JVET-X-0043

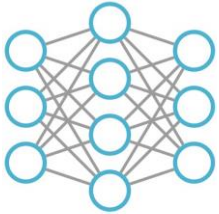
[AHG11 & AHG6] DOVC: Deep Omnidirectional Video Compression

Qipu Qin[1], Cheolkon Jung[1], Dan Zou[2], and Ming Li[2]

[1]Xidian University, China

[2]OPPO, China

Introduction



■ Background

- Deep learning-based video compression (DLVC or DVC) has achieved great advances in improving coding efficiency.
- DLVC/DVC can be summarized into the following two aspects:
  - (1) Combine deep learning with traditional hybrid video compression.
  - (2) Establish a novel deep video compression framework.

■ Motivation

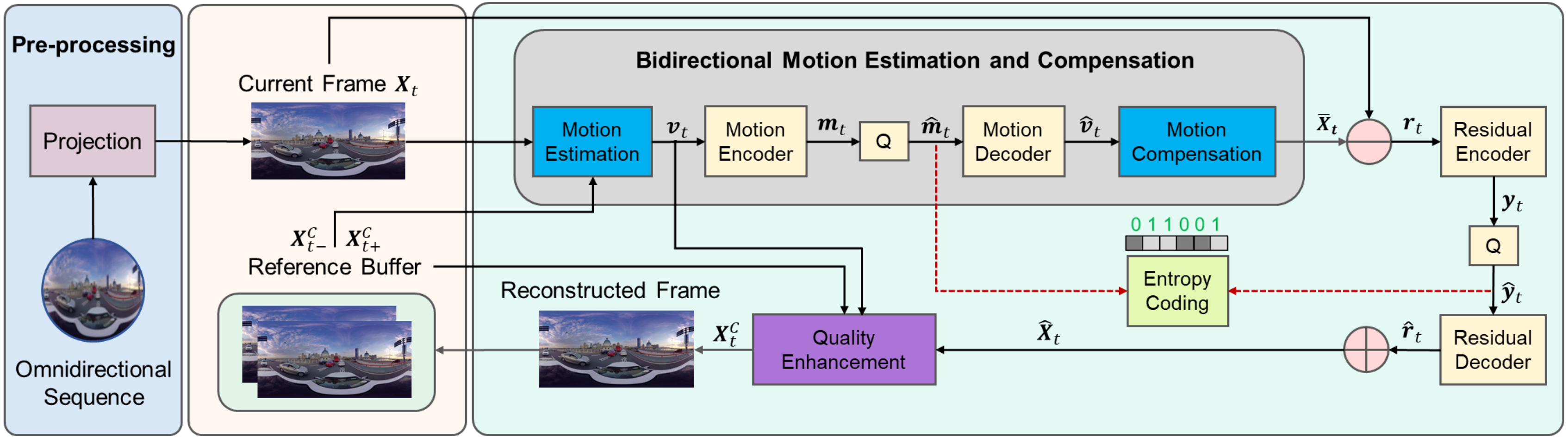
- Without considering omnidirectional videos.
- Limited to the use of optical flow network.
- High complexity and local optimization.

Proposed Solution

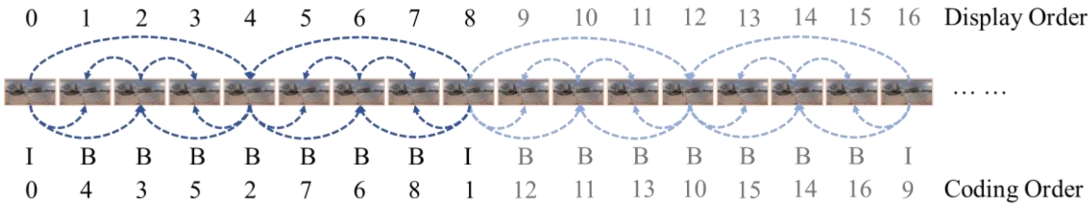
Architecture

DOVC mainly contains:

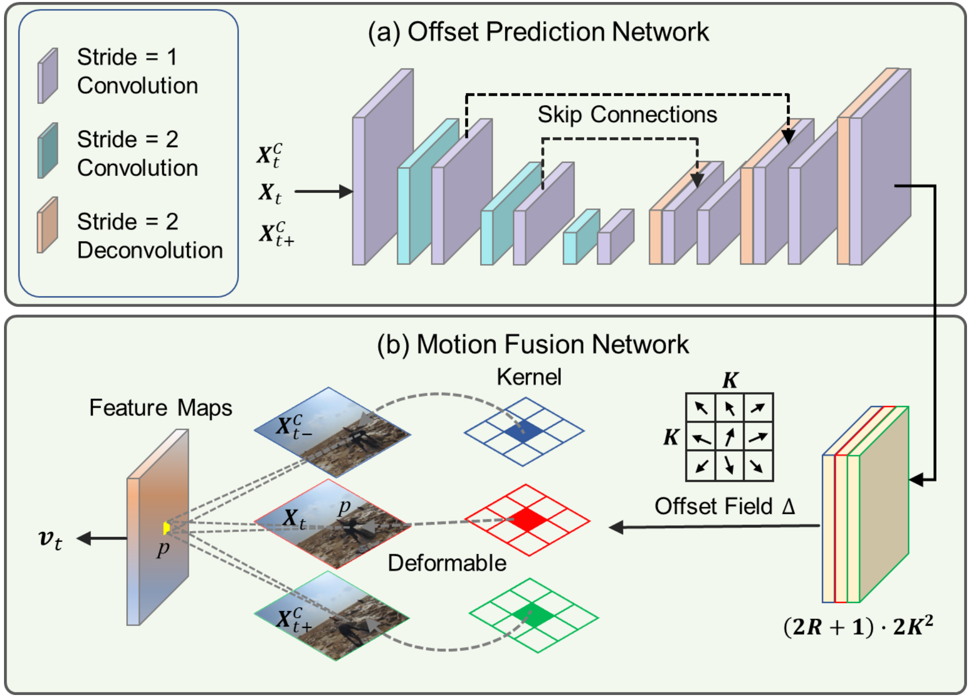
- Projection
- Bidirectional motion estimation
- Motion encoder/decoder
- Bidirectional motion compensation
- Residual encoder/decoder
- Quality enhancement
- Entropy coding



Bidirectional Motion Estimation

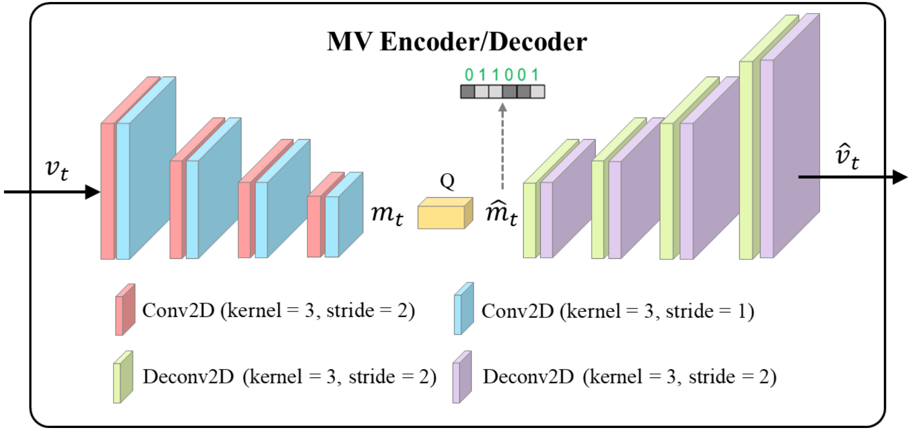


- Offset prediction network
- Offset field Δ
- Motion fusion : Deformable convolution
- Motion vector

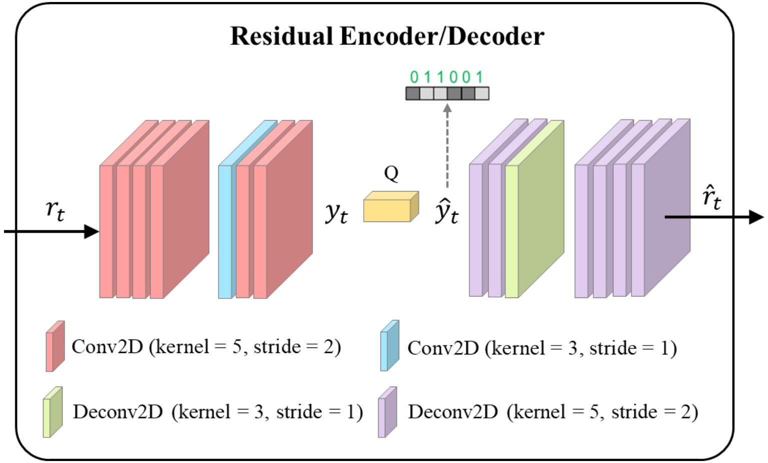


Motion and Residual Encoder/Decoder

- Auto-encoder style network to encode/decode motion vector and residual information.
- Quantization.
- $\hat{m}_t$ and $\hat{y}_t$ are sent to the entropy coding module to write the bit-stream.



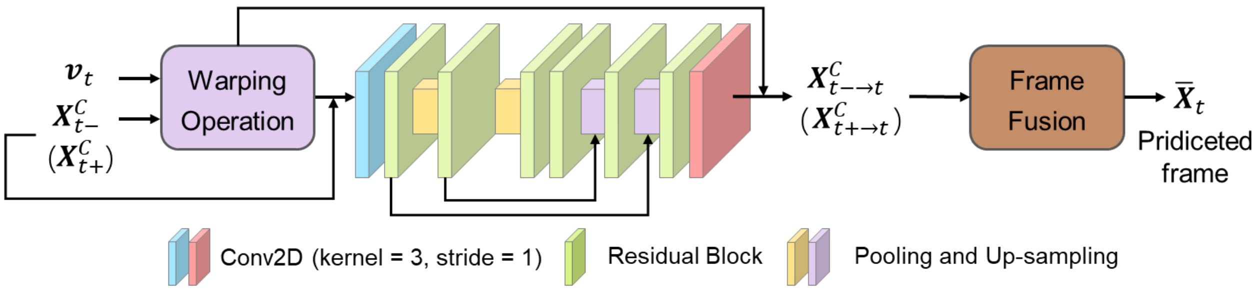
Motion encoder/decoder

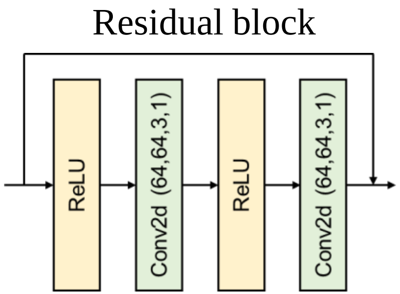


Residual encoder/decoder

Bidirectional Motion Compensation

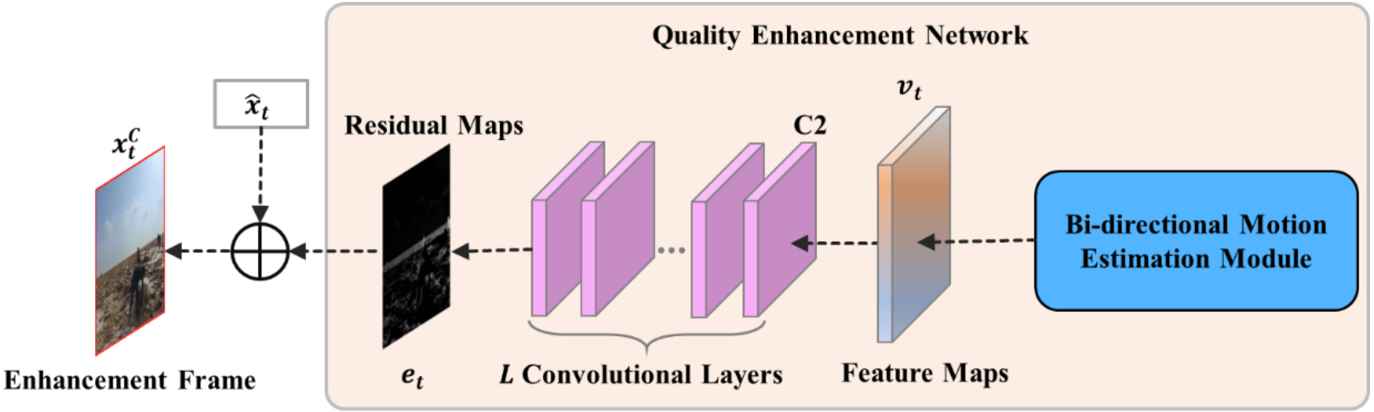
- Bidirectional prediction mode.
- Warping operation.
- Convolutional layers (2) + Residual blocks (6).
- Frame fusion.





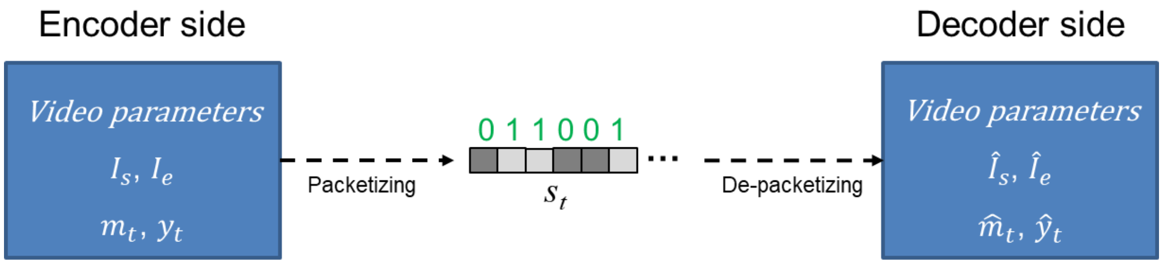
Quality Enhancement

- Reuse temporal information from motion estimation.
- Output enhanced residual map.
- Plain but effective.



Entropy Coding

- For I frame, DOVC adopts BPG tool for image compression and decompression.
- For B/P frame, DOVC decoder utilizes the arithmetic coder based on neural networks.
- The library encodes a feature map into a bitstream or decode a bitstream into a feature map.



Projection and Loss Function

- Pre-processing stage: Sphere-to-plane projection.
- The most popular formats: ERP and CMP.
- The weighted factors of ERP and CMP.
- Loss Function.

$$\boldsymbol{w}_{erp}(i,j) = \cos\left(\left(j - \frac{Height}{2} + \frac{1}{2}\right) \cdot \frac{\pi}{Height}\right)$$

$$\boldsymbol{w}_{cmp}(i,j) = \left(3 + \frac{(i+1)^2 + (j+1)^2 - (i+j)\cdot a}{a^2/4}\right)^{-3/2}$$

$$\boldsymbol{W}(i,j) = \frac{\boldsymbol{w}(i,j)}{\sum_{i=0}^{Width-1}\sum_{j=0}^{Height-1}\boldsymbol{w}(i,j)}$$

$$WMSE = \sum_{i=0}^{Width-1}\sum_{j=0}^{Height-1}\left(\boldsymbol{X}(i,j) - \hat{\boldsymbol{X}}'(i,j)\right)^2 \cdot \boldsymbol{W}(i,j)$$

$$\boxed{\min\left\{J = \boldsymbol{R} + \lambda \cdot WMSE\left(\boldsymbol{X}_t, \hat{\boldsymbol{X}}_t\right)\right\}}$$

Loss Function

Experimental Results

Experimental Setup

- Training environment: Nvidia Tesla V100 (32G), Python3.6, PyTorch1.6, CUDA10.0.
- Test environment: GeForce GTX1080Ti (12G), Python3.6, PyTorch1.6, CUDA10.1.

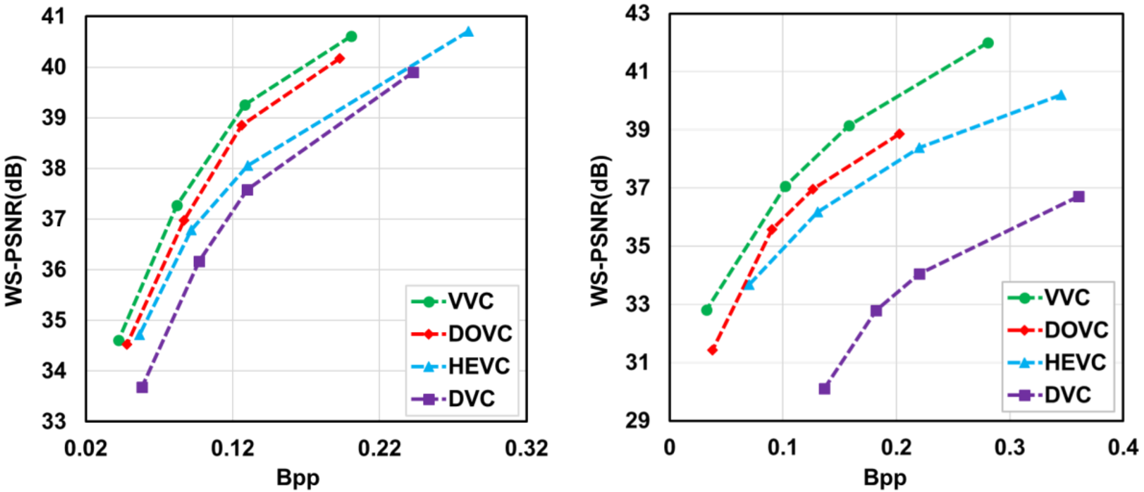
- Anchor environment: Intel Xeon CPU (Dual processor, RAM 32.G), HM-16.16 (with 360Lib-5.0), VTM-11.0 (with 360Lib-12.0), Visual Studio2013.

- Training dataset: VQA-ODV[1] .
- Test dataset: 360-degree sequences provided by JVET.

- Other details: Epoch = 100, Batch Size = 4, Learning Rate = 1e-4, Patch Size = 1280*1280, Optimizer = ADAM, $\lambda = 256, 512, 1024, 2048$.

[1] Proc. ACM Multimedia

Objective Metrics Comparison



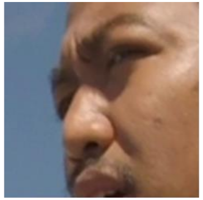
Rate-distortion performance in terms of WS-PSNR.

BD-BR (%) and BD-WS_PSNR (dB) performances of DOVC method in comparison with HEVC/H.265 and DVC.

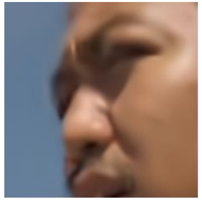
Term Name	DOVC vs HEVC/H.265		DOVC vs DVC	
Projection Format	CMP	ERP	CMP	ERP
BD-BR (%)	-14.2889	-27.7762	-27.7760	-58.8419
BD-WS_PSNR (dB)	0.4837	0.7597	1.2531	3.9060

Visual Comparison

(a) SkateboardInLot (frame = 158)

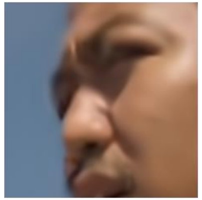


(b) Raw



(c) DOVC
0.07450 bpp, 37.02dB

(d) DVC
0.07975 bpp, 36.68dB



(e) HEVC/H.265
0.07621 bpp, 37. 08dB

Visual comparison in SkateboardInLot (CMP format)

Ablation Study

- Analysis of Motion Estimation and Compensation Modules

(a) 5^{th} Frame (Raw)

(b) Fused Offset map (DOVC)

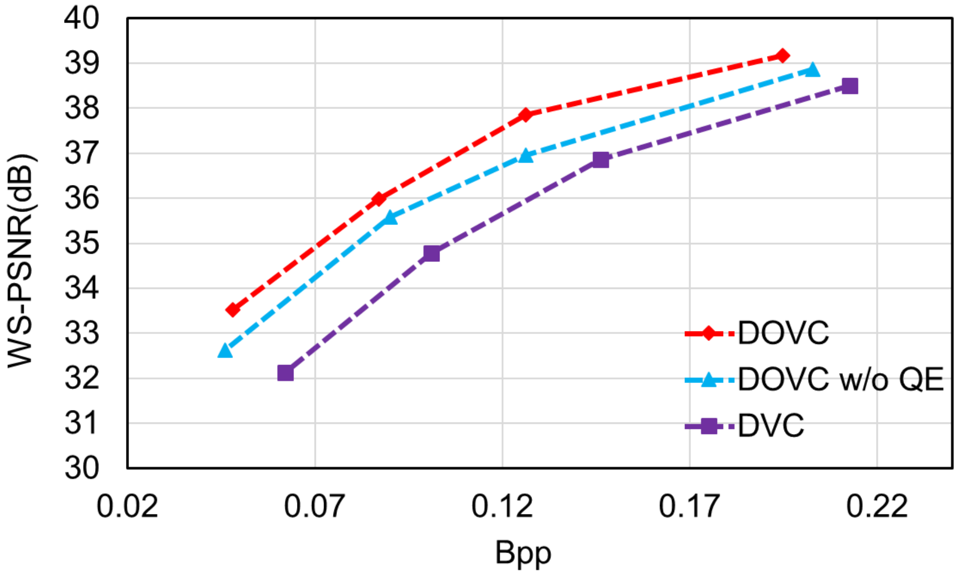
(c) Optical Flow map

(d) Bidirectional compensation (DOVC): 32.5dB

(e) Unidirectional compensation (Optical Flow): 31.7dB

Visualization results on SkateboardInLot (CMP format) of ablation studies for motion estimation and motion compensation.

- Analysis of Quality Enhancement Module



Ablation study on the quality enhancement in DOVC.

The quality enhancement module is removed in DOVC. The results of DOVC without QE drop nearly 0.23dB when compared with the complete DOVC model. Although the quality enhancement module is removed, the performance of DOVC without QE still surpasses DVC.

- Comparison of Coding Time Complexity

Comparison of coding time complexity of HEVC, VVC and DOVC

Class	Sequences	Coding Time Complexity Comparison (s)		
		VVC	HEVC	DOVC
S1	ChairliftRide	238798.367	84503.725	1512.226
	Gaslamp	70911.438	32773.284	1475.378
	Harbor	133933.894	48565.107	1522.904
	KiteFlite	363812.892	104027.615	1506.412
	SkateboardInLot	183657.670	65565.102	1460.329
	Trolley	90323.569	41697.319	1494.871
S2	Balboa	629755.557	112035.185	1456.837
	BranCastle2	162387.403	52067.384	1488.358
	Broadway	639893.832	124027.615	3117.108
	Landings2	89323.569	41565.107	1468.843
Average		260279.819	70682.744	**1650.327**

Conclusion

- **Contribution:** An end-to-end deep omnidirectional video compression framework (DOVC) with CNNs.

- **BD-BR and BD-WS_PSNR:** DOVC achieves average 21% reduction in BD-BR and average 0.6217dB gain in BD-WS_PSNR over HM-16.16 (360Lib-5.0) under LDP configuration for encoding omnidirectional videos.

- **Coding time of DOVC:** only 0.0234 times that of HM-16.16 (360Lib-5.0) and 0.0064 times that of VTM-11.0 (360Lib-12.0).

Recommendation to JVET:

- Omnidirectional videos with much higher resolution (6K and 8K), it is necessary to explore an efficient compression framework for them.

- Deep learning shows outstanding non-linear fitting ability, which can be successfully applied to the omnidirectional video compression.

- Therefore, we propose a new **EE on coding omnidirectional videos** using deep NNs or including this topic in an existing EE. We recommend JVET to consider investigating this topic for further research.

THANK YOU!