

JCTVC-M0142:  
SCE5: Buffer Size Estimation of SCE5.2.x

Kazushi.Sato@jp.sony.com  
jun.xu@am.sony.com

# Agenda

- Introduction / Problem Statement
- Buffer Size Estimation
- Comparison of Coding Efficiency and Buffer Size
- Discussions
- Additional Remark on Complexity
- Conclusion

# Introduction / Problem Statement

- On the usage of motion data buffer for SHVC is being studied under SCE5.2.x.
  - It is proposed by SCE5.2.1 that temporal motion prediction be omitted at the enhancement layer to reduce the required buffer size, as the collocated base layer motion information takes part of TMVP in the enhancement layer.
  - It is proposed by SCE5.2.2 to postpone motion data compression after encoding/decoding of the enhancement layer, or 2-stage motion data compression for improving coding efficiency.
- At the 12th JCT-VC meeting in Geneva, it was suggested to study on the buffer size requirements for SCE5.2.x proposals.
- This contribution provides information on the buffer sizes required for each of SCE5.2.x proposals.
  - Trade-off of buffer sizes and coding efficiency is also studied.

# Buffer Size Estimation [1/]

- For each of the motion information, the following data needs to be stored:
  - Mvx (16bits)
  - Mvy (16bits)
  - Refidx (4bits)
- So totally  $16+16+4 = 36\text{bits}$  will be necessary.
  - If byte-alignment is considered, **40bits** will be necessary.
- Minimum inter PU size specified in HEVC version 1 is either 4x8 or 8x4.
  - With our implementation motion data for each of 4x4 units is prepared, and in the case that PU size is 4x8, same motion data are stored for 2 consecutive blocks in vertical direction, and in the case that PU size is 8x4, same motion data are stored for 2 consecutive blocks in horizontal direction.

# Buffer Size Estimation [2/]

- Assuming that
  - W: EL Width
  - H: EL Height
  - R: Scalability Ratio (2, 1.5 or 1)
  - D: Decimation Ratio of colblmv (4, 2 or 1)
  - N: Number of DPB
- Then (bi-directional case)
  - $EL\_pixel\_buf\_size = W * H * 8[bit] * 1.5 = 12 * W * H$
  - $BL\_mv\_buf\_size = (W/R) * (H/R) * (1/4D * 1/4D) * 40 [bit] * 2 = (5 / (D^2 R^2)) * W * H$
  - $EL\_mv\_buf\_size = W * H * (1/16 * 1/16) * 40 [bit] * 2 * N = 5/16 * N * W * H$
- Only L0 and L1 motion data of the current frame need to be stored for the base layer,
- whereas L0 and L1 data for all reference frames need to be stored for the enhancement layer as it may be used for encoding/decoding of the future frames within the same layer.

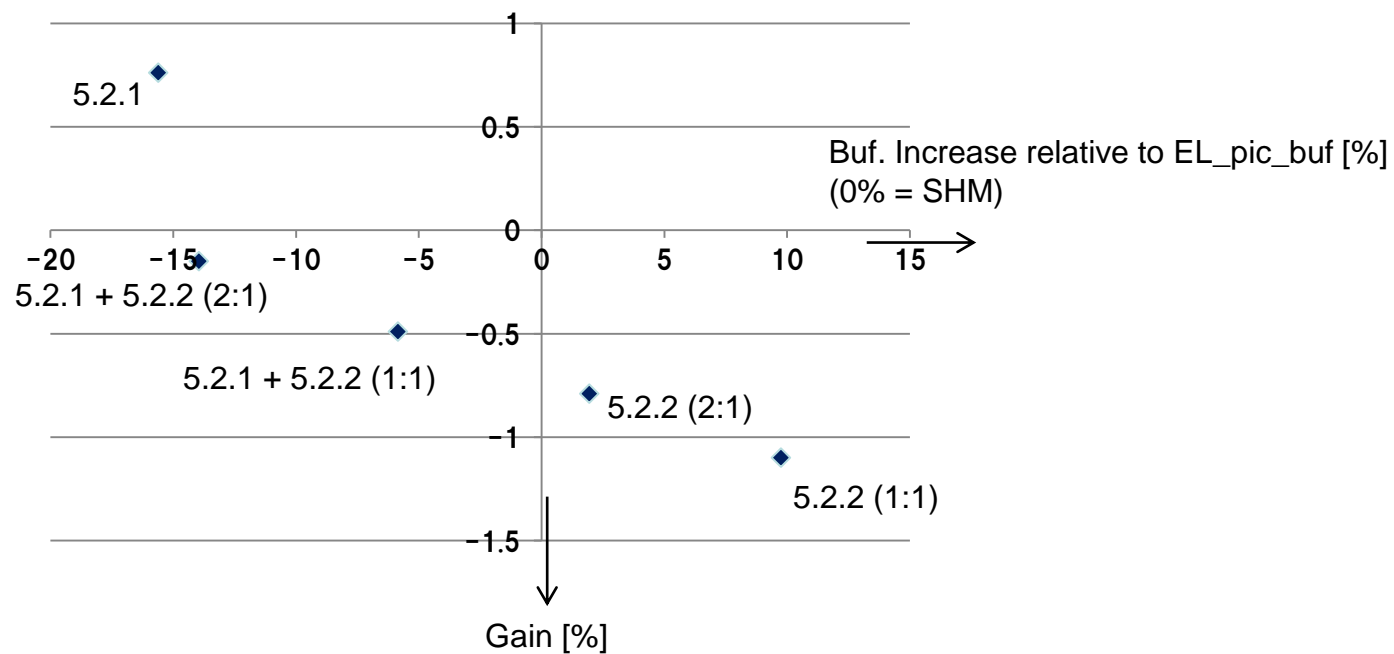
# Buffer Size Estimation [3/]

- Assuming that N=6 (bi-directional case)
  - $EL\_mv\_buf\_size / EL\_pixel\_buf\_size = (5/16) / 12 * 6 = 15.6\%$
  - $BL\_mv\_buf\_size / EL\_pixel\_buf\_size = (5 / D^2R^2) / 12 = 5 / (12 * D^2 * R^2)$ 
    - **=0.65% (R=2; D=4)**
    - **=1.16% (R=1.5; D=4)**
    - **=2.6% (R=1; D=4)**
    - **=2.6% (R=2; D=2)**
    - **=4.6% (R=1.5; D=2)**
    - **=10.4% (R=1; D=2)**
    - **=10.4% (R=2; D=1)**
    - **=18.5% (R=1.5; D=1)**
    - **=41.7% (R=1; D=1)**

# Comparison of Coding Efficiency and Buffer Size [1/]

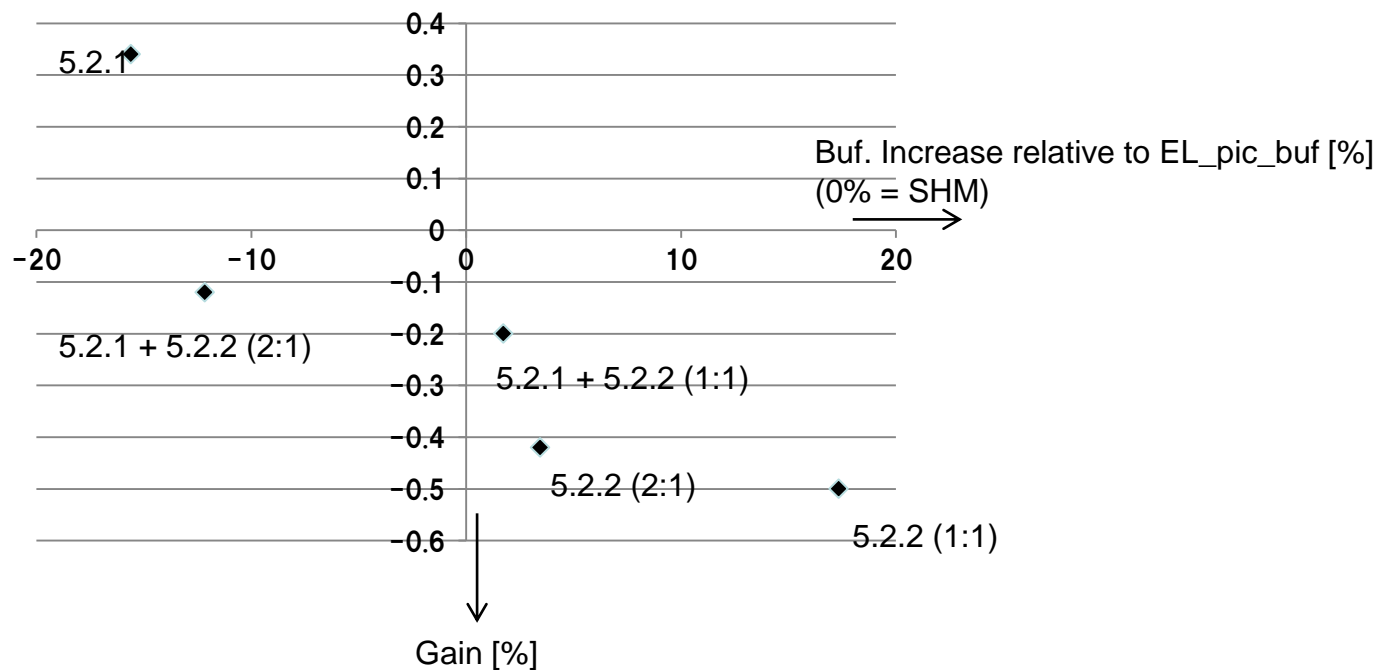
- Coding efficiency and required buffer sizes of the following methods are compared:
  - SCE5.2.1: Disabling TMVP
  - SCE5.2.2 (2:1) : Half BL MV buffer compression
  - SCE5.2.2 (1:1) : No BL MV buffer compression
  - SCE5.2.1 + SCE5.2.2 (2:1)
  - SCE5.2.1 + SCE5.2.2 (1:1)

# BufferSize vs Gain: RA2x

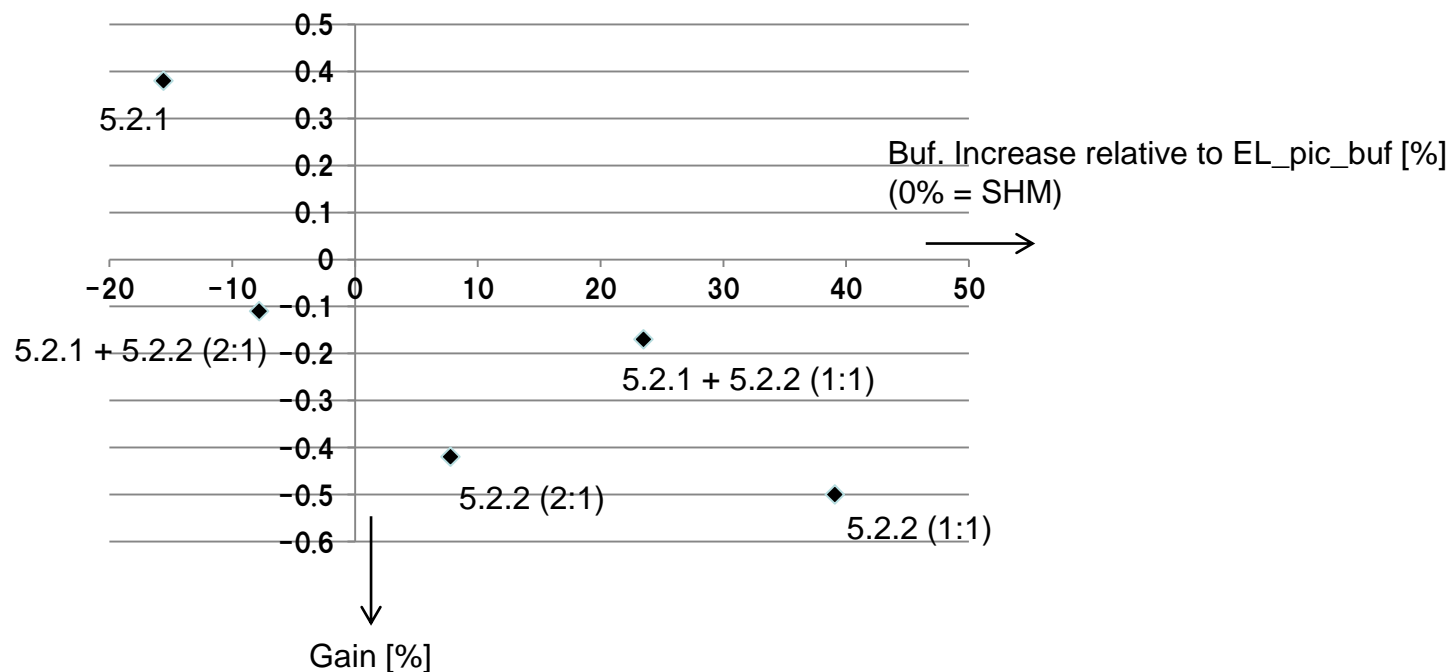




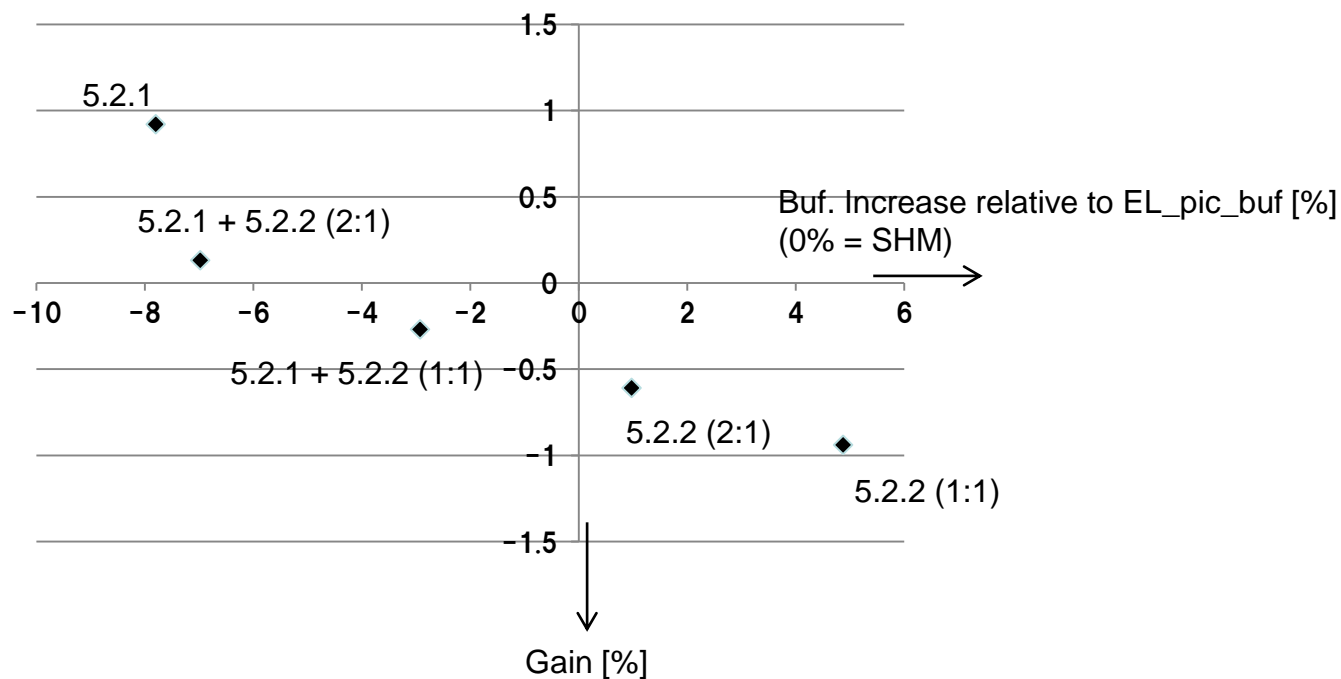
# BufferSize vs Gain: RA1.5x



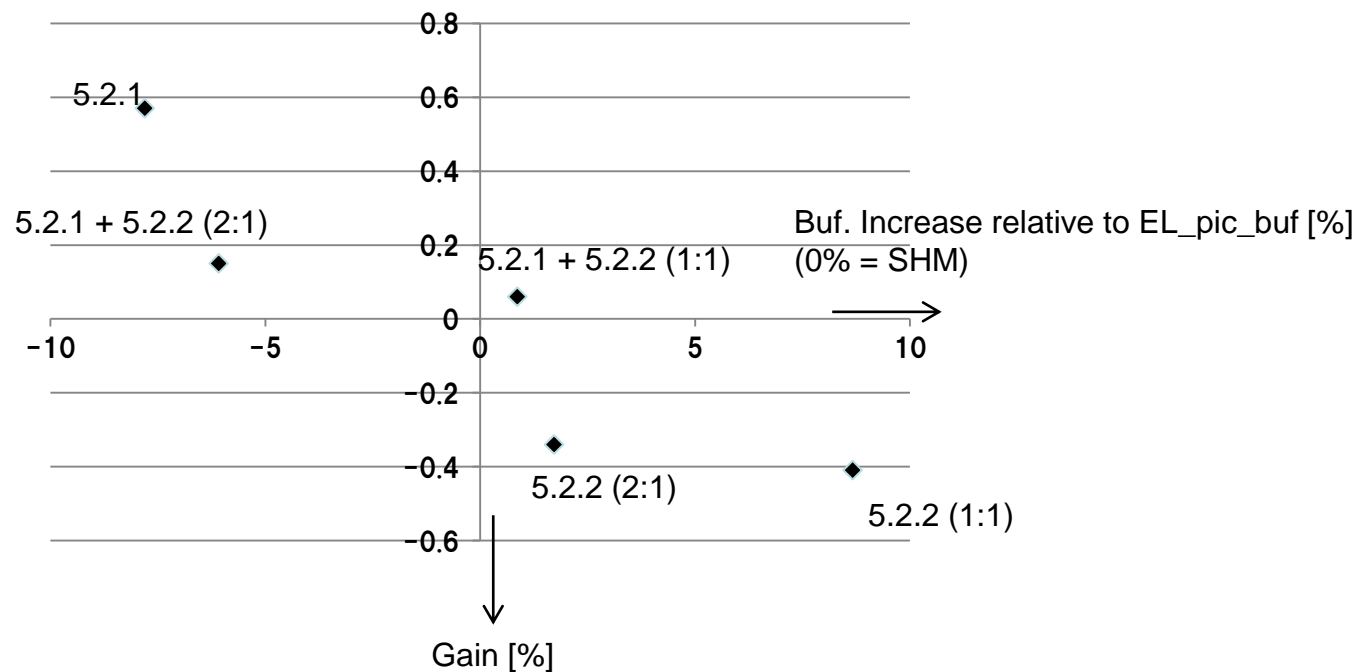
# BufferSize vs Gain: RASNR



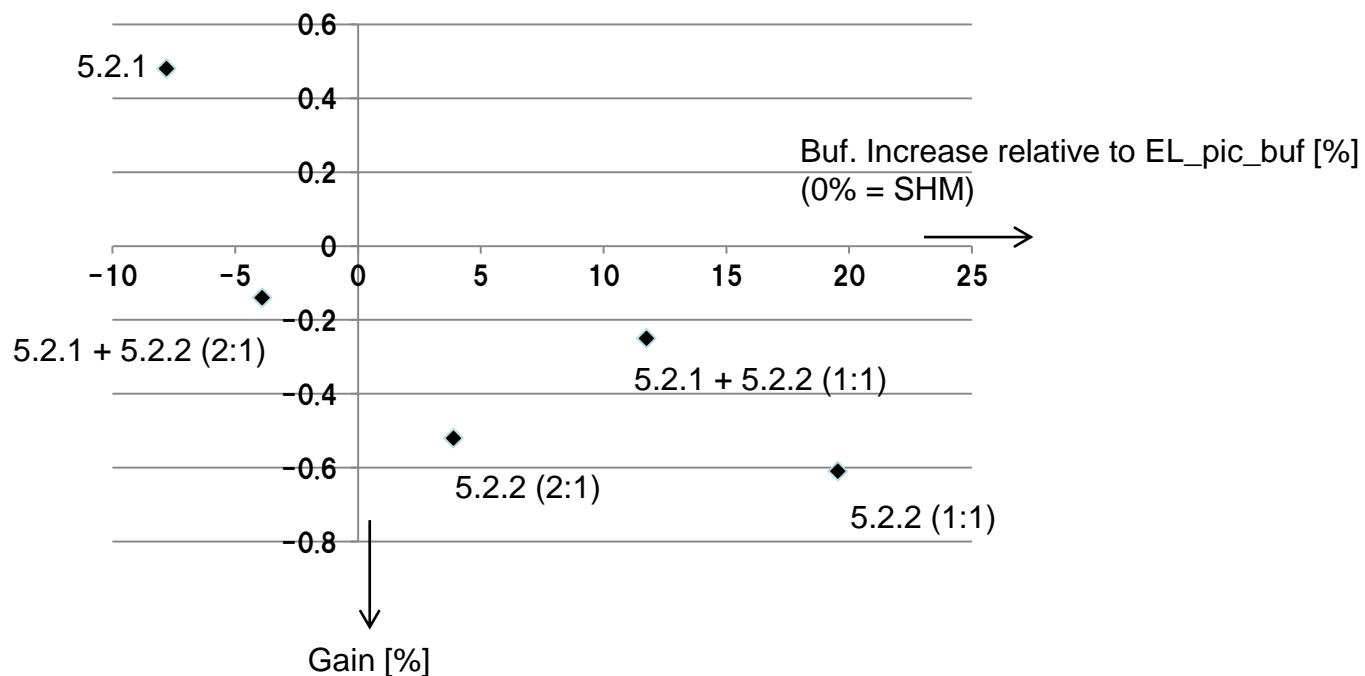
# BufferSize vs Gain: LD2x



# BufferSize vs Gain: LD1.5x



# BufferSize vs Gain: LDSNR



# Comparison of Coding Efficiency and Buffer Size [2/]

- Assuming, for example, that the resolution of the enhancement layer is 3840x2160 pixels and the bit-depth is 8, increase (positive values) or decrease (negative values) of buffer sizes with each of the proposals [in the unit of bytes] are shown below:

	RA_2x	RA_1.5x	RA_SNR	LD_2x	LD_1.5x	LD_SNR
5.2.2 (2:1)	242611	427991	970445	121306	213996	485222
5.2.2 (1:1)	1213056	2157373	4864666	606528	1078687	2432333
5.2.1	-1940890	-1940890	-1940890	-970445	-970445	-970445
5.2.1+5.2.2 (2:1)	-1735603	-1512899	-970445	-867802	-756449	-485222
5.2.1+5.2.2 (1:1)	-727834	216484	2923776	-363917	108242	1461888

# Discussions

- The method of SCE5.2.2 (1:1) brings around 1% gain for the RA/LD\_2x cases, but causes increase in buffer size by around 40% compared to EL\_pixel\_buffer\_size with the RA\_SNR case.
  - Similar gain can be obtained with SCE5:2:2 (2:1) with much less increase in buffer size.
- The method of SCE5.2.1 can reduce buffer size by 15.6% but causes loss in coding efficiency by 0.8 to 0.9% for RA/LD\_2x cases.
  - This loss can be compensated by combining with SCE5.2.2 (2:1) or SCE5.2.2 (1:1).
    - **Coding efficiency of both methods are similar while the latter requires more buffer size.**
- Therefore methods of SCE5.2.2 (2:1) or combination of SCE5.2.1+SCE5.2.2 (2:1) provide good trade-off between coding efficiency and implementation cost.

## Additional Remark on Complexity

- At the 12th JCT-VC meeting in Geneva, concern was raised that required buffer size of the 2-stage motion data compression is less than the postponing of motion data compression but it requires additional operations for compression.
- According to a software engineer in Sony overhead of this operation can be ignored, as “motion data compression” operation is just decimation of motion data.



# Conclusion

- In this contribution trade-off between buffer size and coding efficiency on the methods related to motion data compression for SHVC is studied.
- It is shown that the methods of SCE5.2.2 (2:1) or SCE5.2.1+SCE5.2.2 (2:1) provide good trade-off between coding efficiency and implementation cost.
- 2-stage motion data compression requires additional operation, but this overhead can be ignored, as the process of motion data compression actually is just decimation.
- It is recommended to consider adoption of SCE5.2.2 (2:1) or combination of SCE5.2.1+SCE5.2.2 (2:1) into SHVC working draft.

								MVsizeInPic x DPBnum		
		W	H	R	D	DPB	BL_mv_buf_size [bit]	EL_mv_buf_size [bit]	total MV data [byte]	diff [byte]
4KSNR	curr. (4:1)	3840	2160	1	4	6	2592000	15552000	2268000	
4KSNR	5:2:2 (2:1)	3840	2160	1	2	6	10368000	15552000	3240000	972000



"SONY" or "make.believe" is a registered trademark and/or trademark of Sony Corporation.

Names of Sony products and services are the registered trademarks and/or trademarks of Sony Corporation or its Group companies.

Other company names and product names are the registered trademarks and/or trademarks of the respective companies.