

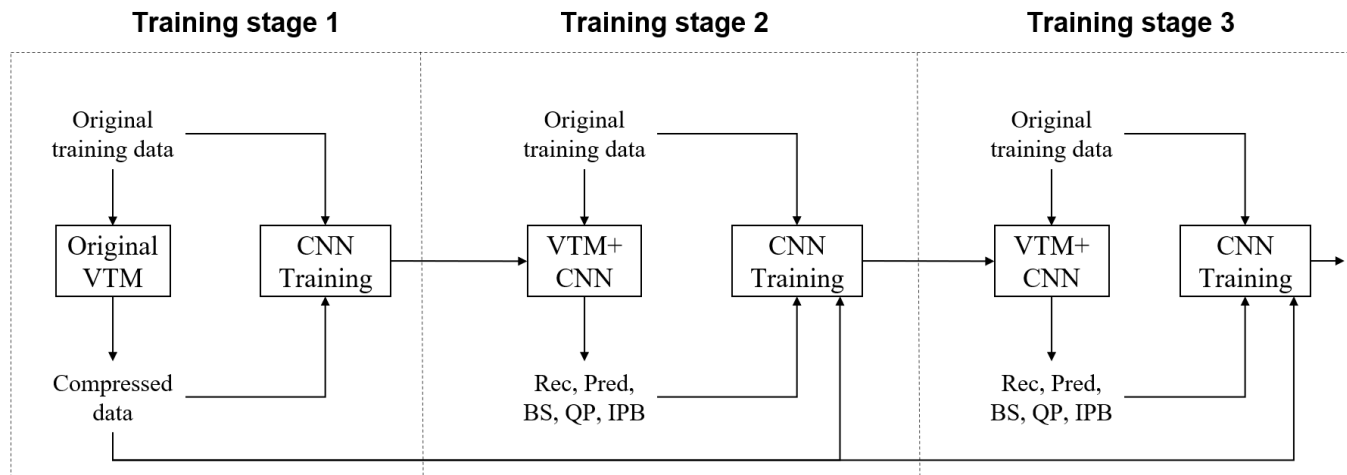
AHG11: QP Distance-Based Progressive Learning for Enhancing HOP In-Loop Filter

Penghao Fu, Cheolkon Jung, and Qipu Qin

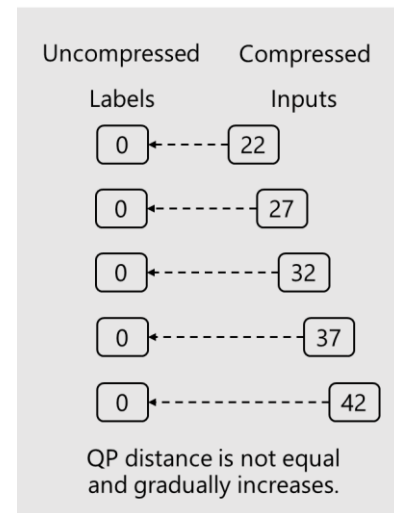
School of Electronic Engineering

Xidian University, China

Motivation: Imbalanced QP Gap



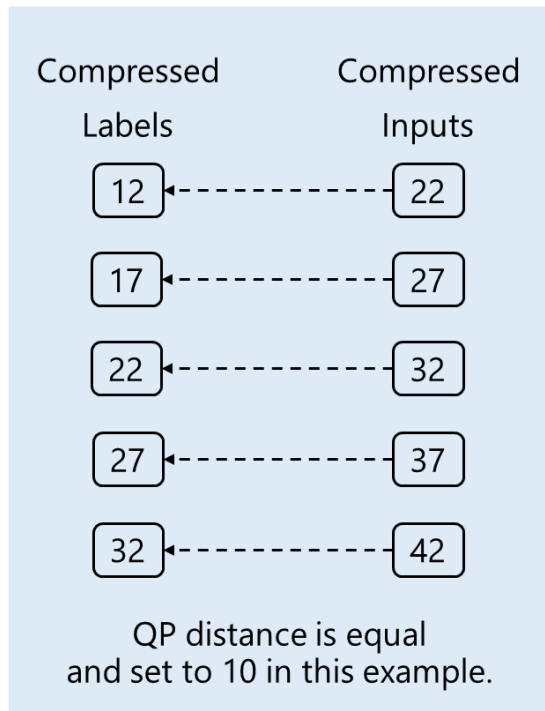
Training pipeline for HOP model generation in JVET-AE0191



Imbalanced QP Gap

- JVET suggests a three-stage training strategy to train HOP in-loop filter in JVET-AE0191.
- If the input is a compressed video frame with a large QP, the feature information contained in the compressed video frame is limited, which has too **much gap from its label** from the uncompressed video frames to reconstruct.
- Moreover, the uncompressed video frames do not contain compression artifacts so that the label from the uncompressed video frames are not effective in capturing the relationship between the compressed input and its uncompressed label.

Motivation: QP Distance



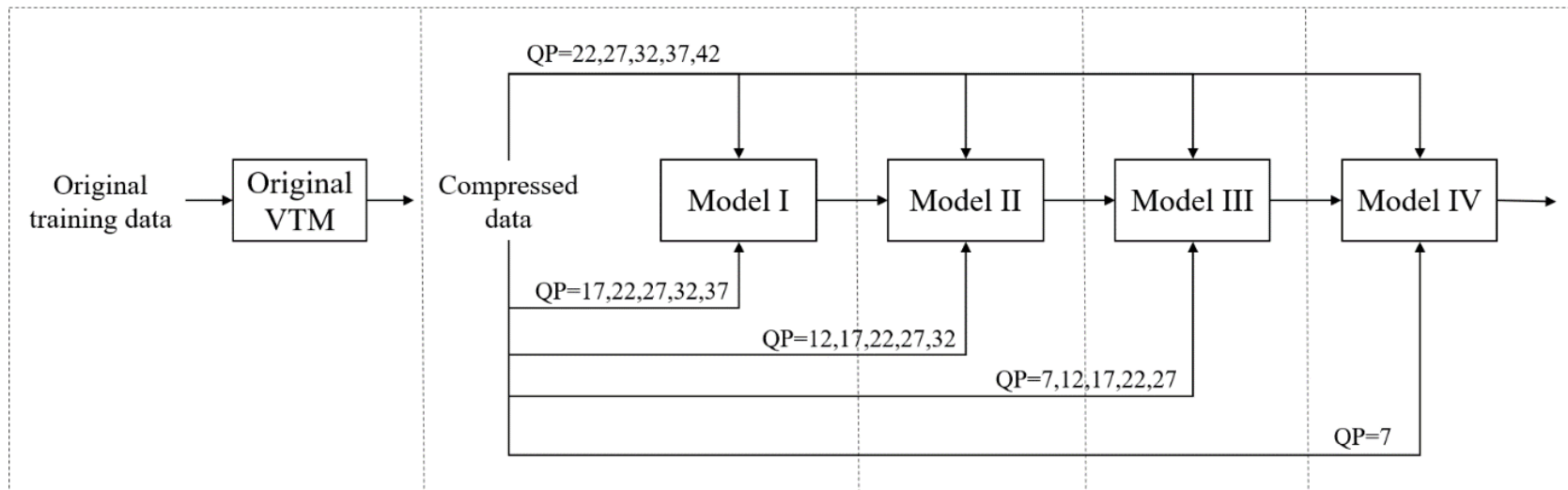
QP distance

- The QP distance-based training strategy takes compressed data at lower QP as label for training rather than using the uncompressed data as label.
- The QP distance-based training strategy can address the imbalanced QP gap between the compressed input and its uncompressed label, while strengthening the HOP learning ability.

Progressive Learning Based on QP Distance

Training data generation

Progressive learning based on QP distance

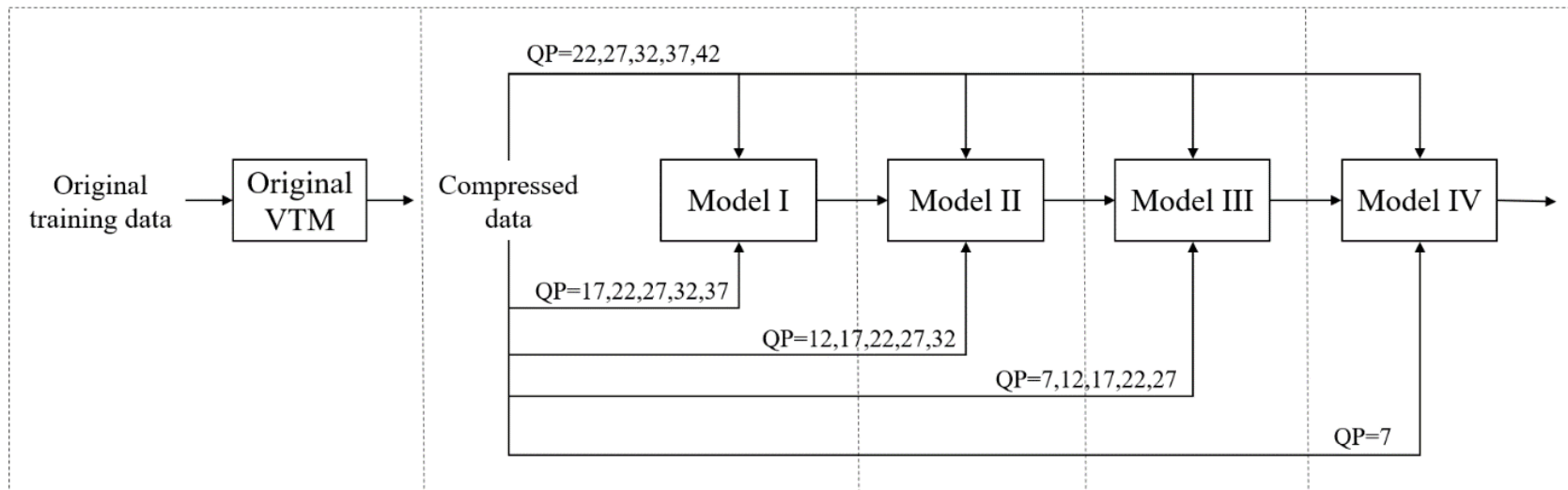


- To enhance the HOP in-loop filter, this contribution introduces QP distance in the HOP model training.
- This contribution proposes progressive learning based on QP distance to address the imbalanced QP gap between the compressed input and its label while strengthening the HOP learning ability.

Progressive Learning Based on QP Distance

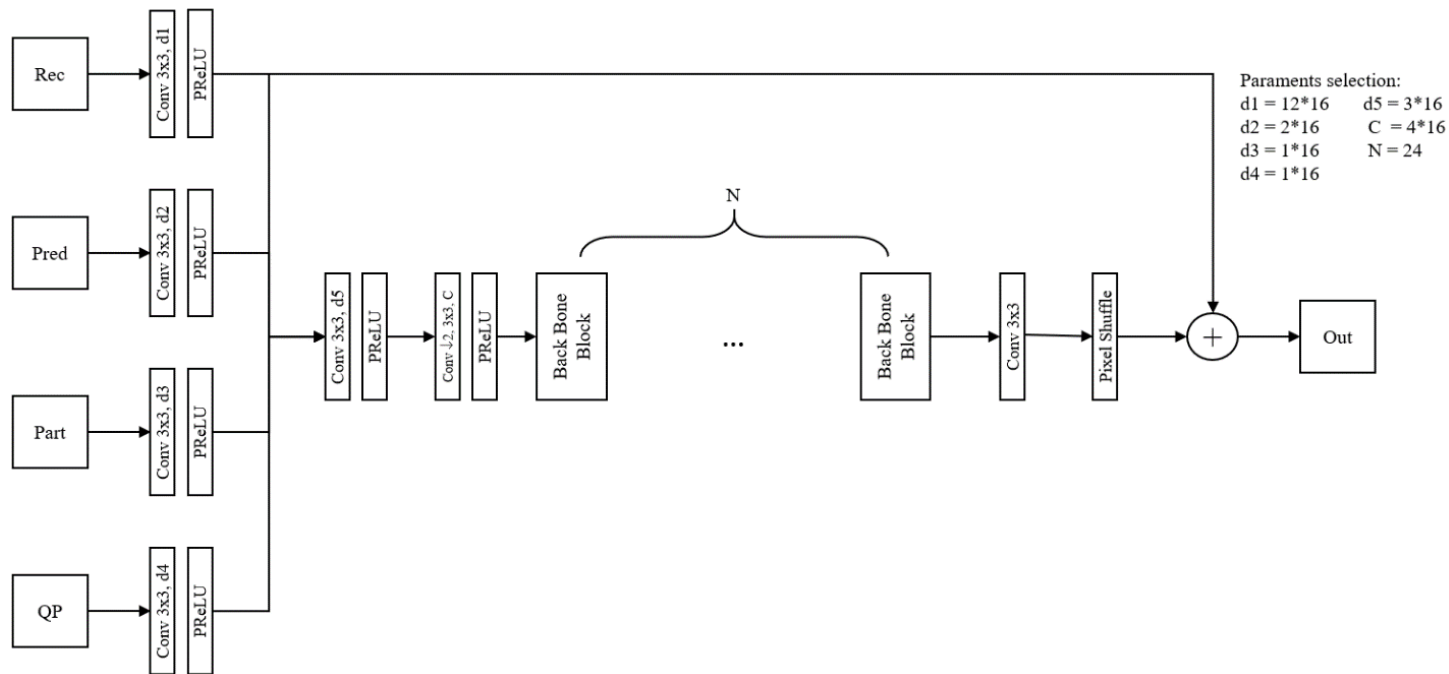
Training data generation

Progressive learning based on QP distance



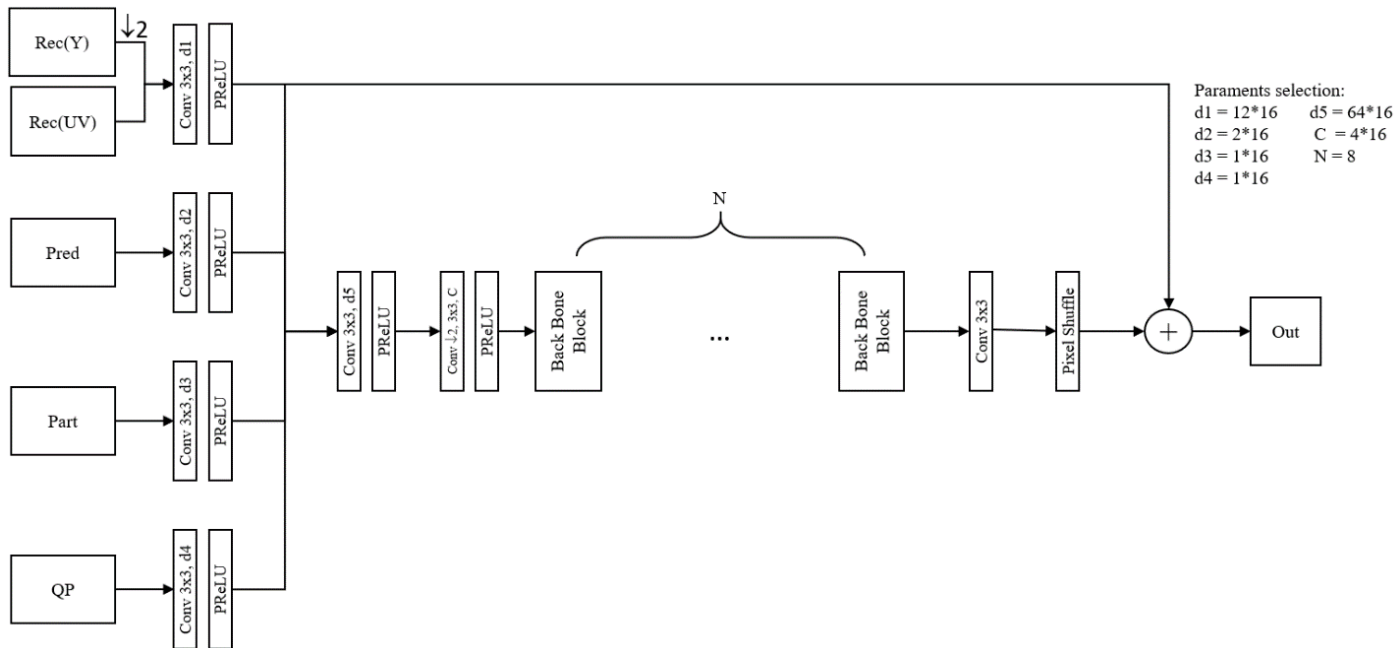
- When setting the QP distance to 5, the input QPs for the training set are set to {22, 27, 32, 37, 42}, while the corresponding label QPs are set to {17, 22, 27, 32, 37}. Subsequently, in the next training stage, the model from the previous stage is loaded, and the QP distance for the training set is increased. In the first three stages, the QP distance increases by 5, while in the final stage, the label QPs are all set to 7.

HOP Model for Luma



- We set the input as reconstructed frame (Rec), predicted frame (Pred), partition map (Part), and QP map (QP).
- The remaining structures are not modified, and the trained HOP model is embedded into VTM-11.0_NNVC-2.0 for evaluation.

HOP Model for Chroma



- Similarly, we set the input as the luma reconstructed frame (Rec(Y)), the chroma reconstructed frame (Rec(UV)), predicted frame (Pred), partition map (Part), and QP map (QP).
- The remaining structures are not modified, and the trained HOP model is embedded into VTM-11.0_NNVC-2.0 for evaluation.

Network Information in Training Stage

<u>Network Information in Training Stage</u>		
Mandatory	GPU Type	GeForce RTX 4090
	Framework:	PyTorch 2.0.1
	Number of GPUs per Task	1
	Epoch:	120 per stage (total 4*120)
	Batch size:	32
	Training time:	~50h/luma model, ~11h/chroma model
	Training data information:	BVI-DVC, DIV2K
	Training configurations for generating compressed training data (if different to VTM CTC):	VTM-11.0, QP {7,12,17,22, 27, 32, 37, 42}
	Loss function:	L1 (first 90 epochs) and L2 (last 30 epochs)
Optional		
	Number of iterations	12260
	Patch size	144x144
	Learning rate:	1e-4
	Optimizer:	ADAM
	Preprocessing:	
	Other information:	

Network Information in Inference Stage

Network Information in Inference Stage		
Mandatory	HW environment:	
	GPU Type	CPU only
	Framework:	Libtorch v1.9.0
	Number of GPUs per Task	0
	Number of Parameters (Each Model)	1.44 M for luma, 0.76 M for chroma
	Total Number of Parameters (All Models)	2.2M (2 models)
	Parameter Precision (Bits)	32
	Memory Parameter (MB)	6.4M
	Multiply Accumulate (MAC)/pixel	371K for luma, 259 K for chroma
Optional		
	Total Conv. Layers	
	Total FC Layers	
	Total Memory (MB)	
	Batch size:	1
	Patch size	144x144
	Changes to network configuration or weights required to generate rate points	
	Peak Memory Usage (Total)	
	Peak Memory Usage (per Model)	
	Border handling	
	Other information:	

Experimental Results

In the first training stage (**Model I**), the QP distance is set to 5, and the training set's input QPs are set to {22, 27, 32, 37, 42}, with corresponding label QPs set as {17, 22, 27, 32, 37}.

BD-rate of Model I under AI configuration

	All Intra Main10					
	BD-rate Over VTM-11.0					
	Y-PSNR	U-PSNR	V-PSNR	Y-MSIM	U-MSIM	V-MSIM
Class A1	-5.99%	-15.29%	-17.12%	-5.73%	-18.42%	-19.22%
Class A2	-6.23%	-15.20%	-14.06%	-6.74%	-16.50%	-13.90%
Class B	-6.18%	-15.50%	-16.51%	-6.20%	-17.60%	-18.48%
Class C	-6.51%	-14.70%	-15.97%	-6.21%	-17.34%	-18.24%
Class E	-9.39%	-18.99%	-18.49%	-9.50%	-19.13%	-21.38%
Overall	-6.77%	-15.82%	-16.41%	-6.76%	-17.75%	-18.27%
Class D	-6.20%	-13.89%	-15.82%	-5.30%	-16.08%	-17.78%

Experimental Results

In the second training stage (**Model II**), the QP distance is set to 10, and the training set's input QPs are set to {22, 27, 32, 37, 42}, with corresponding label QPs set as {12, 17, 22, 27, 32}.

BD-rate of Model II under AI configuration

	All Intra Main10					
	BD-rate Over VTM-11.0					
	Y-PSNR	U-PSNR	V-PSNR	Y-MSIM	U-MSIM	V-MSIM
Class A1	-7.05%	-15.55%	-17.12%	-6.77%	-17.96%	-20.18%
Class A2	-7.22%	-17.18%	-17.11%	-7.80%	-18.83%	-16.88%
Class B	-7.19%	-15.00%	-16.62%	-7.20%	-18.33%	-20.16%
Class C	-7.68%	-15.92%	-17.90%	-7.45%	-19.23%	-21.14%
Class E	-10.67%	-20.92%	-21.53%	-10.76%	-20.78%	-24.17%
Overall	-7.86%	-16.65%	-17.89%	-7.88%	-18.96%	-20.50%
Class D	-7.45%	-15.41%	-17.73%	-6.53%	-19.11%	-20.56%

Experimental Results

In the third training stage (**Model III**), the QP distance is set to 15, and the training set's input QPs are designed as {22, 27, 32, 37, 42}, with corresponding label QPs set as {7, 12, 17, 22, 27}.

BD-rate of Model III under AI configuration

	All Intra Main10					
	BD-rate Over VTM-11.0					
	Y-PSNR	U-PSNR	V-PSNR	Y-MSIM	U-MSIM	V-MSIM
Class A1	-7.37%	-15.57%	-17.07%	-7.22%	-18.19%	-20.14%
Class A2	-7.52%	-17.67%	-17.21%	-8.26%	-19.55%	-17.29%
Class B	-7.52%	-14.29%	-16.74%	-7.64%	-18.65%	-20.97%
Class C	-8.06%	-15.55%	-18.22%	-7.89%	-19.54%	-21.88%
Class E	-11.02%	-21.43%	-21.76%	-11.16%	-21.63%	-24.58%
Overall	-8.20%	-16.54%	-18.04%	-8.32%	-19.42%	-21.02%
Class D	-7.87%	-15.49%	-18.12%	-7.04%	-20.36%	-21.68%

Experimental Results

In the final training stage (**Model IV**), the training set's input QPs are designed as {22, 27, 32, 37, 42}, with corresponding label QPs are all set to {7, 7, 7, 7, 7}.

BD-rate of Model IV under AI configuration

	All Intra Main10					
	BD-rate Over VTM-11.0					
	Y-PSNR	U-PSNR	V-PSNR	Y-MSIM	U-MSIM	V-MSIM
Class A1	-7.39%	-15.40%	-17.67%	-7.33%	-18.35%	-20.43%
Class A2	-7.63%	-18.03%	-17.31%	-8.40%	-20.23%	-17.19%
Class B	-7.62%	-13.98%	-16.98%	-7.76%	-18.50%	-21.21%
Class C	-8.24%	-15.45%	-18.49%	-8.04%	-19.57%	-22.10%
Class E	-11.14%	-20.37%	-21.69%	-11.28%	-20.67%	-24.01%
Overall	-8.31%	-16.28%	-18.27%	-8.45%	-19.36%	-21.08%
Class D	-8.05%	-15.35%	-17.96%	-7.24%	-20.65%	-21.76%

Experimental Results

BD-rate of the three-stage training strategy (JVET-AE0191) under AI configuration

	Y-PSNR	U-PSNR	V-PSNR
Class A1	-7.05%	-17.89%	-20.06%
Class A2	-6.97%	-19.25%	-16.77%
Class B	-7.00%	-18.24%	-19.89%
Class C	-7.95%	-18.65%	-21.27%
Class E	-10.39%	-20.44%	-21.57%
Overall	-7.78%	-18.81%	-19.98%
Class D	-7.83%	-17.73%	-21.23%
Class F	-5.53%	-16.45%	-16.17%
Class H	#VALUE!	#VALUE!	#VALUE!

Interdigital

	Y-PSNR	U-PSNR	V-PSNR
Class A1	-7.10%	-17.83%	-21.74%
Class A2	-6.89%	-18.32%	-16.19%
Class B	-6.96%	-17.82%	-18.90%
Class C	-8.02%	-18.38%	-20.81%
Class E	-10.52%	-20.30%	-21.33%
Overall	-7.80%	-18.44%	-19.75%
Class D	-7.89%	-17.40%	-20.71%
Class F	-5.42%	-16.03%	-16.28%
Class H	#VALUE!	#VALUE!	#VALUE!

OPPO

	Y-PSNR	U-PSNR	V-PSNR
Class A1	-7.25%	-17.84%	-20.96%
Class A2	-6.99%	-19.16%	-16.83%
Class B	-7.01%	-18.24%	-19.86%
Class C	-7.95%	-18.37%	-20.97%
Class E	-10.39%	-20.39%	-21.56%
Overall	-7.82%	-18.72%	-20.07%
Class D	-7.82%	-17.82%	-20.98%
Class F	-5.52%	-16.16%	-15.92%
Class H	#VALUE!	#VALUE!	#VALUE!

Bytedance

Conclusion

- This contribution proposes progressive learning based on QP distance for enhancing HOP in-loop filter.
- Experimental results show that in the AI configuration, compared with VTM-11.0_NNVC-2.0, the HOP model trained by the proposed training strategy achieves an average BD rate reduction of {8.31% (Y), 16.28% (U), 18.27% (V)}.
- It should be noted that the QP-distance based training strategy is not only applicable to the training of NNLFs, but also widely extended to various NN-based image/video coding models, such as intra-frame prediction, inter-frame prediction, and super-resolution.
- **Recommendation:**
 - It is recommended to navigate the QP distance-based training strategy in EE1 and include it as a training method in CTC.



THANK YOU!

