INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG04 MPEG VIDEO CODING

ISO/IEC JTC1/SC29/WG04/M74618
Oct. 2025, Geneva, CH

Title	**[GSC][JEE 6.7] Efficiency of the three-component rotation for video-based GSC**
Source	**Poznan University of Technology** **Electronics and Telecommunications Research Institute**
Authors	**Adrian Dziembowski, Błażej Szydełko, Kwan-Jung Oh,** **Gwangsoon Lee, Jun Young Jeong**

1 Abstract

This document presents the results of an approach presented in M73364, allowing for reduction of the number of components needed to represent quaternions from 4 to 3 without a need to use Euler angles.

2 Algorithm (description from M73364)

The simplest solution to reduce the number of components needed to represent rotation is to convert quaternions to Euler angles. The equations are commonly known and easy to use. However, while conversion to Euler angles works and provides similar quality when no video compression is performed, it is slightly worse for video encoding, as in some borderline cases pixels change their value from very bright to very dark resulting in temporal flickering.

Therefore, we propose to keep the quaternion representation, but to reduce the number of components by getting rid of the W component. Such a reduction is possible as the quaternions representing rotation are redundant and changing their scale does not change the rotation. So the quaternion [X,Y,Z,W] = [1,2,3,4] represents the same rotation as [2,4,6,8] and so on.

The idea is to normalize all the quaternions - and to divide all the components by W: [X/W, Y/W, Z/W, W/W], basically setting W = 1 for all the splats. In this case the decoder assumes W = 1 and needs only three components per splat.

Such an approach works fine, unless W is very close to zero (or zero – then it does not work at all).

The solution to such a problem is to rotate the splats by 90 degrees around one of the X, Y, Z axes. The rotation changes the quaternion (including W), and among four rotations (no rotation, rotation around X, Y, and Z) we choose the one which gives us the biggest absolute value of W. Then we can do the normalization [X/W, Y/W, Z/W, W/W] without any problems.

The 90-degree rotation of the splat is performed by exchanging the scale components. For example, when the splat is rotated around X axis, SX and SZ are exchanged.

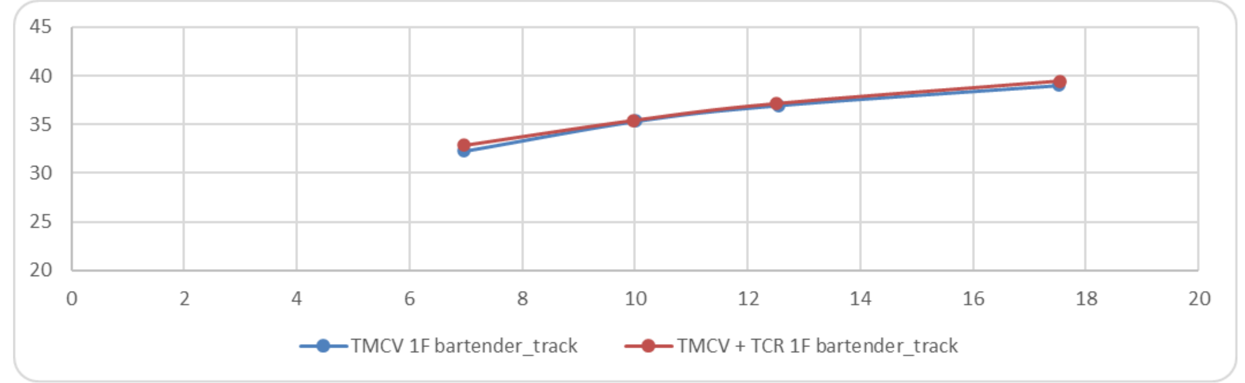
3 Results

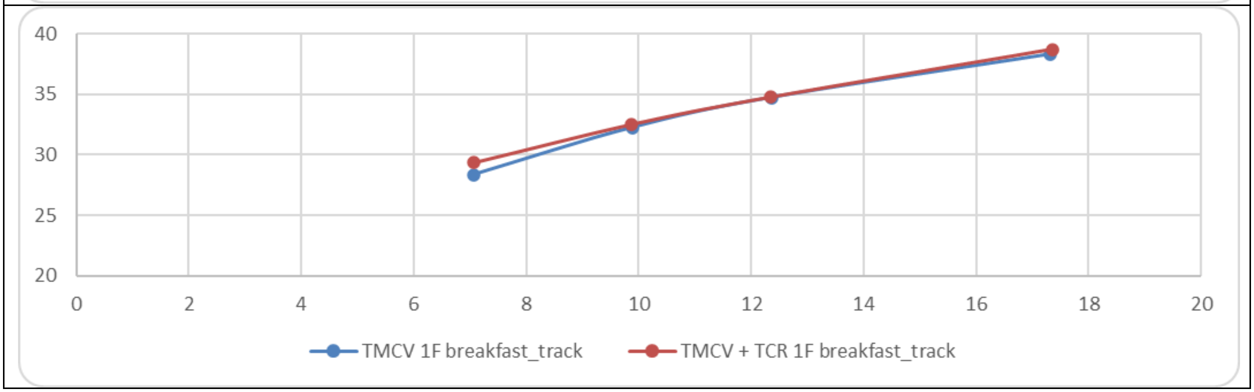
The algorithm has been implemented in MPEG GSC TMCV and has been tested for 1 frame and 32 frames sequences:

3.1 1 Frame

Sequence	BD-rate RGB-PSNR	BD-rate YUV-PSNR	BD-rate YUV-SSIM
bartender_track	-3.3%	-4.3%	-4.4%
breakfast_track	-2.6%	-2.1%	-3.7%

RGB-PSNR:

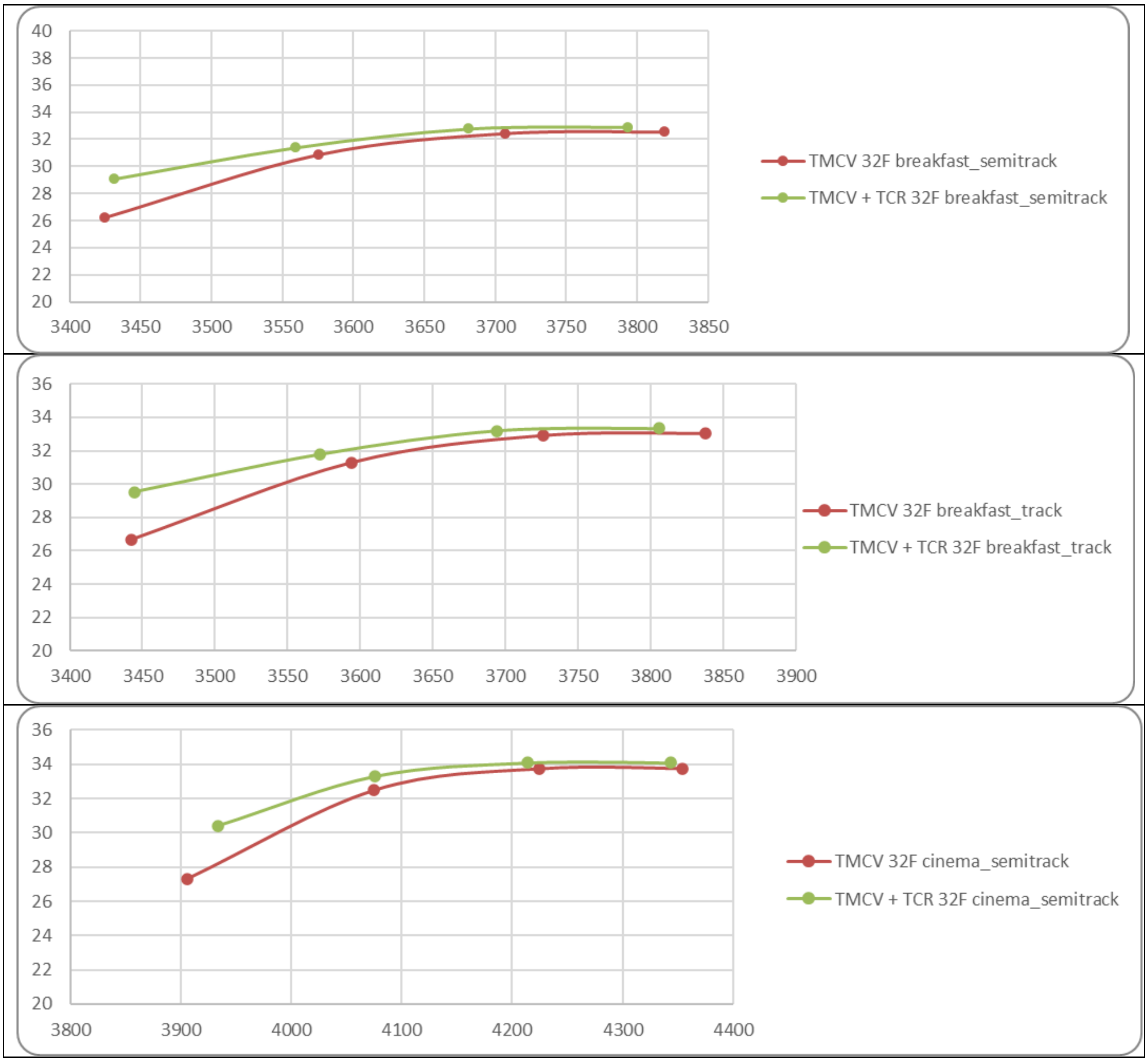


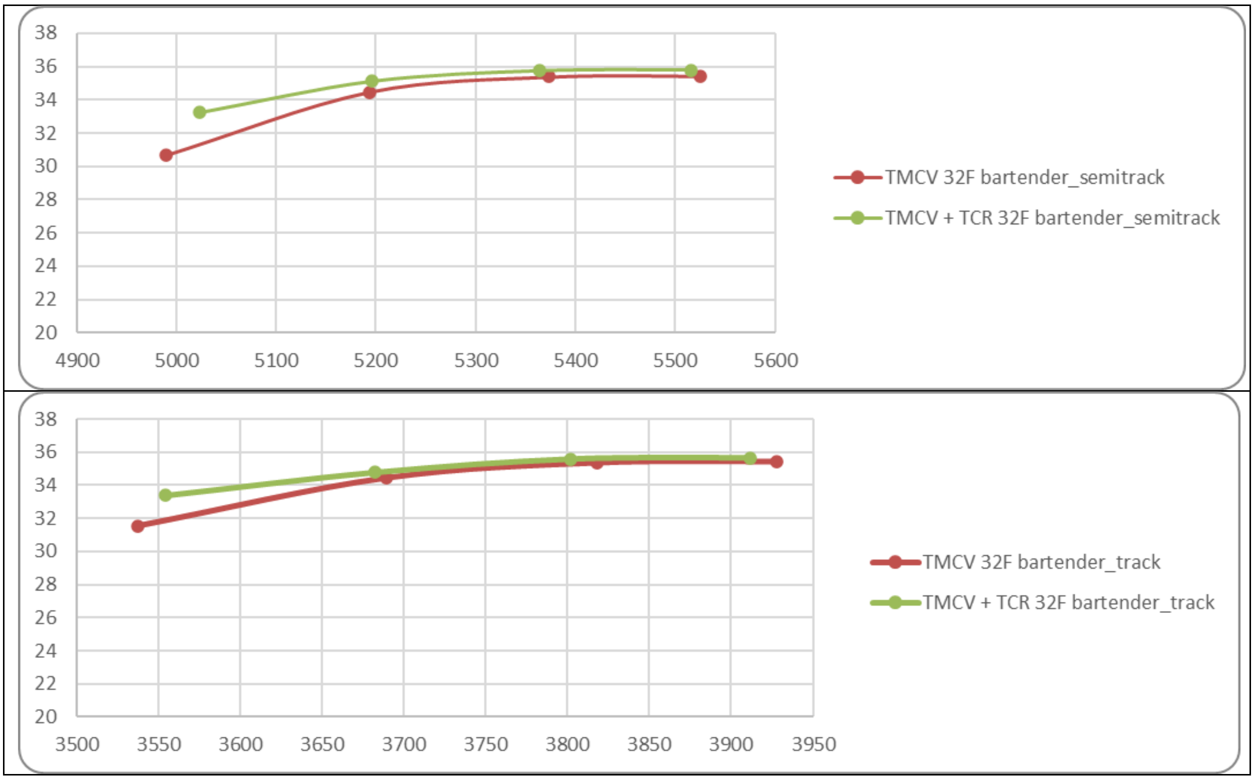


3.2 32 Frames:

Sequence	BD-rate RGB-PSNR	BD-rate YUV-PSNR	BD-rate YUV-SSIM
breakfast_untrack	#ARG!	#ARG!	#ARG!
breakfast_semitrack	-1,5%	-1,5%	-1,8%
breakfast_track	-1,6%	-1,6%	-2,0%
cinema_semitrack	-1,5%	-1,5%	-1,6%
cinema_track	#ARG!	#ARG!	#ARG!
bartender_semitrack	-1,7%	-1,7%	-1,6%
bartender_track	-1,3%	-1,4%	-1,4%

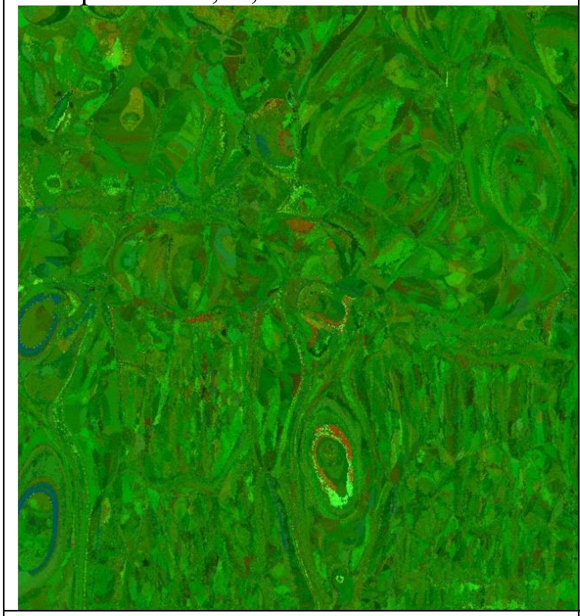
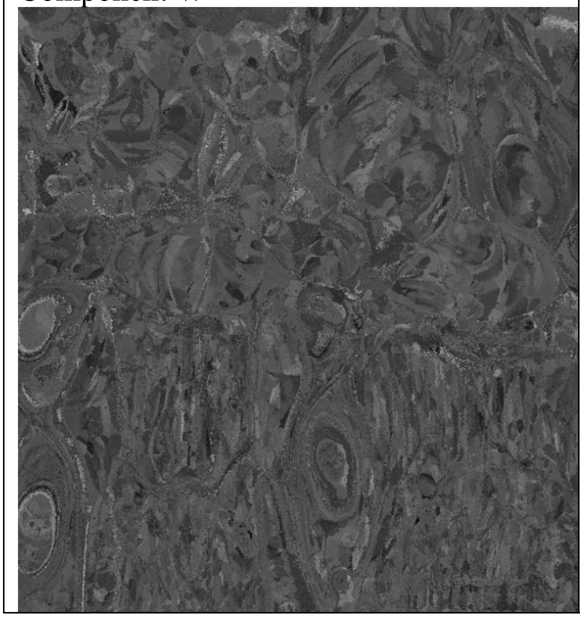
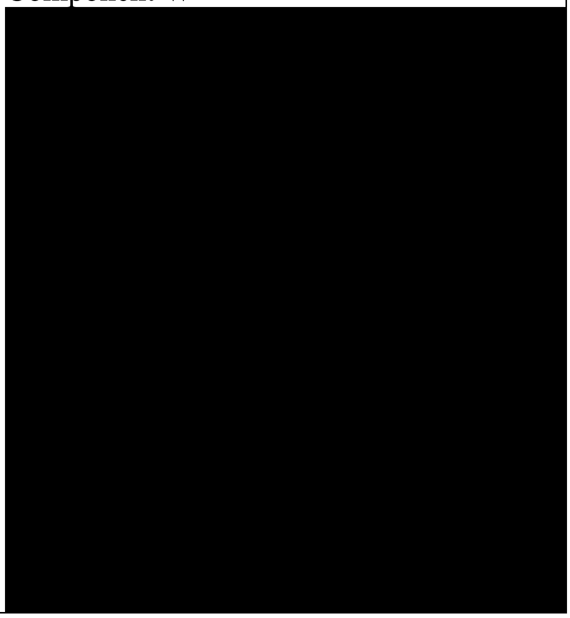
RGB-PSNR results:




38
36
34
32
30
28
26
24
22
20
4900
5000
5100
5200
5300
5400
5500
5600
TMCV 32F bartender_semitrack
TMCV + TCR 32F bartender_semitrack
3500
3550
3600
3650
3700
3750
3800
3850
3900
3950
TMCV 32F bartender_track
TMCV + TCR 32F bartender_track

Remarks:

- Only 3 rotation components are sent – TMCV generates one video with 3 quaternion components X, Y, and Z, and one flat video with single value over entire frame (W); W video could be skipped entirely with minor decoder changes.

Anchor	Proposed
Components X, Y, Z 	Components X, Y, Z 
Component W 	Component W 

- The quality gain is caused by decreasing the range between minimum and maximum value for each of three remaining quaternion components (for each splat the optimal rotation in terms of minimizing the range is chosen), e.g., for Bartender sequence:

Min value	X	Y	Z
Anchor	-1.9506	-2.1704	-2.2526
Proposed	-2.0922	-2.1351	-2.1454

Max value	X	Y	Z
Anchor	5.5092	6.4385	4.9861
Proposed	2.1785	2.1480	2.1135

Range	X	Y	Z
Anchor	7.4598	8.6089	7.2387
Proposed	**4.2707**	**4.2831**	**4.2589**

Quant step size (8 bps)	X	Y	Z
Anchor	$29.14 \cdot 10^{-3}$	$33.63 \cdot 10^{-3}$	$28.28 \cdot 10^{-3}$
Proposed	$16.68 \cdot 10^{-3}$	$16.73 \cdot 10^{-3}$	$16.64 \cdot 10^{-3}$

4 Recommendations

We recommend implementing the proposed algorithm in the MPEG GSC TMCV.

5 Acknowledgment

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2018-0-00207, Immersive Media Research Laboratory).