INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG07 MPEG 3DGH

ISO/IEC JTC1/SC29/WG07/M73343
Jul. 2025, Daejeon, KR

Title **[GSC] Cross-component prediction in video-based GSC**

Source **Poznan University of Technology**
Electronics and Telecommunications Research Institute
Author **Adrian Dziembowski, Błażej Szydełko, Dawid Mieloch,**
Gwangsoon Lee, Jun Young Jeong

1 Abstract

This document proposes an approach of cross-component predictive coding of splat attributes stored in yuv files, which increases the coding efficiency. Preliminary results show that this approach allows to reduce the bitstream by ~20%.

2 Algorithm

The idea is to find correlated components, and for some of them – instead of sending the original values – to send linear regression coefficients, together with the residual image.

The prediction is performed after the sorting (PLAS, FLAS, etc.), on all YUV color components, and all luma spherical harmonics. For rotations, positions, scales, and opacity the prediction is not performed.

In the first step, the correlation between all the pairs of components is calculated. The correlation is calculated in a block manner – separately in BxB blocks (e.g., 64x64 blocks) and then averaged over entire image. In case of multiple frames, correlation is averaged also over entire GOP.

Then, the biggest correlation is selected. For the pair of components with the highest correlation, the linear regression is calculated. This step is also performed in blocks. For each BxB block, separate A and B coefficients are being sent. Coefficient A is transmitted as 32-bit float, B – as a 16-bit integer.

For example, in the case of 768x768 video, 144 pairs of A and B coefficients are sent for a single pair of coefficients, resulting in additional 434 bytes of metadata (incl. 2 bytes for signaling coefficient ids once).

The first of the correlated components is transmitted as it is. The second one is predicted using the linear equation, and the video frame contains only the prediction errors.

Then, the second most correlated components are processed in the same way (note: already predicted component cannot be used for prediction of other components in order to reduce a drift caused by accumulation of errors), and the algorithm is repeated until there are no components which have correlation higher than a threshold (e.g., 25%).

3 Results

In total, three experiments were conducted. In each a different sorting algorithm was used, while the rest of the coding was exactly the same. In each experiment we have compared the proposed cross-component prediction with explicit sending of all the components.

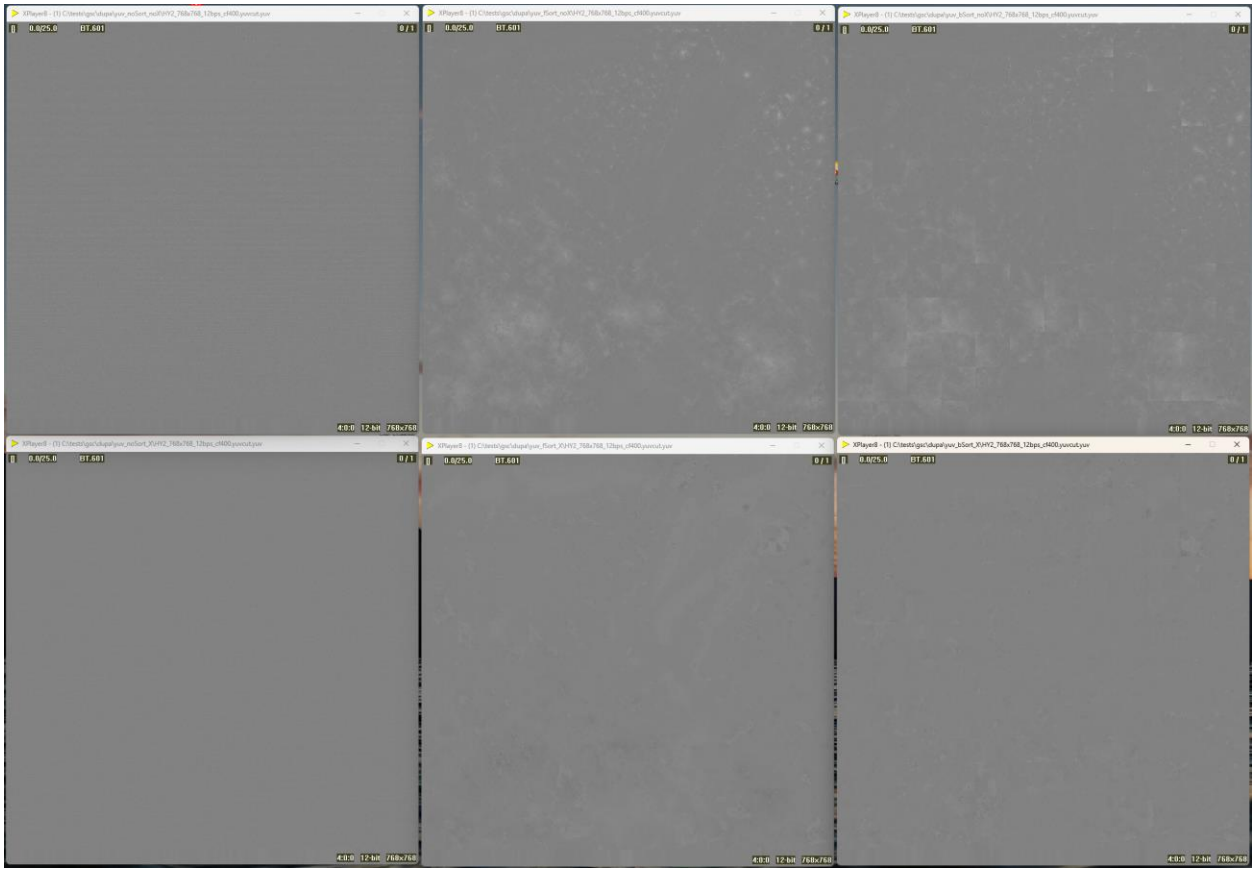

Fig. 1. Cross-component prediction (bottom row) vs. no prediction (top row) for three sorting methods: no sorting (first column), FLAS (second column), and block-FLAS (third column). HY2 component.

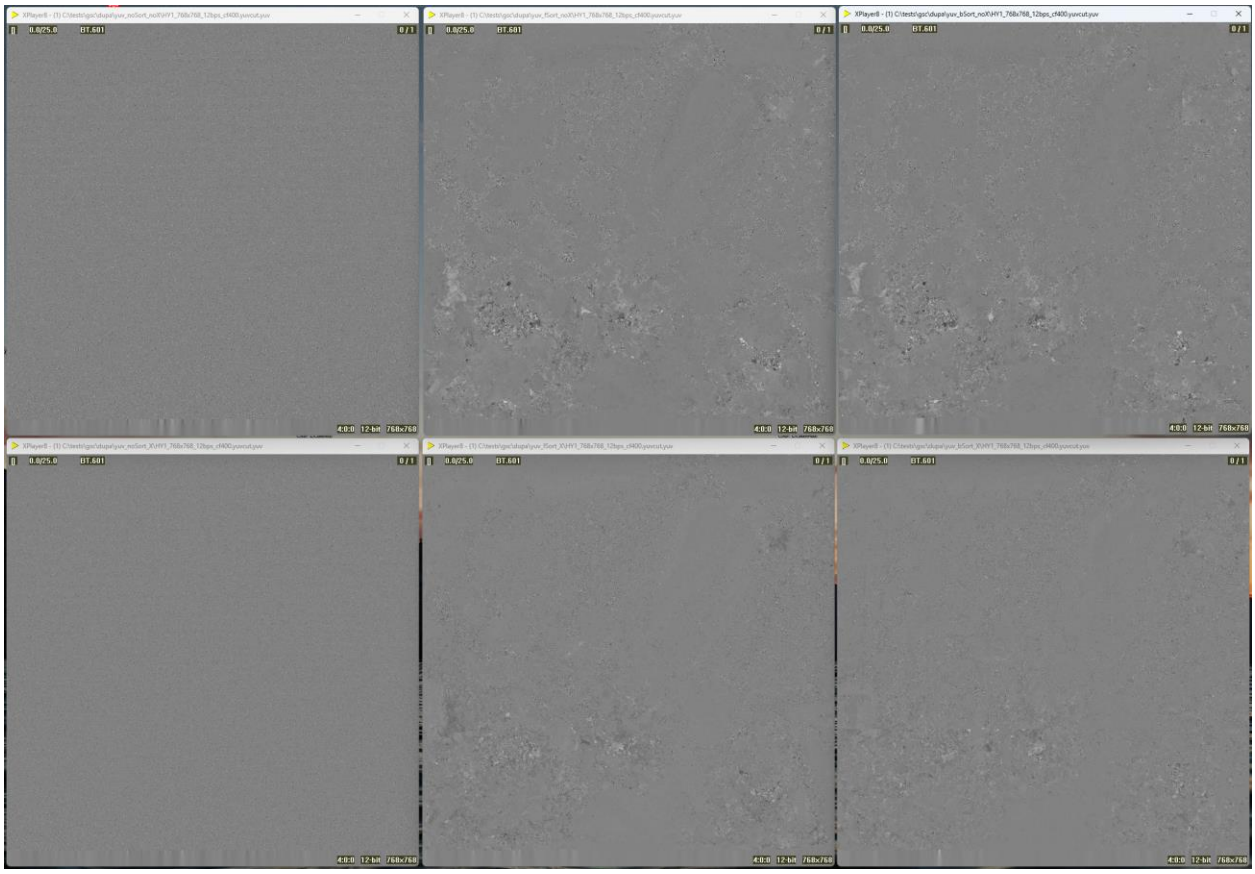

Fig. 2. Cross-component prediction (bottom row) vs. no prediction (top row) for three sorting methods: no sorting (first column), FLAS (second column), and block-FLAS (third column). HY1 component.

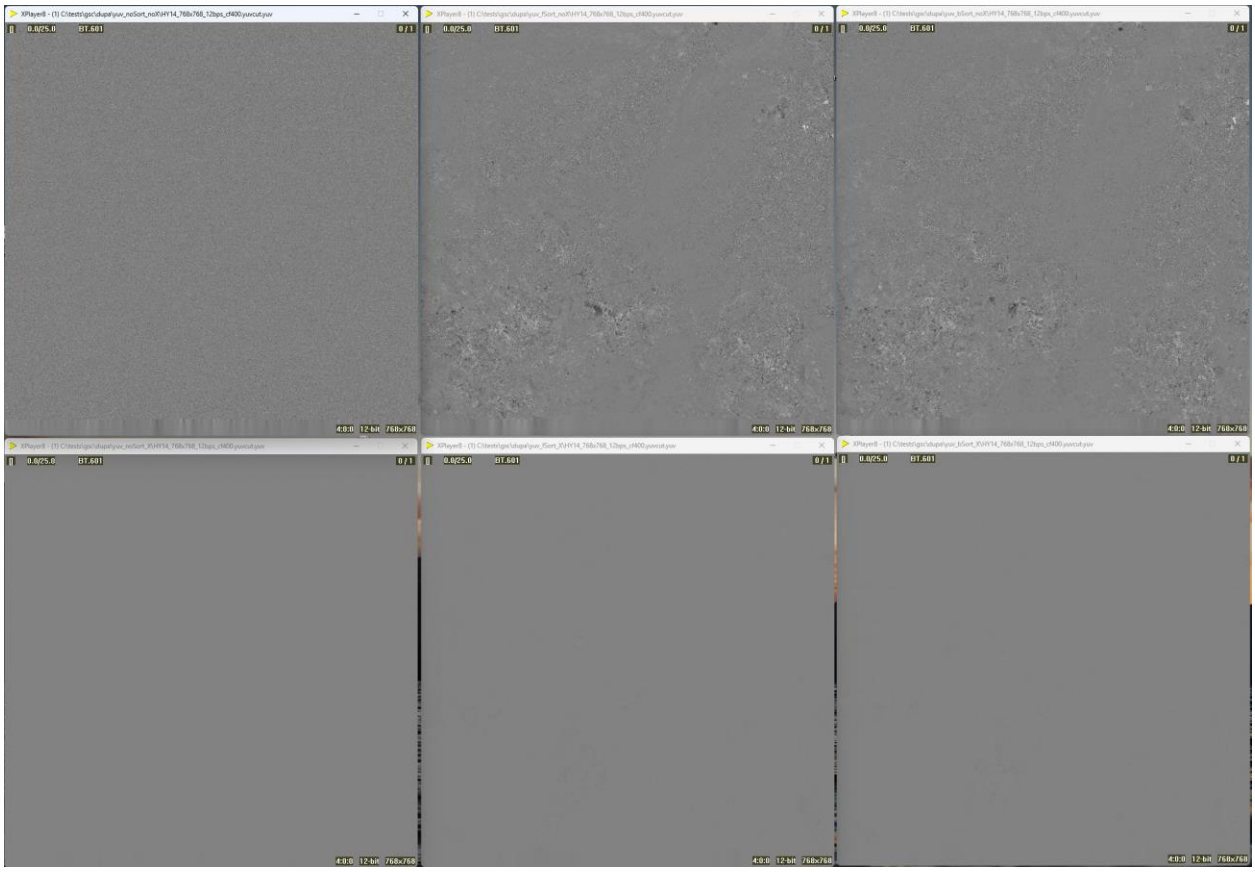

Fig. 3. Cross-component prediction (bottom row) vs. no prediction (top row) for three sorting methods: no sorting (first column), FLAS (second column), and block-FLAS (third column). HY14 component.

Table 1. Bytes needed to store a single frame of the Bartender scene in six tested cases. QP = 4.

Comp	No sorting		FLAS sorting		Block-FLAS sorting	
	No pred	Pred	No pred	Pred	No pred	Pred
CY	399634	399634	290260	290260	291296	291296
CU	291275	287359	251159	251096	251471	250100
CV	292177	292177	260876	261059	262368	261033
H1	316419	303491	274691	271551	275517	267539
H2	212289	180934	155423	131746	156448	130989
H3	296578	296578	265249	265229	265767	259788
H4	338157	338157	306411	306083	306824	300557
H5	316466	32376	273738	30711	274605	30960
H6	223498	92056	174293	74718	175728	74271
H7	297199	70667	265940	53228	265894	53588
H8	345795	315585	314980	282447	314574	278222
H9	370822	334139	337246	300465	337440	296068
H10	337944	80243	306162	57588	306626	57949
H11	316647	133719	273155	107160	274088	107749
H12	253529	201624	212983	175581	213699	176436
H13	301302	182233	269794	147067	269983	147364
H14	344956	80561	313996	59554	313669	59621
H15	388587	377885	361794	347718	362207	347051
Total C,H	5643274	3999418	4908150	3413261	4918204	3390581
Total All	9323141	7679285	7203704	5708815	7227798	5700175

Table 2. Bitrate change caused by using the proposed cross-component prediction.

	No sorting	FLAS sorting	Block-FLAS sorting
CY	100.0%	100.0%	100.0%
CU	98.8%	100.1%	99.6%
CV	100.1%	100.2%	99.7%
H1	96.1%	99.0%	97.3%
H2	85.4%	85.0%	84.0%
H3	100.1%	100.2%	97.9%
H4	100.1%	100.0%	98.1%
H5	10.4%	11.4%	11.4%
H6	41.4%	43.1%	42.5%
H7	23.9%	20.2%	20.3%
H8	91.4%	89.8%	88.6%
H9	90.2%	89.2%	87.9%
H10	23.9%	19.0%	19.0%
H11	42.4%	39.4%	39.5%
H12	79.7%	82.6%	82.8%
H13	60.6%	54.7%	54.7%
H14	23.5%	19.1%	19.1%
H15	97.4%	96.2%	95.9%
Total C,H	70.9%	69.5%	68.9%
Total All	82.4%	79.2%	78.9%

Subjectively, the results obtained with and without sorting are similar. No objective quality was calculated in this experiment.

4 Recommendations

We recommend to explore the approach of cross-component prediction further.

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).