

AHG 16: Updated Common Software Tools for Generative Face Video Compression

Bolin Chen, Yan Ye, Goluck Konuko, Giuseppe Valenzise, Shanzhi Yin and Shiqi Wang

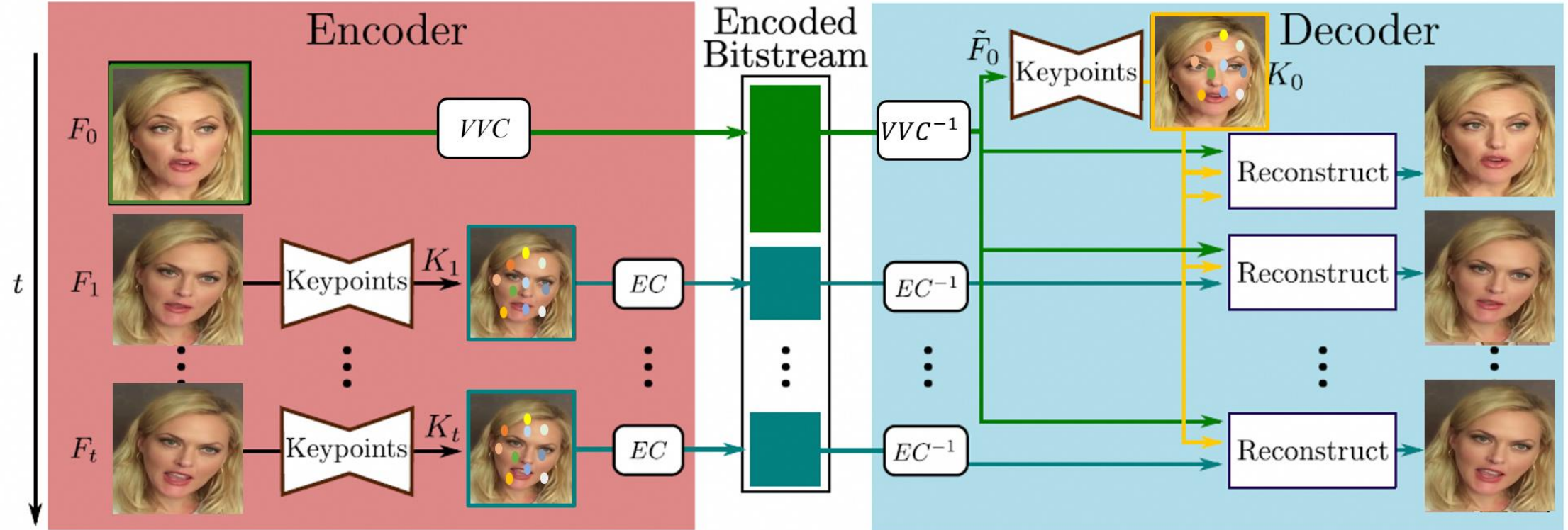
Alibaba Group, CNRS/CentraleSupélec/Université Paris-Saclay and City University of Hong Kong

April. 2024

Proposal Review & Contribution

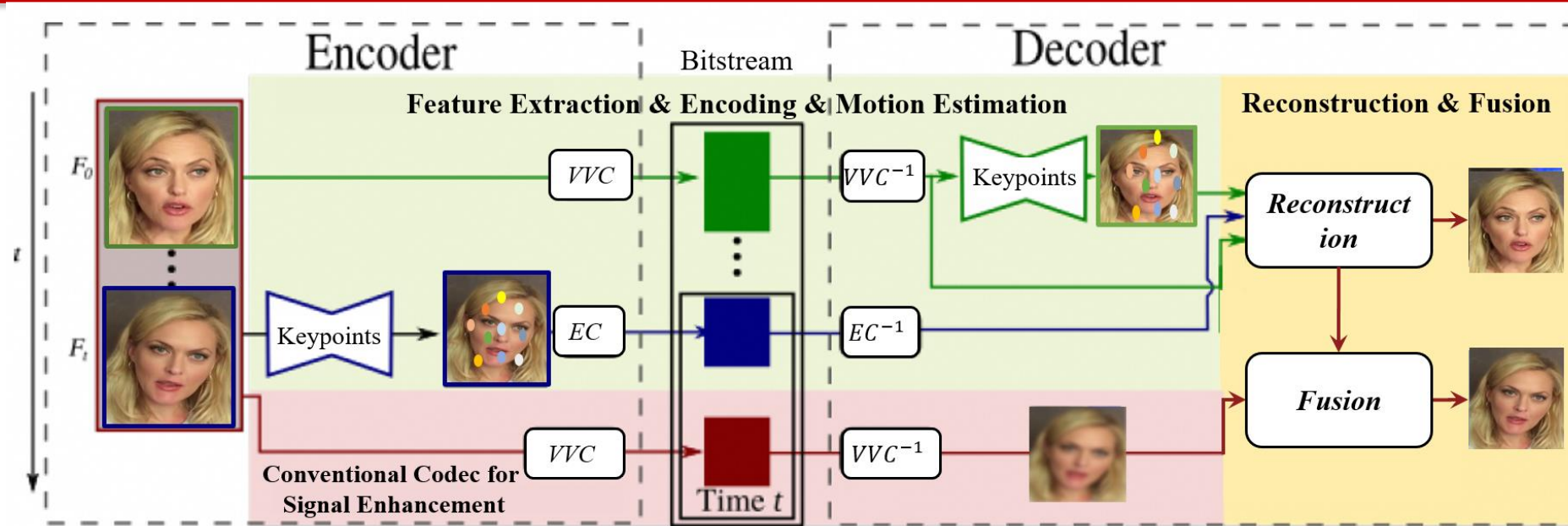
- ❑ **Review:** At the 33th JVET meeting, **a unified software package as well as test conditions** for generative face video coding (GFVC) were established for **JVET AHG16 experimentation**.
- ❑ **Problem:** It has been observed that the current GFVC software can only achieve promising face video compression in ultra-low bitrate ranges, and **cannot provide a more comprehensive coverage for mid-to-high bitrate ranges**.
- ❑ **Contribution:** 1) We newly incorporate new GFVC algorithms into the AHG16 software for **bitrate range extension, performance improvement and GFV SEI “Fusion” function**. 2) We further **improve the AHG16 GFVC software code**, including improving codebase readability and function format uniformity, removing CUDA dependency, testing convenience and so on.

Algorithm Overview-DAC



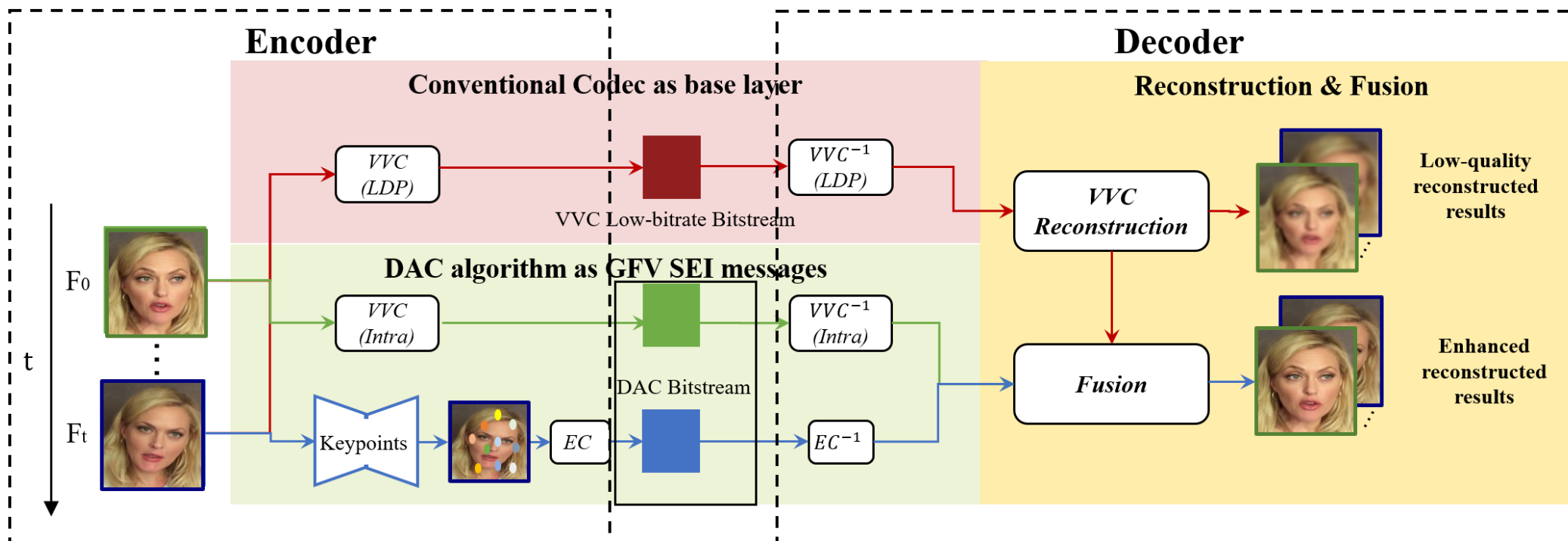
- ❑ The DAC algorithm is similar to FOMM, but it only **uses a group of 10 keypoints** as compact representation without any Jacobian matrices, in order to reduce bit consumption for coding the GFVC parameters without compromising reconstruction quality.

Algorithm Overview-HDAC



- ❑ The HDAC algorithm is an extension of DAC, aiming at overcoming the GFVC limitations in **limited bitrate coverage and unstable reconstruction quality** (e.g., long-term dependencies and background motion).
- ❑ **HDAC augments the DAC bitstream with an auxiliary stream consisting of a very low bitrate version of the video**, obtained through a conventional video codec (e.g., HEVC or VVC). Afterwards, the DAC output and auxiliary decoded video are fed into a fusion module such that the high-quality face video can be reconstructed.
- ❑ This is similar to the **driving picture fusion function in the GFV SEI message** in TuC.

Algorithm Overview-HDAC_Extension



- ❑ The HDAC_Extension can directly **regard the VVC low-bitrate stream as a base layer**, which can be **enhanced by signaling key points** or even other facial representations (3D keypoints and compact feature) through SEI message.
- ❑ A smart decoder that has the ability to **interpret the SEI message could significantly enhance the reconstruction quality of the VVC base layer** with a minimal rate overhead. In this manner, the reconstructed face video can maintain high quality whilst consuming less coding bits.

Proposed Testing Conditions

□ Herein, we follow the test conditions and software reference configurations for GFVC testing specified in **JVET-AG2035** for **low bitrate range**. It should be noted that since the updated algorithms are designed for **middle bitrate coverage**, we propose to modify test settings as follows,

- **Low bitrate ranges (JVET-AG2035):**

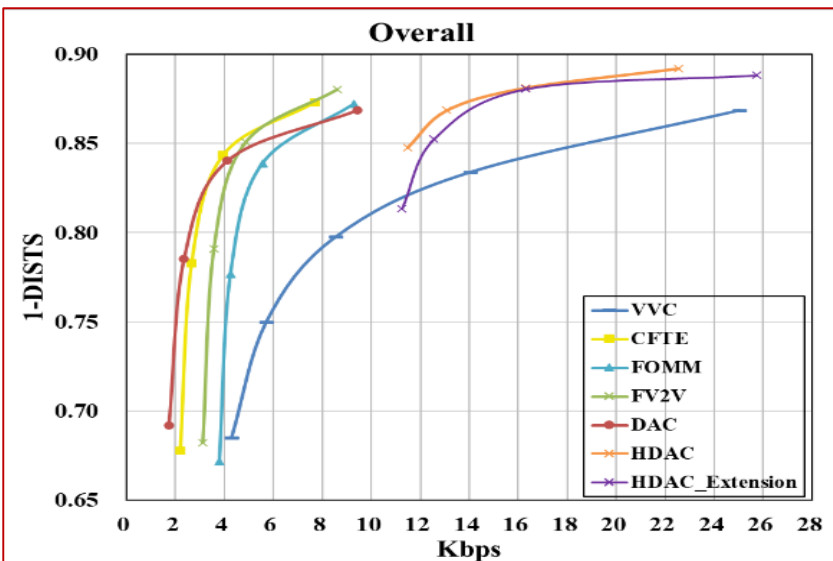
- ✓ VVC: VTM 22.2 LDB , QPs=37/42/47/52
- ✓ DAC: Base picture QPs= 22/32/42/52

- **Middle bitrate ranges (Newly-proposed):**

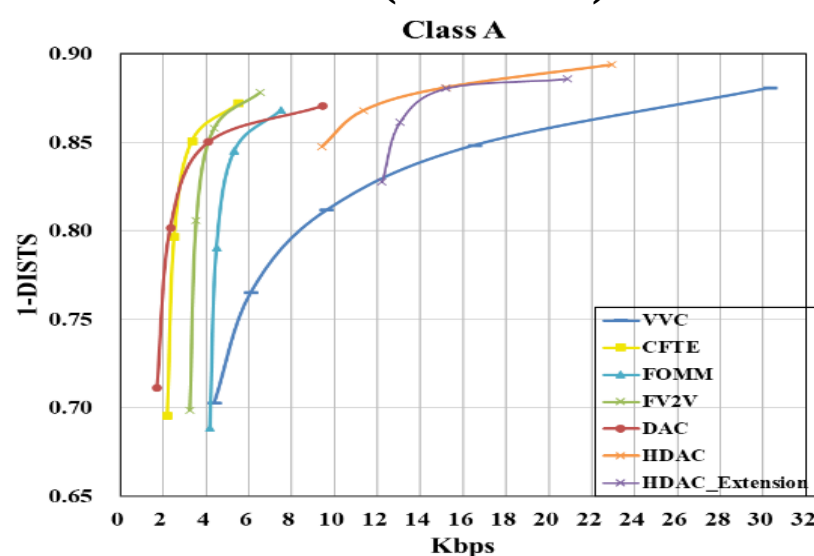
- ✓ VVC: VTM 22.2 LDB , QPs= 32/37/42/47.
- ✓ HDAC: the base picture QP of base-layer GFVC algorithm is **22**. Enhancement layer uses VTM 22.2 LDP with QPs=**32/42/47/52**.
- ✓ HDAC_Extension: base layer uses VTM 22.2 LDP with QP of **42** and Enhancement layer compresses the base picture with QPs of **12/22/32/42**.

RD Performance (Perceptual-level)

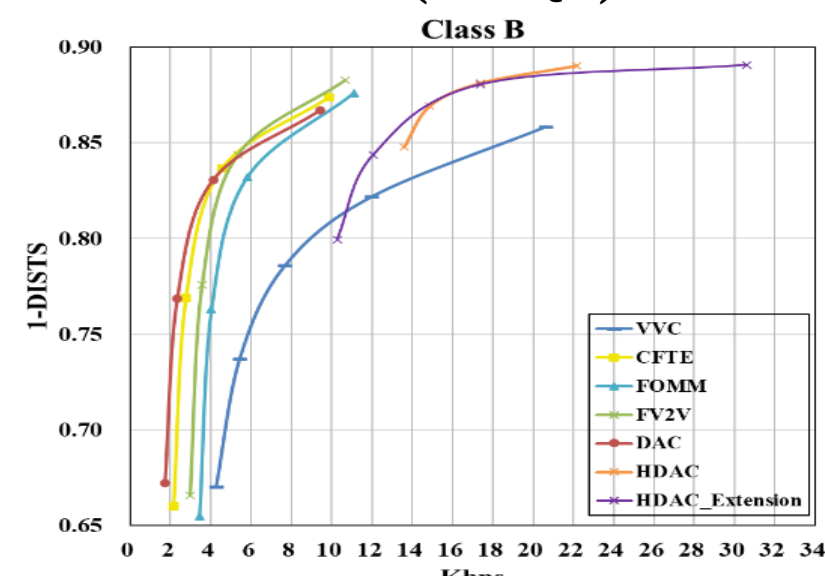
Overall



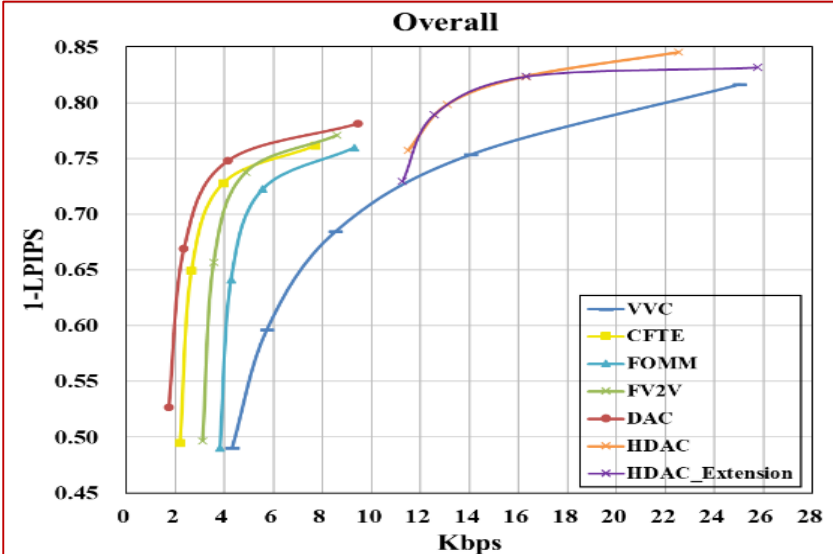
Class A (VoxCeleb)



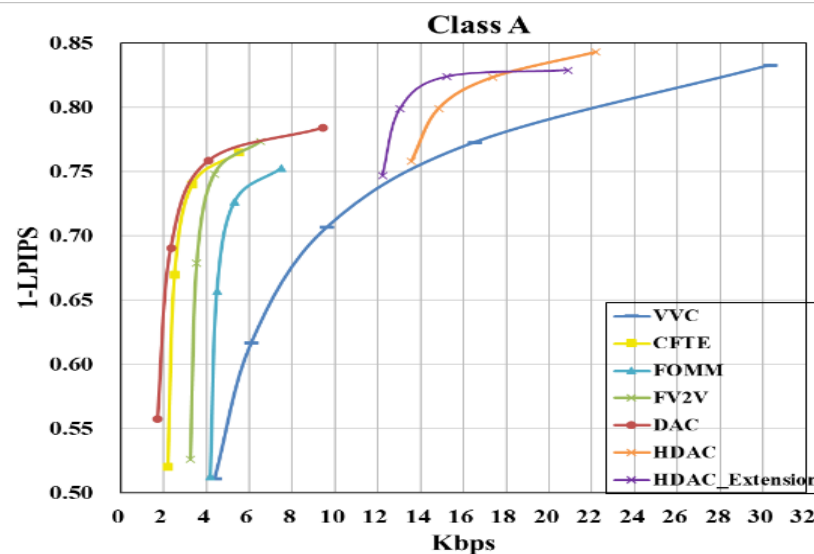
Class B (CFVQA)



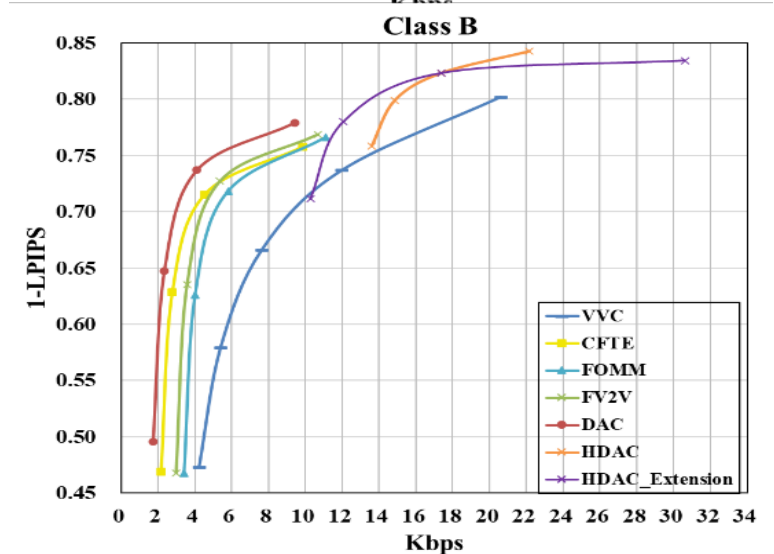
Overall



Class A

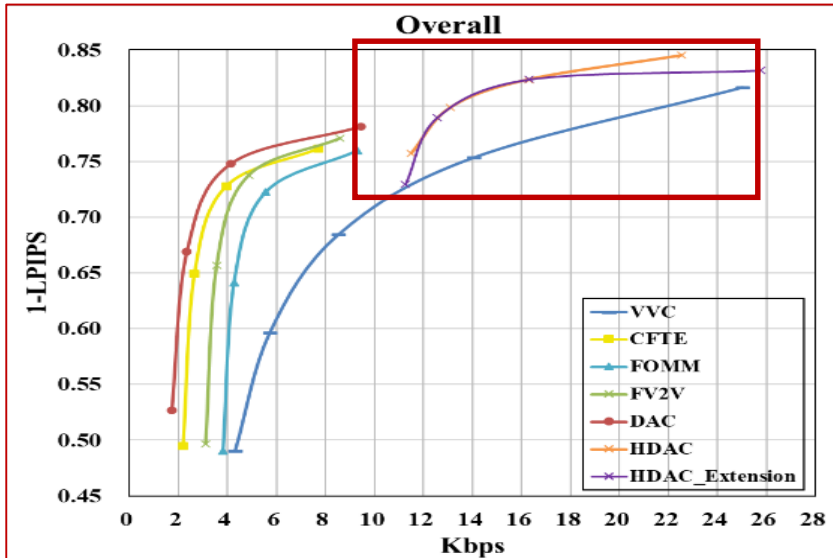
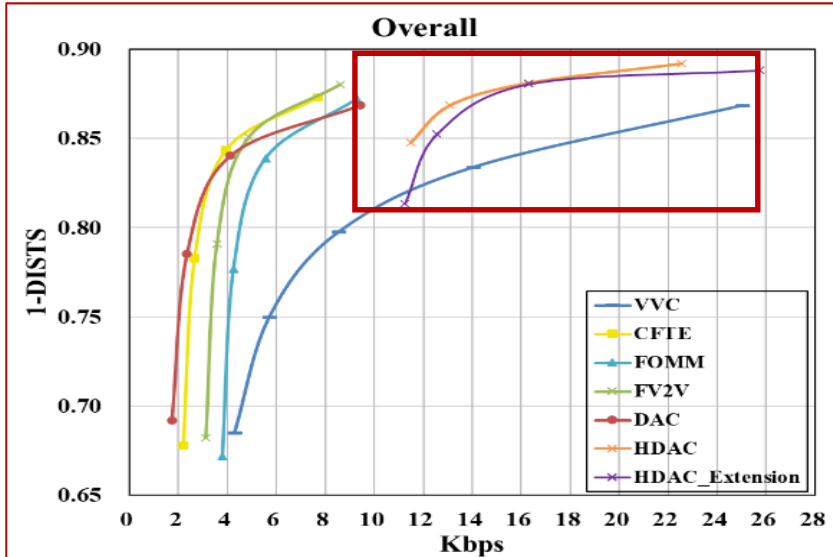


Class B



RD Performance (Perceptual-level)

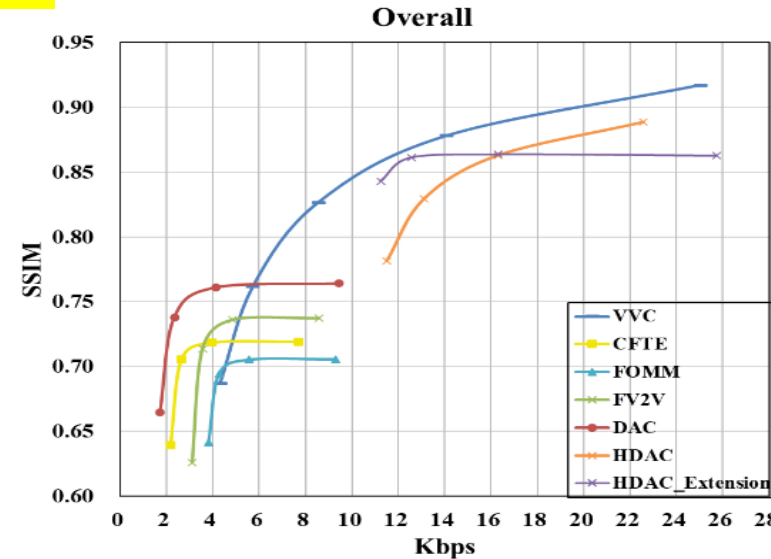
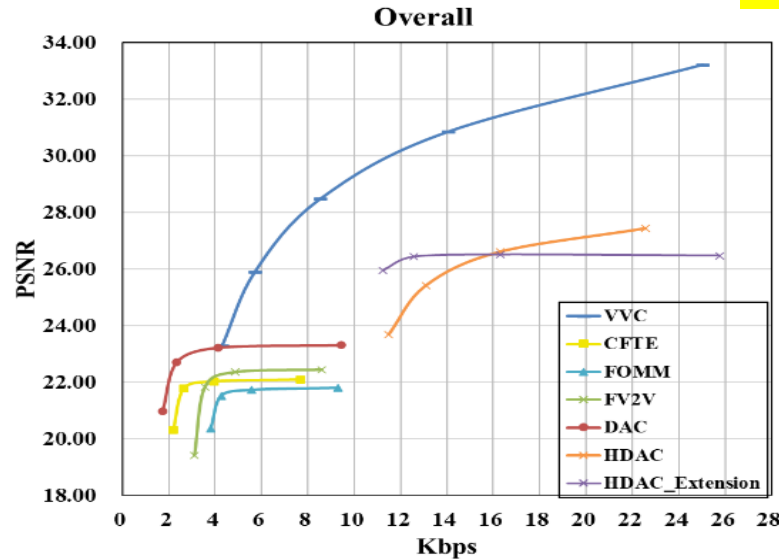
Overall



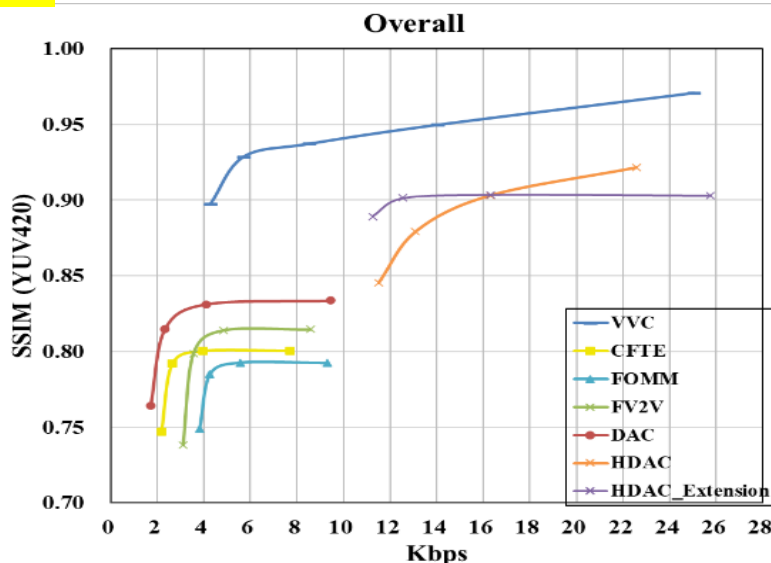
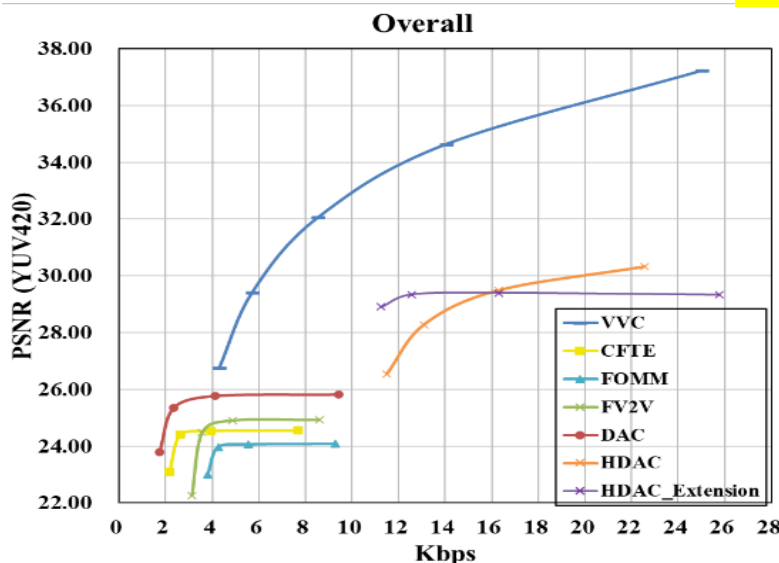
- ❑ DAC algorithm show **better or comparable performances** compared to FOMM, CFTE and FV2V algorithms.
- ❑ HDAC can **extend the current bitrate ranges** of GFVC from ultra-low (<10kbps) to relatively middle (10~30kbps) with promising RD performance compared with VVC.
- ❑ HDAC Extension implementation can directly **enhance the performance of VVC** by taking VVC as a base layer and using DAC with better I frame as its enhancement.

RD Performance (Pixel-level)

RGB 444



YUV 420



□ The updated methods can **improve the fidelity of signal reconstruction.**

□ Although the overall RD performance is **still not as good as VVC**, this is expected as all generative methods do not optimize for sample-level fidelity.

BD-rate Savings

Dataset		DAC		HDAC		HADC_Extension	
		Rate-DISTS	Rate-LPIPS	Rate-DISTS	Rate-LPIPS	Rate-DISTS	Rate-LPIPS
Average	Class A	-72.00%	-73.14%	-45.06%	-39.52%	-26.17%	-28.76%
	Class B	-62.71%	-63.05%	-9.38%	-5.50%	-9.31%	-12.58%
	Overall	-67.35%	-68.10%	-27.22%	-22.51%	-17.74%	-20.67%

❑ In comparison with the VVC anchor with different testing settings, the updated algorithms can achieve performance gains.

Subjective Comparisons - DAC

Original



VVC@5.49kbps



DAC@5.18kbps



Original



VVC@5.75kbps



DAC@4.26kbps



- ❑ At ultra-low bitrates, VVC reconstructed face images are unrecognizable, whereas DAC methods can still reconstruct vivid faces.

Subjective Comparisons -HDAC

Original



VVC@15.32kbps



HDAC@13.82kbps



Original



VVC@13.64kbps



HDAC@13.73kbps



□ HDAC can produce comparable reconstruction quality at middle-range bitrate .

Subjective Comparisons –HDAC_Ext.

Original



VVC@17.28kbps



HDAC_Extension@15.52kbps



Original



VVC@15.44kbps

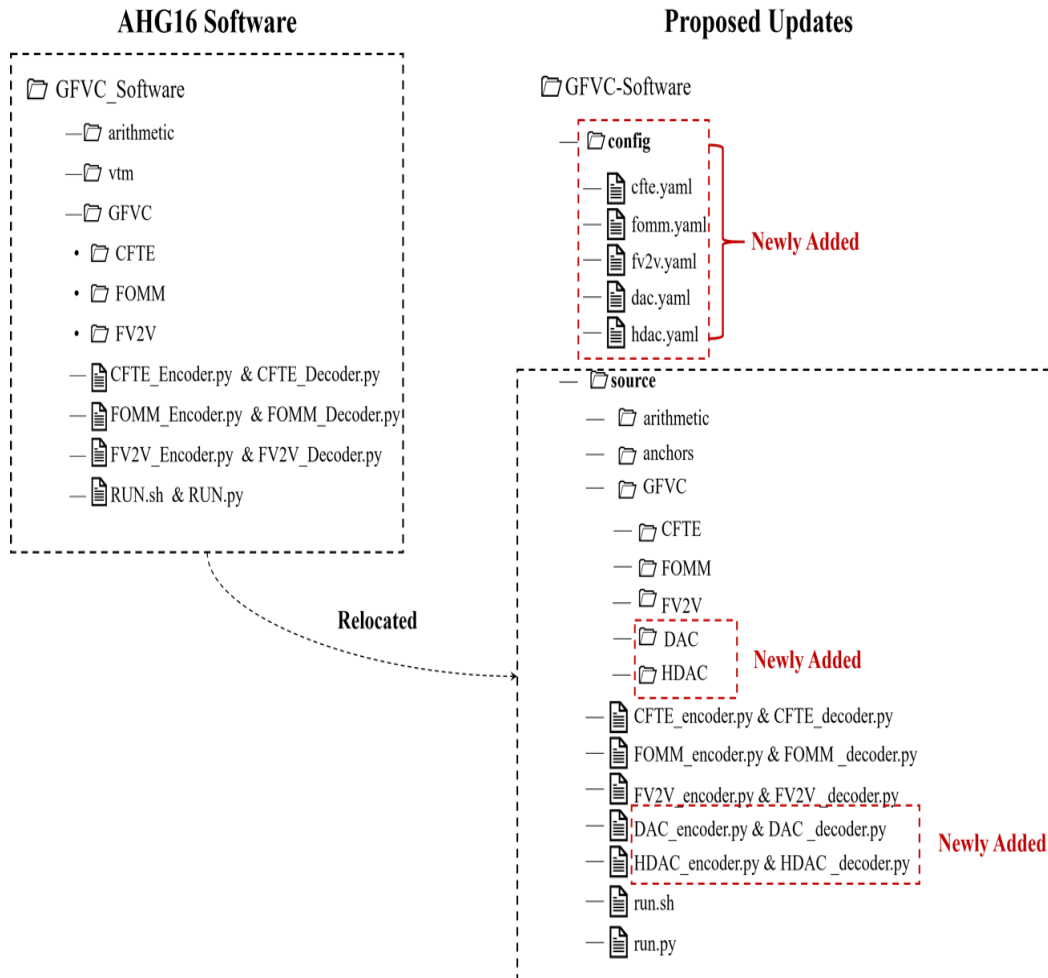


HDAC_Extension@14.44kbps



□ HDAC_Extension can produce comparable reconstruction quality at middle-range bitrate.

❑ Illustration of folder structure for the software implementation of GFVC



❑ Specific Details:

- We made the **codebase more readable** by separating the folder structure and formatting GFVC functions.
- We made the code forward **compatible with the latest versions of pytorch and scikit-learn**.
- We **removed all hardcoded dependencies on CUDA** so that the GFVC experimentation can be **executed on "CPU"**.
- We reconfigured the coding processes to remove the need to store GFVC parameters for all frames before they are coded; this way, we can **achieve on-the-fly coding of GFVC parameters for each frame**.
- We make **all testing procedures** (encoding/decoding/domain converting/quality evaluation/ results summary) finished in **a unified code file**.

- ❑ This contribution proposes to add two new GFVC algorithms (i.e., DAC and HDAC) in the AHG16 software for **bitrate range extension and performance improvement**. The updated HDAC algorithm also can provide an effective **“Fusion” implementation in GFV SEI messages**.
- ❑ This contribution also makes **improvements to the AHG16 software code** regarding codebase readability and function format uniformity, removing CUDA dependency, and so on.
- ❑ It is suggested to **adopt** both elements of this proposal and use the proposed package as the new version of AHG16 software.
- ❑ The crosscheck is supported by **Dolby (Updated Algorithms) and Panasonic (Software optimization)**.



Q & A