

JVET-U0091

AHG9/AHG11: SEI message for carriage of neural network information for post filtering

Byeongdoo Choi, Zeqiang Li, Wei Wang, Wei Jiang,
Xiaozhong Xu, Stephan Wenger, Shan Liu



Background

- EE in this meeting reported significant gains with Neural Network based In-loop filtering.
- CE in JVET-O meeting reported considerable gains with Post filtering as well as In-loop.

EE Results on NN-based In-loop filtering
(JVET-U0023)

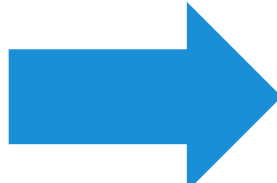
Proposal	Y-PSNR	U-PSNR	V-PSNR	EncT, ×	DecT, ×	Y-PSNR	U-PSNR	V-PSNR	EncT, ×	DecT, ×
JVET-U0060	-1.0%	-3.8%	-3.8%	1.1	35	-1.3%	-2.8%	-3.6%	1.1	33
(JVET-T0057)	-1.3%	-5.8%	-5.9%	1.1	44	-1.6%	-4.5%	-5.4%	1.1	41
JVET-U0074	-0.8%	-7.9%	-5.9%	1.4	243	-2.6%	-10.1%	-9.2%	1.4	303
(JVET-T0069)	-2.2%	-10.4%	-8.6%	1.4	288	-2.9%	-9.3%	-9.1%	1.4	319
	0.1%	-3.7%	-1.7%	1.4	215	-1.4%	-5.7%	-5.7%	1.4	267
	-2.0%	-8.3%	-7.6%	1.4	246	-2.5%	-7.3%	-7.7%	1.3	218
JVET-U0094	-5.3%	-13.1%	-12.9%	1.4	149	-4.8%	-10.1%	-10.6%	1.3	85
(JVET-T0079)	-4.6%	-11.8%	-11.6%	1.2	78	-4.2%	-8.8%	-9.9%	1.1	45
	-4.8%	-11.7%	-11.9%	1.4	148	-4.6%	-9.4%	-10.4%	1.2	85
	-4.9%	-12.7%	-12.6%	2.1	148	-4.4%	-9.9%	-10.7%	1.6	85
	-4.0%	-12.1%	-12.3%	2.1	148	-3.3%	-8.3%	-9.3%	1.6	85
JVET-U0101	-4.1%	-18.5%	-17.8%	1.5	410	-5.3%	-16.7%	-17.7%	1.5	404
(JVET-T0094)	-3.4%	-14.2%	-13.9%	1.2	187	-4.5%	-12.6%	-13.8%	1.3	193
	-3.3%	-15.9%	-15.1%	1.3	212	-4.2%	-14.1%	-14.8%	1.3	224
JVET-U0054										
(JVET-T0128)	-5.0%	-15.9%	-17.1%							

Comparison between In-loop filtering and Post filtering
(JVET-O0030)

AI Over VTM-5.0					
Test#	Y	U	V	EncT	DecT
CE10-2.2a as In-loop filter	-3.93%	-7.89%	-7.29%	134%	74517%
CE10-2.2b as post filter	-3.05%	-9.02%	-10.47%	136%	74083%

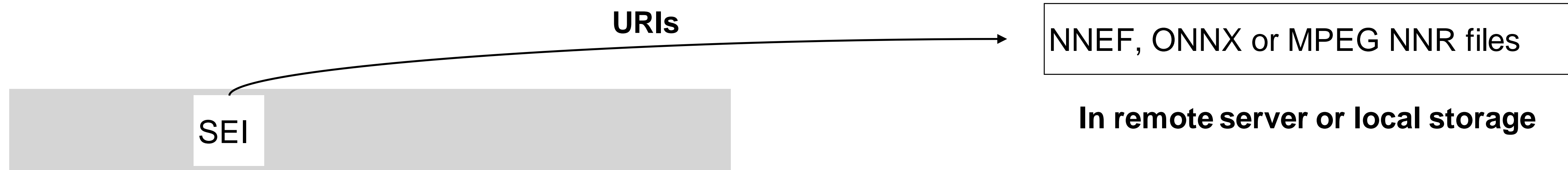
Motivation

- The next versions of VVC may be able to have the best performing NN model with well trained parameters, for in-loop filtering or other normative coding tools.
- On the other hand, it is desirable to have a mechanism to deliver User's proprietary NN models and parameters, for post filtering or other optional processing.
- In order to indicate appropriate NN-based post filtering info for each video sequence or coded picture, two approaches are possible:
 1. NN information are present out of bitstream, but some information in SEI message indicates where the NN information is present.
 2. NN information are contained in the bitstream (in SEI message).

 **A New SEI message having 1, 2 or both information is proposed.**

Design choices of the proposed SEI message

- 1. Indication of external NN model & parameter linkage information.
 - : Network topology and parameters are contained in external files or bitstream.
 - URLs or other external link information are signaled in SEI message.
 - Usually, the external data may be present as NN exchangeable formats like NNEF.

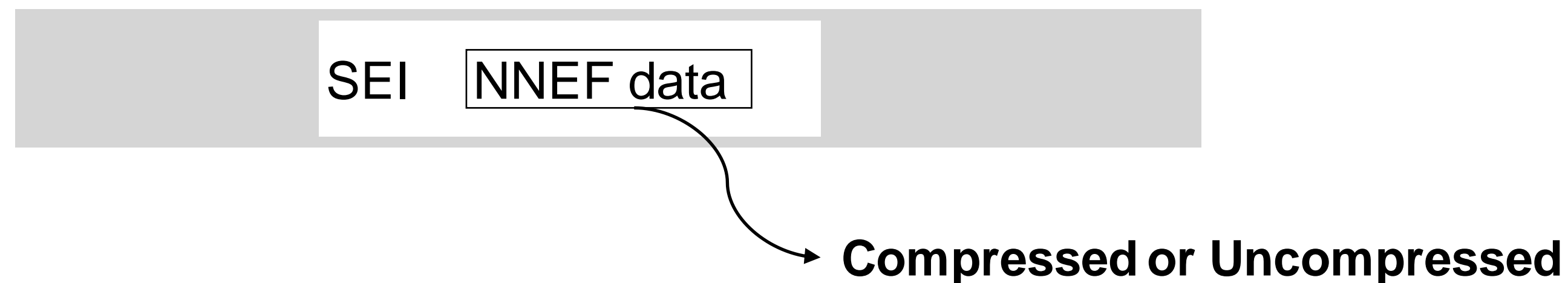


Design choices of the proposed SEI message

■ 2. Signaling of NN model & parameters in SEI message payload.

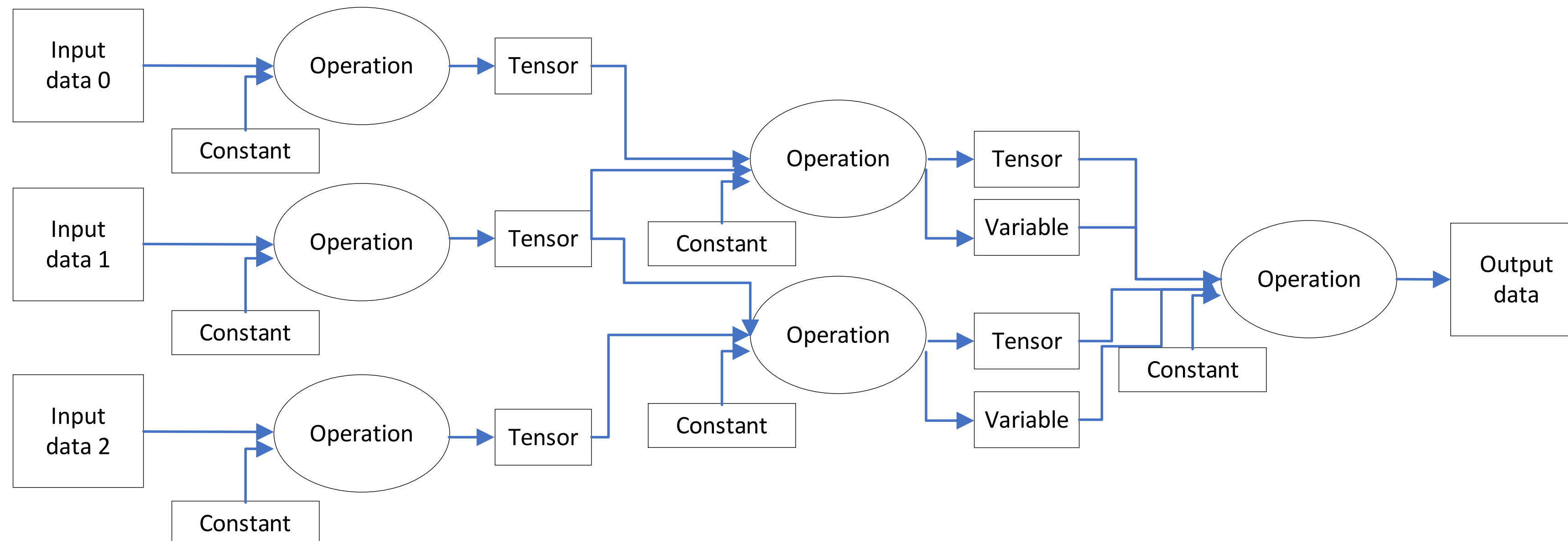
2-1: SEI message contains NN data written with NN exchangeable formats.
(e.g NNEF, ONNX, MPEG-NNR)

2-2: SEI message contains NN data signaled with its own syntax and semantics.
In this case, a lightweight syntax elements to represent NN needs to be developed.
Only simple NN topologies can be represented by the lightweight syntax.
However, most post filtering may use simple topologies based on CNN.



Things to be specified for representation of NN

- Generally, a neural network model can be represented by a computational graph, which is a directed graph with multiple nodes. The nodes consist of operation nodes and data, such as tensor.
- For post-filtering, CNN-based simple models are popularly used. A simple multi-layered feedforward network like CNN can be represented by a linear graph, where each operation node in a layer produces processed data.



Example of SEI syntax structure and elements - 1

postfilter_with_neural_network_info (payloadSize) {	Descriptor
nn_partition_flag	u(1)
nn_topology_info_external_present_flag	u(1)
nn_parameter_info_external_present_flag	u(1)
nn_input_pic_format_present_flag	u(1)
nn_output_pic_format_present_flag	u(1)
nn_postfilter_type_idc	u(3)
num_nn_input_ref_pic	ue(v)
if(nn_partition_flag) {	
num_partitioned_nn_sei_messages	ue(v)
nn_sei_message_idx	ue(v)
}	
if(network_topology_info_external_present_flag) {	
external_nn_topology_info_format_idc	u(4)
num_bytes_external_nn_topology_uri_info	ue(v)
external_nn_topology_uri_info	u(v)
}	
else {	
network_topology_info(input)	
}	
if(network_parameter_info_external_present_flag) {	
external_network_parameter_info_format_idc	u(4)
num_bytes_external_network_parameter_uri_info	ue(v)
external_network_parameter_uri_info	u(v)
}	
else {	
network_parameter_info(input)	
}	

Flag indicates whether NN info is partitioned into multiple SEIs or not

Partitioning information

Linkage info to external NN topology data

or

Internal NN topology data in NNEF, ONNX, MPEG-NNR format or new syntax elements

Linkage info to external NN parameter data

Internal NN parameter data in NNEF, ONNX, MPEG-NNR format or new syntax elements

Example of SEI syntax structure and elements - 2

if(network_input_pic_format_present_flag) {	
nn_input_chroma_format_idc	u(2)
nn_input_bitdepth_minus8	ue(v)
nn_input_pic_width	ue(v)
nn_input_pic_height	ue(v)
nn_patch_size_present_flag	u(1)
if(nn_patch_size_present_flag) {	
nn_input_patch_width	ue(v)
nn_input_patch_height	ue(v)
nn_boundary_padding_idc	u(2)
}	
}	
if(num_network_input_ref_pic > 0) {	
num_fwd_ref_pics_as_input	u(4)
if (NumFwdRefPics > 0) {	
nearest_fwd_ref_pics_used_flag	u(1)
for(i = 0; i < NumFwdRefPics && !nearest_fwd_ref_pics_used_flag; i++) {	
poc_dist_fwd_ref_pic[i]	ue(v)
}	
}	
if (NumBwdRefPics > 0) {	
nearest_bwd_ref_pics_used_flag	u(1)
for(i = 0; i < NumBwdRefPics && !nearest_bwd_ref_pics_used_flag; i++) {	
poc_dist_bwd_ref_pic[i]	u(6)
}	
}	
}	
}	

Picture format of NN input data

Forward and backward reference picture indications for super-resolution processing

Example of SEI syntax structure and elements - 3

network_topology_info(input) {	Descriptor
nn_topology_storage_format_idc	u(4)
nn_topology_compression_format_idc	u(4)
num_bytes_topology_data	ue(v)
if(nn_topology_format_idc>0) {	
for(I = 0; I < num_bytes_topology_data; i++) {	
nn_topology_data_byte[I]	b(8)
}	
}	
else {	
// The following syntax elements are hypothetical examples to show how CNN can be desribed	
num_variables	ue(v)
for(i = 0; i < num_variables; i++)	
define_variable(i)	
num_operation_nodes_types	ue(v)
for(i = 0; i < num_operation_nodes_types; i++)	
define_operation_node(i)	
num_operation_node_executions	ue(v)
for(i = 0; i < num_operation_node_executions; i++)	
operation_node_excution(i)	
}	
}	

define_variable(i) {	Descriptor
nn_variable_class_idc[i]	u(4)
nn_variable_type_idc[i]	u(4)
nn_variable_dimensions[i]	ue(v)
for(j = 0; j < num_bytes_topology_data; j++) {	
nn_variable_dimension_size[i][j]	ue(v)
}	
...	
}	

define_operation_node(i) {	Descriptor
nn_operatoin_class_idc[i]	u(8)
nn_operation_function_idc[i]	u(8)
num_input_variables[i]	ue(v)
num_output_variables[i]	ue(v)
...	
}	

operation_node_execution(i) {	Descriptor
nn_op_node_idx[i]	u(v)
for(j = 0; j < num_input_variables[nn_op_node_idx[i]]; j++) {	
nn_input_variable_idx[i][j]	u(v)
}	
for(j = 0; j < num_output_variables[nn_op_node_idx[i]]; j++) {	
nn_output_variable_idx[i][j]	u(v)
}	
...	
}	

Conclusion

- It is not a proposal with a completed solution, but a trigger to have collaboration to develop the 1st NN-based video coding/processing syntax and specification.
- The design intention aims a flexible representation of NN-based post filtering (in SEI message), with internal and external carriage of NN data.
- Further studies are suggested, more HLS and NN experts are welcome to join.

Thanks!
Q&A