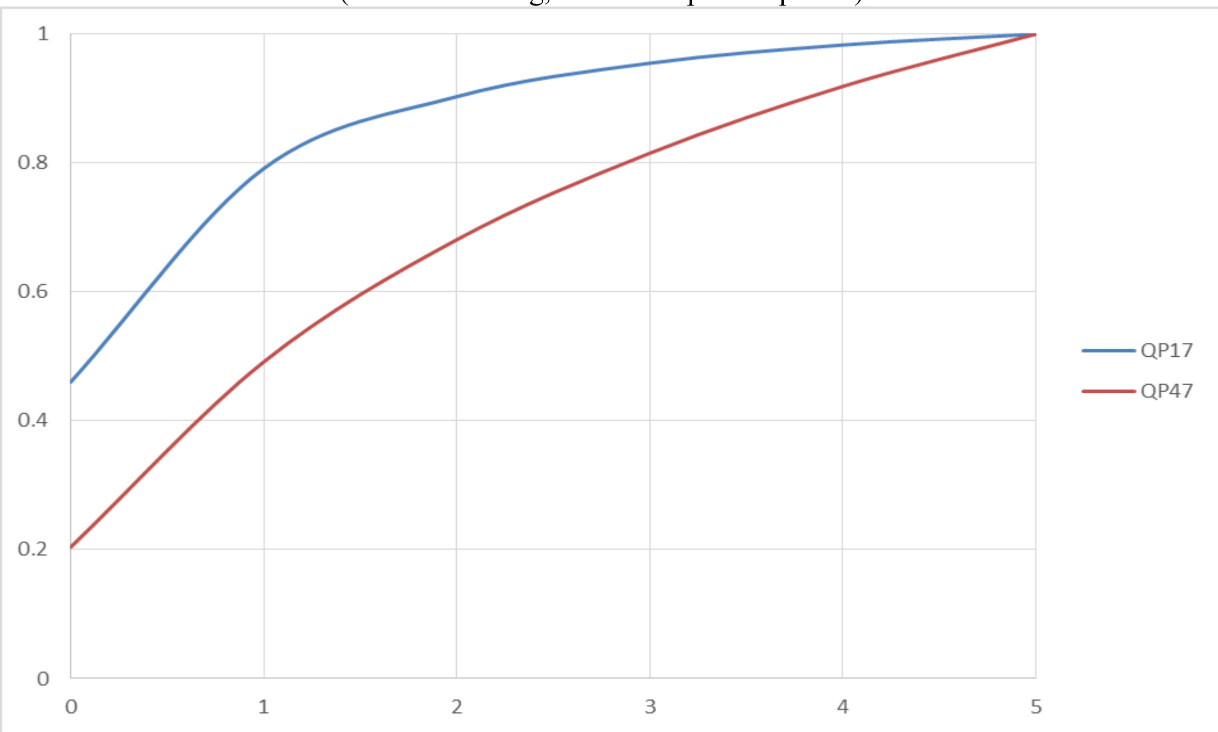

Fig. 4.7 Cumulative distribution of errors for the keypoint ‘Orientation’ parameter (HEVC encoding, PoznańStreet sequence).

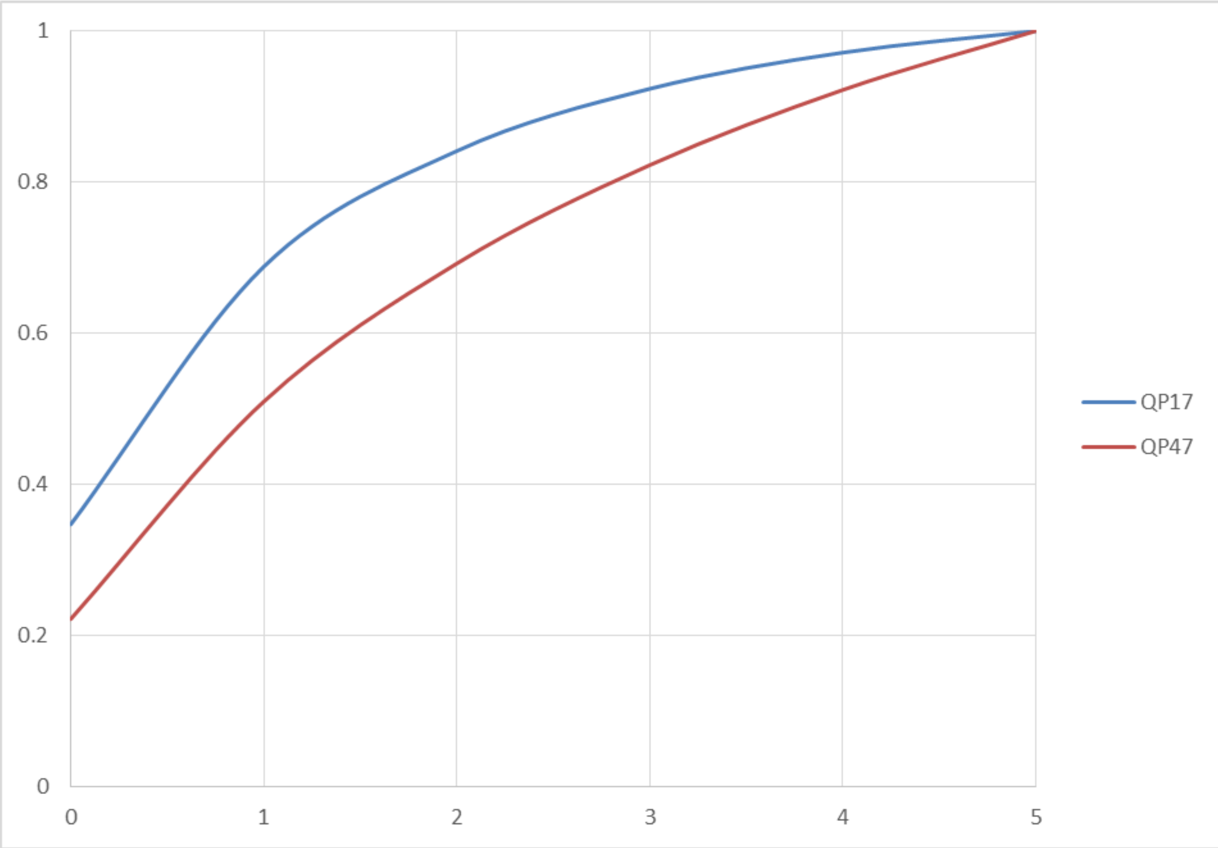

Fig. 4.8 Cumulative distribution of errors for the keypoint ‘Orientation’ parameter (VVC encoding, PoznańCarpark sequence).

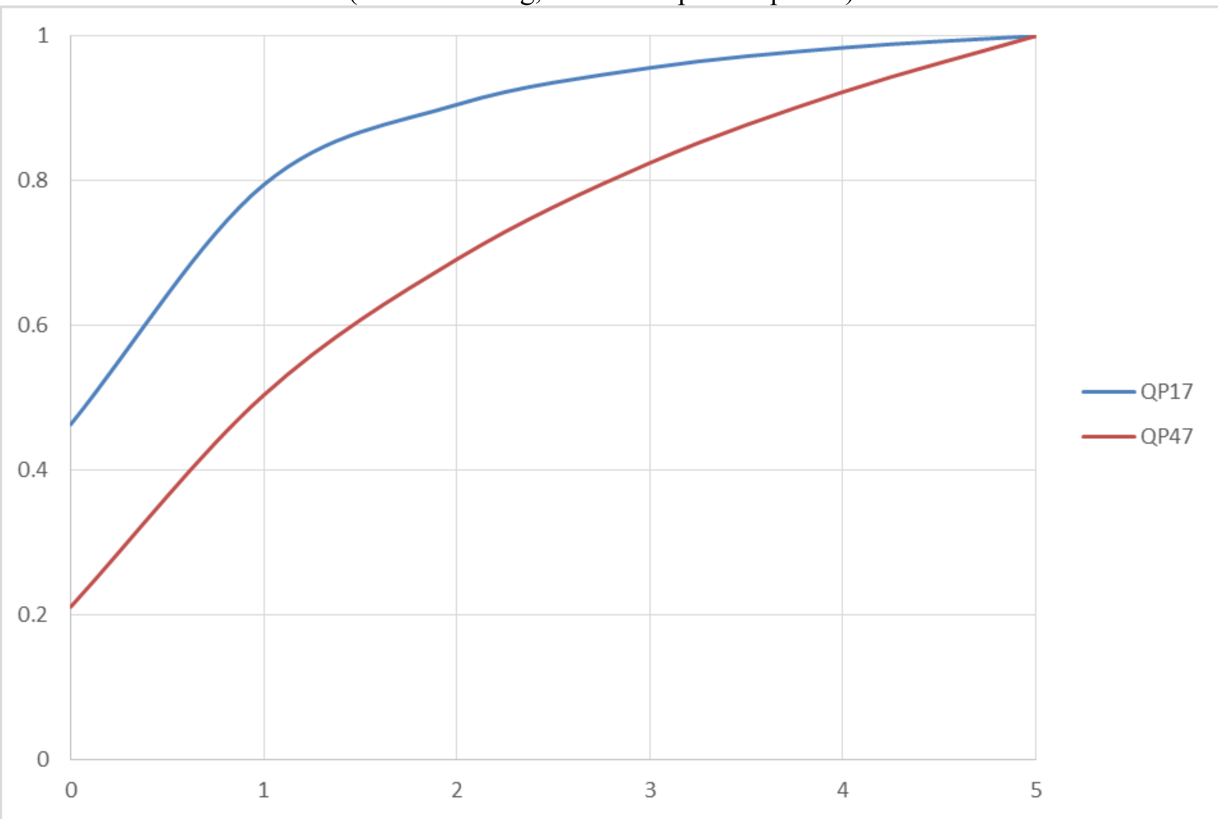

Fig. 4.9 Cumulative distribution of errors for the keypoint ‘Orientation’ parameter (VVC encoding, PoznańStreet sequence).

c) 'Size' parameter

The size parameter of keypoint is the radius, is a property of SIFT descriptor and it corresponds to a circle that wraps a squared patch of dimension. The size parameter affects the size of the area in which the descriptor associated with the keypoint is determined.

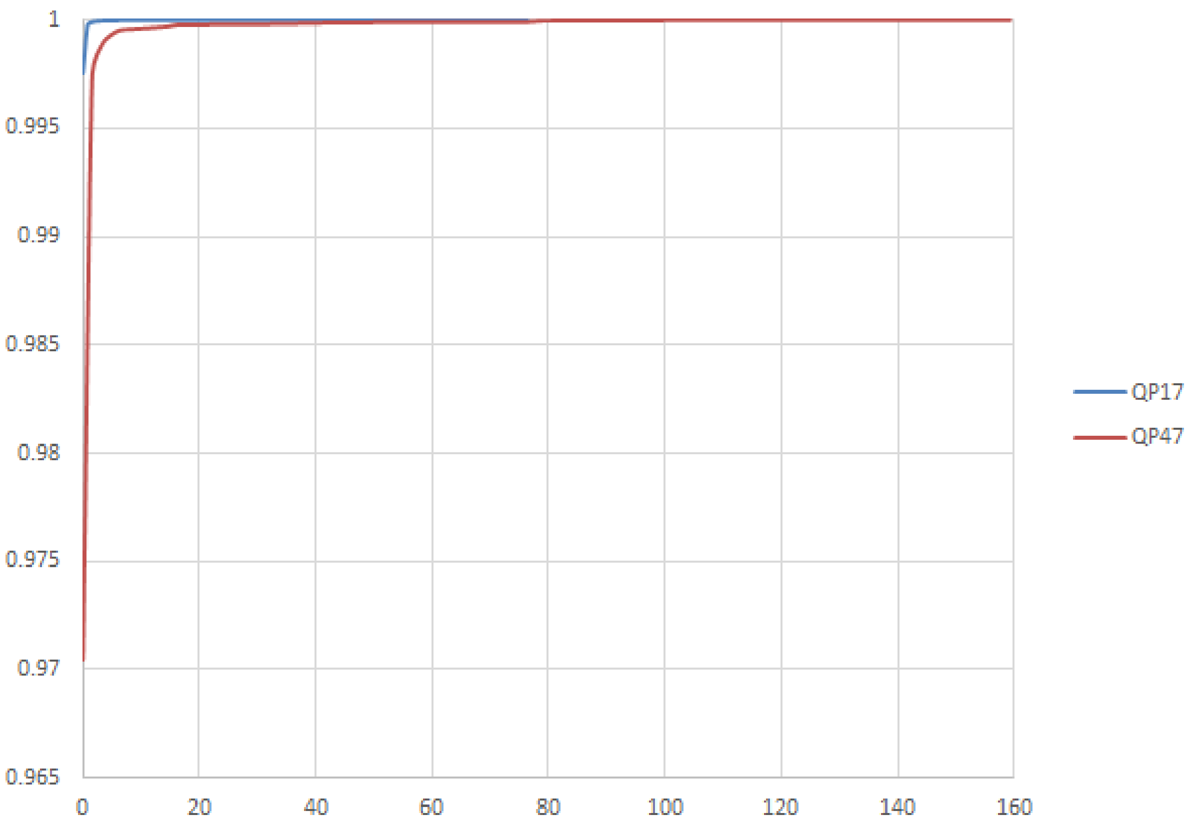

Fig. 4.10 Cumulative distribution of errors for the keypoint 'Size' parameter (HEVC encoding, PoznańCarpark sequence).

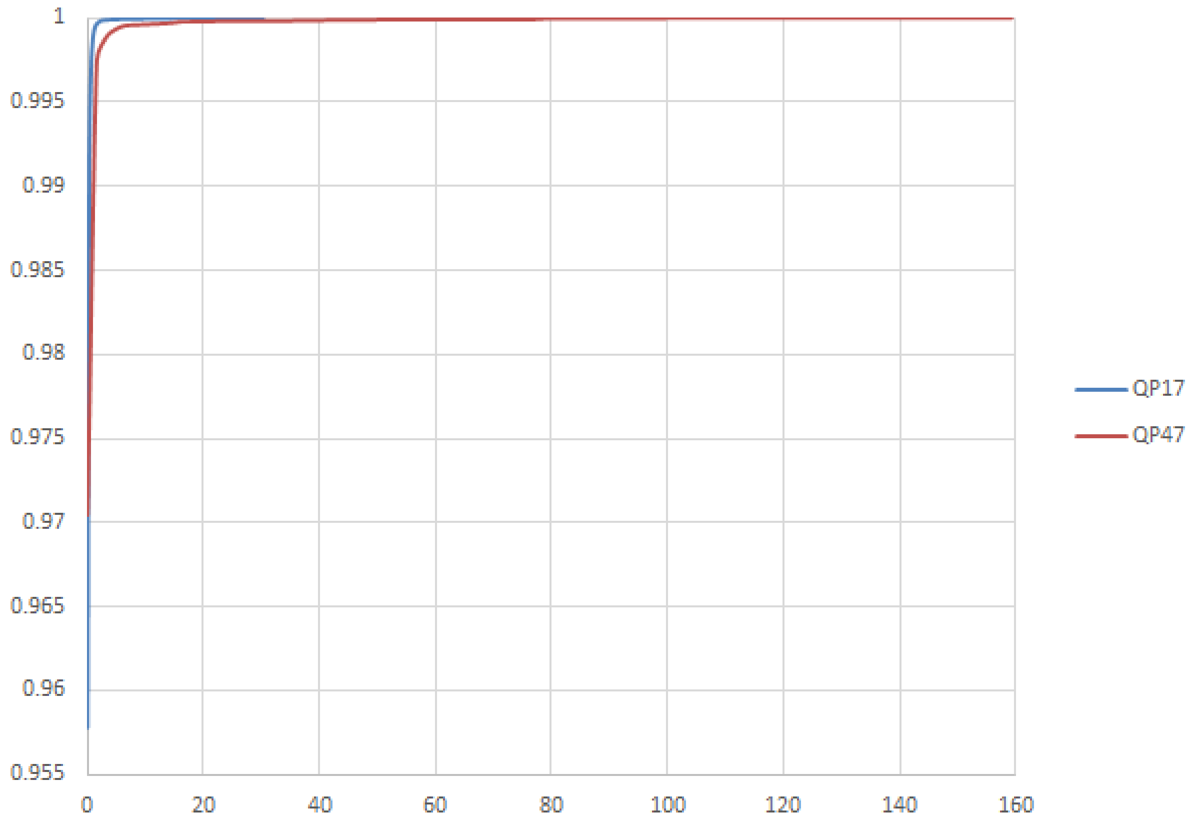

Fig. 4.11 Cumulative distribution of errors for the keypoint ‘Size’ parameter (HEVC encoding, PoznańStreet sequence).

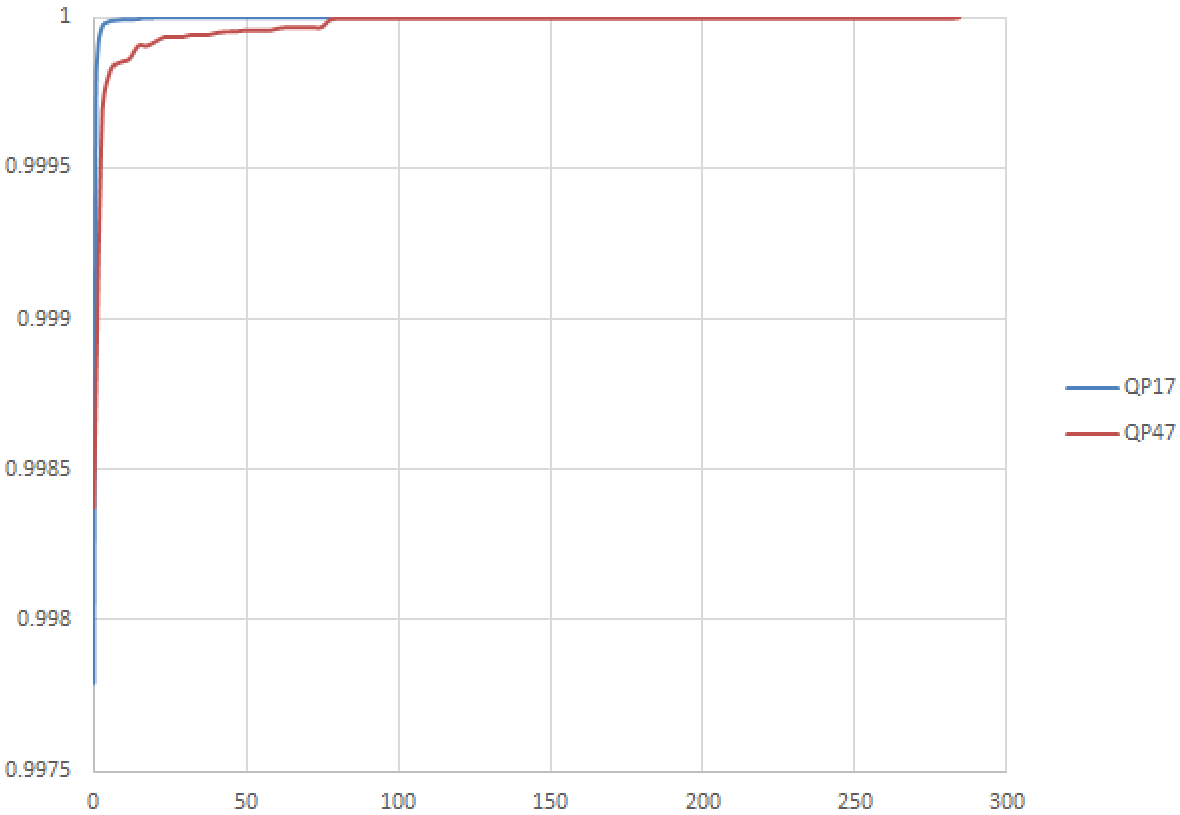

Fig. 4.12 Cumulative distribution of errors for the keypoint ‘Size’ parameter (VVC encoding, PoznańCarpark sequence).

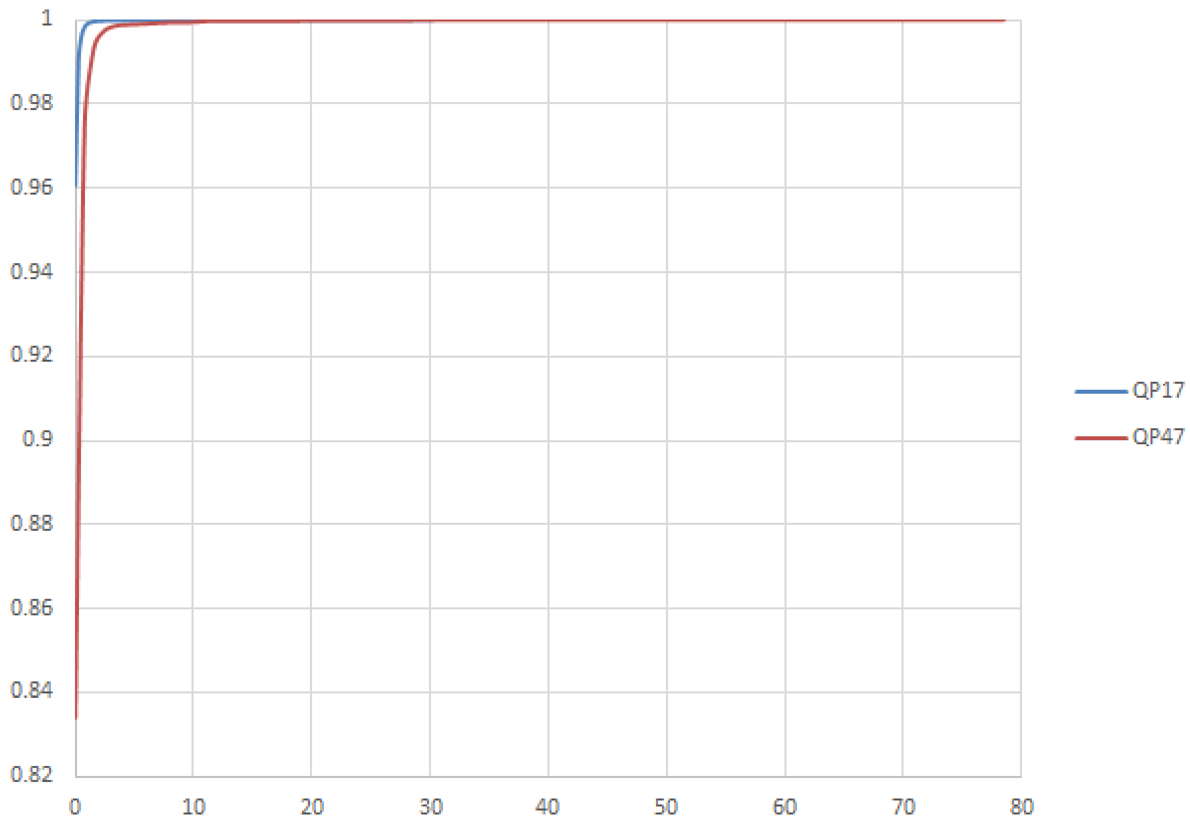

Fig. 4.13 Cumulative distribution of error for the keypoint ‘Size’ parameter (VVC encoding, PoznańStreet sequence).

The cumulative distributions from Figs. 4.2 – 4.13 demonstrate that for most keypoints that survive compression, the variations of their parameters are relative small. Therefore only minor residual information would need to be transmitted as side information for those keypoints.

5. Conclusions

The purpose of the experiments was to discover the influence of compression SIFT keypoint parameters. The knowledge on such effects is required in order to develop mechanisms that will reduce the redundancy between SIFT parameters and compressed video. The results roughly appoint the amount of data that needs to be corrected on the receiver side by the use of some side information on “lost” SIFT keypoints. Such a hybrid transmission of compressed video and a part of features appears as an interesting solution for prospective Video Coding for Machines.

At the same time, it is important to note that MPEG CDVS [11] uses parameters in the keypoint detection phase, which are determined in a similar way as in SIFT.

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] **Błąd! Nie można odnaleźć źródła odwołania.**, "**Błąd! Nie można odnaleźć źródła odwołania.**",.