INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC 1/SC 29/WG 4
MPEG VIDEO CODING

ISO/IEC JTC 1/SC 29/WG 4 m 60667
October 2022, Online

Title: **MIV Decoder-Side Depth Estimation profile**

Source: **Adrian Dziembowski, Dawid Mieloch (PUT)**
Jun Young Jeong, Gwangsoon Lee (ETRI)

Abstract

In this contribution, we describe a new "MIV Decoder-Side Depth Estimation" (MIV DSDE) profile, which can be a successor of the MIV Geometry Absent (MIV GA) profile for MIV ed.2. When compared to MIV GA, where geometry sub-bitstreams were completely absent in the bitstream, proposed MIV DSDE profile is much more flexible and covers more use cases and applications. It is recommended to (1) adopt the MIV DSDE profile in MIV ed.2 and (2) adopt casme_decoder_side_depth_estimation_flag in the casps_miv_extension.

1 Introduction

The main idea of the MIV DSDE profile is to enable any depth estimation or refinement at the decoder side. However, the profile allows for including partial (or even complete) geometry information in the bitstream. This information may be used on the decoder side to improve the depth estimation process. Experimental results presented in m58048 and m59516 prove that such an approach is beneficial and allows to speed up the decoding process (which includes the depth estimation step) and also improve the quality of depth maps estimated at the decoder side.

The proposed MIV DSDE profile covers many already presented applications, including ones described in m58048 and m59516, but also potential future ones, e.g. the refinement of depth transmitted as a mosaic of patches (i.e., the refinement of geometry atlases for A17 and V17 anchors).

The proposed MIV DSDE profile is not backward compatible with MIV ed.1, because of the addition of **casme_decoder_side_depth_estimation_flag**.

2 Existing and potential use cases of decoder-side depth estimation

This section presents the review of already proposed methods of compression using DSDE. On the basis of performed review and the direction of MIV development, we also try to foresee the potential future DSDE-based immersive video compression methods.

2.1 MIV Geometry Absent

Usage of existing MIV GA profile follows basic principles of DSDE [3]: no geometry present in the bitstream. It was shown to work very efficiently for low bitrates [1], to be codec-agnostic [11] and depth-estimator-agnostic [12], [1]. This scheme unfortunately yields highest computational complexity.

2.1.1 Geometry Assistance SEI

A set of encoder-derived features which helps the decoder-side estimator to improve the quality of depth maps and to speed up the process [3]. It is done by including into the bitstream information, e.g., how to narrow the range of possible depth levels which should be considered for each block of the depth map, or which block can be skipped in the estimation, as the depth values from the previous frame can be copied and applied for static regions [6].

Extended geometry assistance SEI was already included into the MIV ed. 2, it includes possibility of using:

- typical geometry assistance (features sent for all views, no recursion, [13]),
- recursive features [5],
- features sent for a subset of views [14],
- recursive features sent for a subset of views [15],
- different level of details for different views (quantization step, block width, number of splits set per view),
- possibility for adding new schemes of feature extraction (not only block-based rectangular grid).

2.2 MIV Main without depth transmission (fully occupied patches)

This scheme was shown in [4]. The method is based on recovering pruned views, but no temporal redundancy removal [16] is being used, as depth maps are not included into bitstream, so occupancy cannot be extracted from it. The recovered pruned views were fed into unmodified IVDE.

2.3 MIV Main without depth transmission enhanced w/ external occupancy

The previous scheme can be enhanced by using the external occupancy video (not available for MIV Main, but available for MIV Extended). Here, the temporal redundancy removal can be still used.

2.4 Simplified MIV Main with depth refinement

Depth refinement is a form of depth estimation, therefore, such post-processing can be treated as decoder-side depth estimation. It was already mentioned by the Group that it should be tested, e.g. on the MIV Main without pruning (V17 anchor).

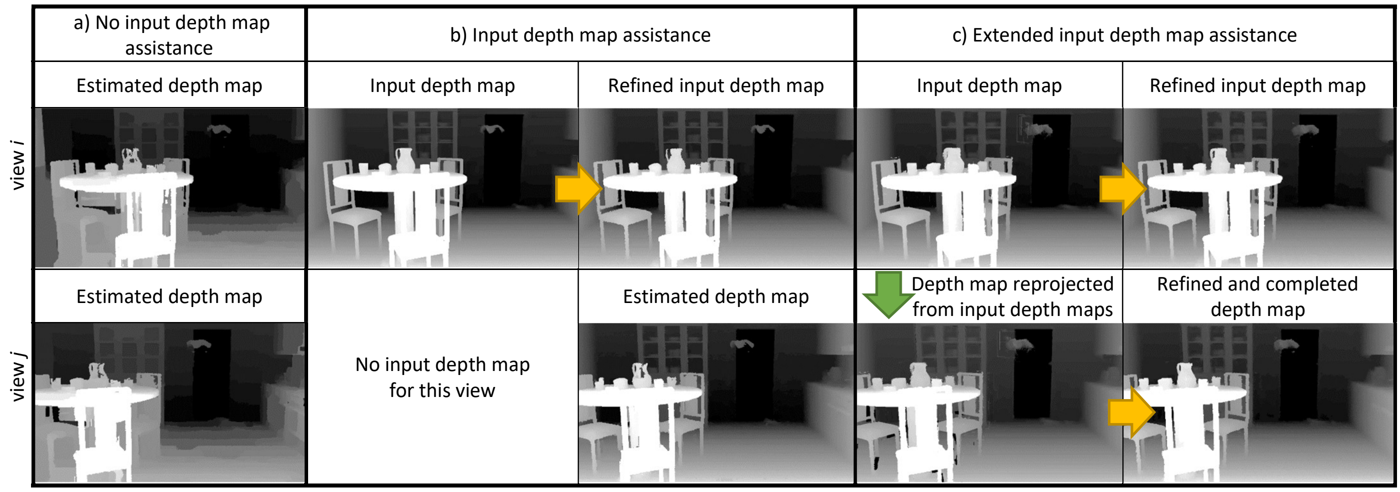
The software for depth refinement was earlier shared for MPEG activities [7], the only change would be using it on compressed video. Because this refinement method does not utilize any color information, its usability is very high for decoder-side depth estimation. The quality of encoded views can be very low, especially for very low bitrates. Use of the proposed refinement, independent of the quality of input views, can be particularly desirable in such applications.

2.5 MIV Main with depth refinement

The depth refinement can be also a part of MIV Main encoding (as in A97 anchor). This foreseen scheme would work similarly as in MIV Main without depth transmission [4], but here it is necessary to also recover pruned depth maps, which can be later refined.

2.6 Input Depth Map Assistance

Based on the scheme of encoding a part of depth maps in order to improve the depth estimation performed at the decoder side [8]. The input depth maps, available for a subset of views, are simultaneously refined and used to improve the quality of depth maps estimated for other views. It is possible by utilizing the modified depth estimation method based on global multi-view optimization.



Simplified overview of decoder-side depth estimation for two views, performed without assistance and for two proposed methods: a) depth maps for views i and j are estimated in the decoder; b) depth maps for views i and j are acquired in the joint process of estimation and refinement which utilizes input depth map of view i; c) input depth map of view i is reprojected to view j and becomes input depth map for view j, both input depth maps are used in the joint estimation and refinement process which results in depth maps for views i and j.

2.7 Extended Input Depth Map Assistance

In order to further improve the quality of estimated depth maps and decrease the computational complexity of their estimation, we proposed the approach, in which the set of available input depth maps is reprojected to other views for which the input depth maps are not available for all pixels [9]. These reprojected depth maps are also fed into the modified depth estimator which refines them in the same way as input depth maps in the previous approach, but also estimates depth for empty regions.

2.8 Towards real-time DSDE

This scheme is based on the approach described above, but no reestimation of transmitted depth or no reestimation of transmitted and reprojected depth is being done.

2.9 Usage of depth sensors

This is assumed that depth sensors data will be used in MIV ed.2 [10]. Now, in MIV Main, depth already can be sent without attributes, but this data is not used for rendering. We foresee that depth from sensors could be reprojected to other views and used as input depth map, as in the input depth map assistance scheme.

3 Proposal

Table A-1: Allowable values of syntax element values for the MIV toolset profile components

	Profile name				
Syntax element	**MIV Main**	**MIV Extended**	**MIV Extended Restricted Geometry**	**MIV Geometry Absent**	**MIV DSDE**
vuh_unit_type	V3C_VPS, V3C_AD, V3C_GVD, V3C_AVD, or V3C_CAD	V3C_VPS, V3C_AD, V3C_OVD, V3C_GVD, V3C_AVD, V3C_PVD, or V3C CAD	V3C_VPS, V3C_AD, V3C_AVD, V3C_PVD, or V3C CAD	V3C_VPS, V3C_AD, V3C_AVD, V3C_PVD, or V3C CAD	V3C_VPS, V3C_AD, V3C_OVD, V3C_AVD, V3C_GVD, V3C_PVD, or V3C CAD
ptl_profile_toolset_idc	64	65		66	67
ptl_profile_reconstruction_idc	255	255		255	255
ptc_restricted_geometry_flag	-	0	1	-	-
vps_miv_extension_present_flag	1	1	1	1	1
vps_packing_information_present_flag	0	0, 1	0, 1	0, 1	0, 1
vps_map_count_minus1[atlasID]	0	0	0	0	0
vps_occupancy_video_present_flag[atlasID]	0	0, 1	0	0	0, 1
vps_geometry_video_present_flag[atlasID]	1	0, 1	0	0	0, 1
vme_embedded_occupancy_enabled_flag	1	0, 1	0	0	0, 1
casme_decoder_side_depth_estimation_flag	0	0	0	1	1
gi_geometry_MSB_align_flag[atlasID]	0	0	0	0	0
ai_attribute_count[atlasID]	0, 1	0, 1, 2	2	0, 1	0, 1
ai_attribute_type_id[atlasID][attrIdx]	ATTR_TEXTURE	ATTR_TEXTURE, ATTR_TRANSPARENCY	ATTR_TEXTURE, ATTR_TRANSPARENCY	ATTR_TEXTURE	ATTR_TEXTURE
ai_attribute_dimension_minus1[atlasID][attrTextureIdx]	2	2	2	2	2
ai_attribute_dimension_minus1[atlasID][attrTransparencyIdx]	-	0	0	-	-
ai_attribute_dimension_partitions_minus1[atlasID][attrIdx]	0	0	0	0	0
ai_attribute_MSB_align_flag[atlasID][attrIdx]	0	0	0	0	0
asps_long_term_ref_atlas_frames_flag	0	0	0	0	0

asps_pixel_deinterleaving_enabled_flag	0	0	0	0	0
asps_patch_precedence_order_flag	0	0	0	0	0
asps_raw_patch_enabled_flag	0	0	0	0	0
asps_eom_patch_enabled_flag	0	0	0	0	0
asps_plr_enabled_flag	0	0	0	0	0
asme_patch_constant_depth_flag	0	0, 1	1	0, 1	0, 1
vps_geometry_video_present_flag[atlasID] || asme_patch_constant_depth_flag	-	1	1	0, 1	0, 1
vps_packed_video_present_flag[atlasID]	0	0, 1	0, 1	0, 1	0, 1
afps_lod_mode_enabled_flag	0	0	0	0	0
afps_raw_3d_pos_bit_count_explicit_mode_flag	0	0	0	0	0
afti_single_tile_in_atlas_frame_flag	1	0, 1	0, 1	0, 1	0, 1
ath_type	I_TILE	I_TILE	I_TILE	I_TILE	I_TILE
atdu_patch_mode[tileID][patchIdx]	I_INTRA	I_INTRA	I_INTRA	I_INTRA	I_INTRA
asps_atlas_sequence_parameter_set_id	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive
afps_atlas_frame_parameter_set_id	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive
afps_atlas_sequence_parameter_set_id	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive
aaps_atlas_adaptation_parameter_set_id	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive
ath_atlas_frame_parameter_set_id	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive
ath_atlas_adaptation_parameter_set_id	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive	0..63, inclusive

8.3.2.5 Common atlas sequence parameter set MIV extension syntax

casps_miv_extension( ) {	**Descriptor**
casme_depth_low_quality_flag	u(1)
casme_depth_quantization_params_present_flag	u(1)
casme_vui_params_present_flag	u(1)
casme_decoder_side_depth_estimation_flag	u(1)
if(casme_vui_params_present_flag)	
vui_parameters( )	
}	

8.4.2.5 Common atlas sequence parameter set MIV extension semantics

casme_decoder_side_depth_estimation_flag equal to 1 specifies that the V3C bitstream contains either no geometry sub-bitstream or geometry sub-bitstream with the geometry samples that are not intended to be used for view rendering. When not present, the value of casme_decoder_side_depth_estimation_flag is inferred to be equal to 0.

NOTE – A decoder should perform a depth estimation or refinement process in order to obtain geometry samples that are used for view rendering.

8.3.2.6.2 MIV view parameters list syntax

miv_view_params_list( ) {	**Descriptor**
mvp_num_views_minus1	u(16)
mvp_explicit_view_id_flag	u(1)
if(mvp_explicit_view_id_flag)	
for(v = 0; v <= mvp_num_views_minus1; v++)	
mvp_view_id[v]	u(16)
for(v = 0; v <= mvp_num_views_minus1; v++) {	
camera_extrinsics(v)	
mvp_inpaint_flag[v]	u(1)
}	
mvp_intrinsic_params_equal_flag	u(1)
for(v = 0; v <= mvp_intrinsic_params_equal_flag ? 0 : mvp_num_views_minus1; v++)	
camera_intrinsics(v)	
if(casme_depth_quantization_params_present_flag) {	
mvp_depth_quantization_params_equal_flag	u(1)
for(v = 0; v <= mvp_depth_quantization_equal_flag ? 0 : mvp_num_views_minus1; v++)	
depth_quantization(v)	
}	
mvp_pruning_graph_params_present_flag	u(1)

if (mvp_pruning_graph_params_present_flag)	
for(v = 0; v <= mvp_num_views_minus1; v++)	
pruning_parents(v)	
if(casme_decoder_side_depth_estimation_flag) {	
mvp_depth_reprojection_flag	u(1)
}	
}	

8.3.2.6.2 MIV view parameters list syntax

mvp_depth_reprojection_flag equal to 1 specifies that the V3C bitstream contains geometry samples which do not correspond to any attribute sample. mvp_depth_reprojection_flag equal to 0 specifies that the V3C bitstream contains no geometry samples which do not correspond to any attribute sample. When not present, the value of mvp_depth_reprojection_flag is inferred to be equal to 0.

NOTE – A decoder can reproject transmitted geometry samples between views before the depth estimation or refinement process.

4 How to use the profile with existing and potential compression methods with decoder-side depth estimation

	G17 anchor	Input depth map assistance	Extended input depth map assistance assistance	V17 anchor	A17 anchor	A17 without depth transmission (fully-occupied patches)	A17 without depth transmission enhanced w/ external occupancy	V17 with depth refinement	A17 with depth refinement	Faster extended input depth map assistance
ai_attribute_count[atlasID]	1,1,1,1*	1,1,1	1,1,1,0	1,1	1,1	1,1	1,1	1,1	1,1	1,1,1,0
vps_geometry_video_present_flag[atlasID]	0,0,0,0	1,0,0	1,0,0,1	1,1	1,1	0,0	0,0	1,1	1,1	1,0,0,1
vps_occupancy_video_present_flag[atlasID]	0,0,0,0	0,0,0	0,0,0,0	0,0	0,0	0,0	1,1	0,0	0,0	0,0,0,0
vme_embedded_occupancy_enabled_flag	0	1	1	1	1	0	0	1	1	1
casme_decoder_side_depth_estimation_flag	1	1	1	0	0	1	1	1	1	1
mvp_depth_reprojection_flag	0	0	1	-	-	0	0	0	0	1
mvp_reestimate_all_geometry_flag	1	1	1	-	-	1	1	1	1	0
mvp_keep_transmitted_geometry_flag [v]	-	-	-	-	-	-	-	-	-	1
mvp_keep_reprojected_geometry_flag [v]	-	-	-	-	-	-	-	-	-	0/1

* Notation “1,1,1,0” means, that a flag is true for atlases 0, 1, and 2, and false for atlas 3.

Yellow color – possible future SEI.

5 Recommendations

We recommend:

- adopting the MIV DSDE profile in MIV ed.2,
- adopting **casme_decoder_side_depth_estimation_flag** in the casps_miv_extension,
- adopting **mvp_depth_reprojection_flag** in the miv_view_params_list.

6 References

[1] D. Mieloch et al., “Overview and Efficiency of Decoder-Side Depth Estimation in MPEG Immersive Video,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 32(9), pp. 6360-6374, 09.2022.

[2] ISO/IEC DIS 23090-12, Information technology — Coded Representation of Immersive Media — Part 12: MPEG immersive video.

[3] P. Garus et al., “Immersive Video Coding: Should Geometry Information be Transmitted as Depth Maps?,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 32(5), pp. 3250-3264, 05.2022.

[4] M. Milovanović, F. Henry, M. Cagnazzo and J. Jung, “Patch Decoder-Side Depth Estimation In Mpeg Immersive Video,” in ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1945-1949, 2021.

[5] B. Szydełko, A. Dziembowski, D. Mieloch, M. Domański, and G. Lee, “Recursive block splitting in feature-driven decoder-side depth estimation”, ETRI Journal, vol. 44, pp. 38– 50, 2022.

[6] A. Dziembowski, D. Mieloch, J.Y. Jeong, G. Lee, „[MIV] Extended geometry assistance SEI”, ISO/IEC JTC1/SC29/WG4 MPEG VC M60248, Online, 18-22.07.2022.

[7] D. Mieloch, A. Dziembowski and M. Domański, “Depth map refinement for immersive video,” IEEE Access, vol. 9, pp. 10778-10788, 2021.

[8] D. Klóska, D. Mieloch, A. Dziembowski, M. Domański, G. Lee, J.Y. Jeong, „Decoder-side depth estimation with input depth assistance”, ISO/IEC JTC1/SC29/WG4 MPEG VC M58048, Online, 11-15.10.2021.

[9] D. Mieloch, A. Dziembowski, B. Szydełko, D. Klóska, G. Lee, J.Y. Jeong, „[MIV] Decoder-side depth estimation with extended input depth assistance”, ISO/IEC JTC1/SC29/WG4 MPEG VC M59516, Online, 25-29.04.2022.

[10] V.K.M. Vadakital et al., “The MPEG Immersive Video Standard—Current Status and Future Outlook,” IEEE MultiMedia, vol. 29(3), pp. 101-111, 2022.

[11] A. Grzelka, A. Dziembowski, D. Mieloch, M. Domański, “The Study of the Video Encoder Efficiency in Decoder-side Depth Estimation Applications,” in International Conference on Computer Graphics, Visualization and Computer Vision WSCG, 2022.

[12] S.L. Ravi, “DIBR for Immersive Videos: Traditional Depth Estimators versus Learning-Based Depth Estimators,” in SVCP2022 - Summer School on Video Coding and Processing, 2022.

[13] G. Clare, “[MIV] Combination of m56626 and m56335 for Geometry Assistance SEI message”, ISO/IEC JTC1/SC29/WG4 MPEG VC 56950, Online, April 2021.

[14] B. Szydełko, “Partial geometry assistance information”, ISO/IEC JTC1/SC29/WG4 MPEG VC M58047, Online, October 2021,

[15] B. Szydełko, “Effectiveness of recursive splitting in feature extraction for subset of views”, ISO/IEC JTC1/SC29/WG4 MPEG VC M58334, Online, October 2021,

[16] A. Dziembowski et al. “Spatiotemporal redundancy removal in immersive video coding,” *Journal of WSCG*, vol. 30, no. 1-2, pp. 54-62, 2022.