|  |  |
| --- | --- |
|  | Moving Picture, Audio and Data Coding  by Artificial Intelligence  www.mpai.community |

|  |  |
| --- | --- |
| N375 | 2021/09/30 |
| Source | Video group |
| Title | MPAI-EVC Evidence Project description |
| Target | MPAI Members |

Since the day MPAI was announced, there has been considerable interest in the application of AI to video coding. In the Video Codec world research effort focuses on radical changes to the classic block-based hybrid coding framework to face the challenges of offering more efficient video compression solutions. AI approaches can play an important role in achieving this goal.

According to a survey of the recent literature on AI-based video coding, performance improvements up to 30% are expected. Therefore, MPAI is investigating whether it is possible to improve the performance of the Essential Video Coding (EVC) modified by enhancing/replacing existing video coding tools with AI tools keeping complexity increase to an acceptable level.

In particular, EVC Baseline Profile has been selected because it is made up with 20+ years old technologies and has a compression performance close to HEVC, and the performance of its Main Profile exceeds that of HEVC by about 36 %. Additionally, some patent holders have announced that they would publish their licence within 2 years after approval of the EVC standard.

Figure 1 describes the reference codec architecture. The red circles represent the encoder block candidates for enhancement/replacement with AI tools.

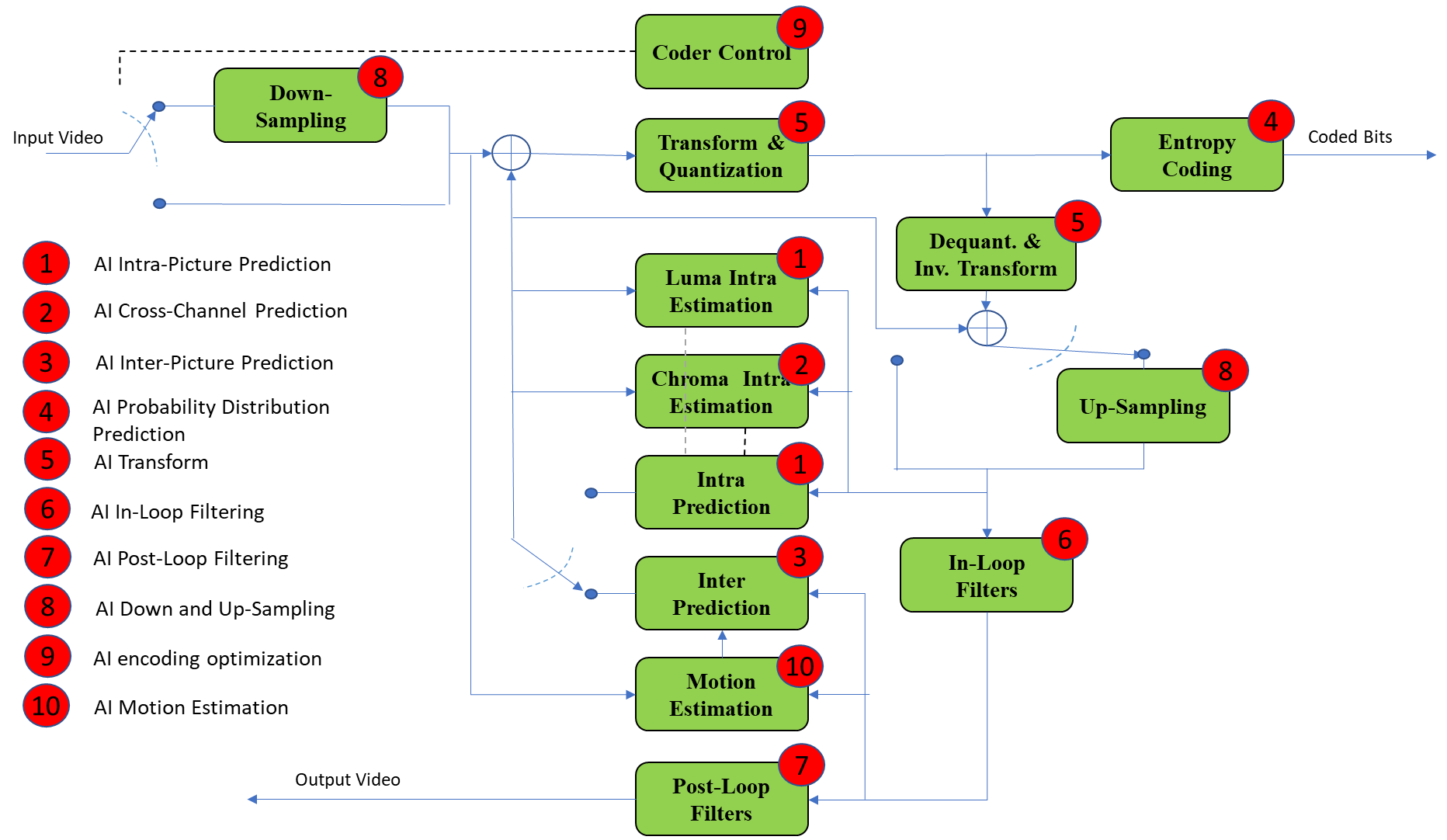


Figure 1: Hybrid video codec schema. Potential AI improvement

While MPAI-EVC Use Cases and Requirements is ready and would enable the General Assembly to proceed to the Commercial Requirements phase, MPAI has made a deliberate decision not to move to the next stage because it first wanted to make sure that there was indeed confirmation that the individual results collected from different papers were confirmed when implemented in a unified platform.

The name of the project is MPAI-EVC Evidence project with the goal to enhance EVC to reach at least 25% improvement over EVC baseline profile.

MPAI members are globally distributed all around the world and they work across multiple software frameworks. The first challenge was to enable them to collaborate in real-time in a shared environment. The main goals of this environment are:

* + to test independently developed AI tools on a common EVC code base
  + to run the training and inference phases in “plug and play” manner

To address these requirements, the group decided to adopt a solution based a networked server application listening for inputs over an UDP/IP socket.

The group is working on three tools (Intra prediction, Super Resolution, in-loop filtering) and for each tool there are three phases: database building, learning phase and inference phase.

**Intra prediction enhancement**

In this section we describe our experiments at enhancing the EVC baseline mode 0 (DC) intra predictor by means of a convolutional autoencoder. In the following we consider only 32x32 CUs, however the described method can be extended in principle to 16x16 and 8x8 CUs (we leave 4x4 predictors to our future endeavours). The autoencoder architecture we rely upon is inspired by Context Encoders [pat2016] and is depicted in Figure 2. It includes a 4-layers convolutional encoder and a 3-layers convolutional decoder for a total of 4M learnable parameters. The encoder subnetwork receives in input a 64x64 patch representing the decoded context available at the decoder (D0, D1, D2), whereas the bottom-right corner (P3) represents the 32x32 DC intra predictor the autoencoder must enhance. The network outputs an enhanced predictor (P3’) and is trained to minimize the MSE between the enhanced intra predictor P3’ and the original, uncompressed, 32x32 co-located block O3 (Figure 3) [wang2019].

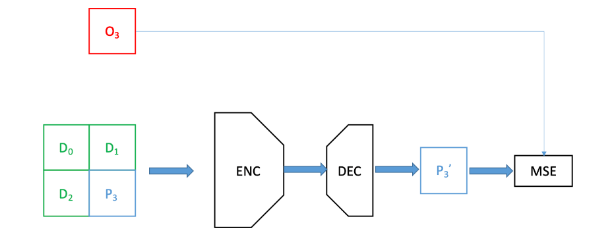


Figure 2: the convolutional autoencoder used to generate a 32x32 intra predictor from a 64x64 context.

For this purpose, we have generated a training set consisting of 1M patches of 32x32 predictors from the AROD dataset [arod]. Namely, we have encoded 750 randomly drawn images and for each 32x32 CU we saved to disk the 64x64 decoded context, the corresponding DC predictor and the original uncompressed colocated patch. These patches are organized in the so called Original-Decoded-Predicted (ODP) format as 64x192 images (Figure 3), constituted by three 64x64 pixels images, where the lower right hand side patch is the codec processed patch while the surrounding patches are the neighbourhood information (causal context).

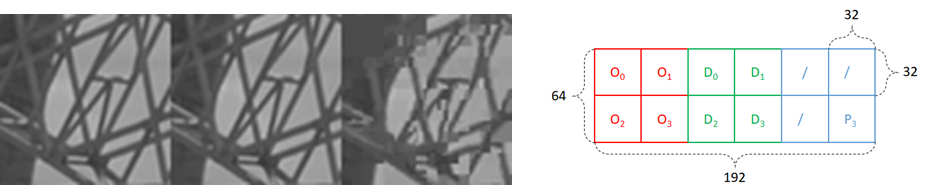


Figure 3: the ODP patch format we used to store the training patches and an example of ODP patch.

The trained autoencoder is then wrapped within a networked server application listening for inputs over an UDP/IP socket. The EVC encoder is patched to send, for each 32x32 CU, the 64x64 decoded context to the server for the mode 0 EVC intra predictor. The server feeds the received 64x64 context into the trained autoencoder and returns to the EVC encoder the enhanced 32x32 predictor. The EVC encoder finally replaces the standard DC intra predictor with the autoencoder-enhanced DC predictor and this predictor is then put into competition with the other 4 EVC intra predictors (modes 1-4) and encoding proceeds as usual.

We experiment encoding the 1st frame of the Class B sequences (1920x1080) BasketballDrive, BQTerrace, Cactus, Kimono1, ParkScene and computed BD-rate over the QP range = {22, 27, 32, 37, 42, 47}.

Table 1 shows the BD-rate gains of the DC-enhanced EVC encoder over the reference EVC encoder (we recall that 32x32 CUs intra predictors only are enhanced). The proposed method achieves consistent gains over almost all tested sequences, with an average gain of about 0.8% and over 1.5% gains with the sequence BasketballDrive (gains stem from high QPs, above 32: about 1%, Table 2). The rightmost column of the Table 1 shows the fraction of times our enhanced DC predictor is preferred by the EVC encoder RDO algorithm to the other 4 intra predictors: our predictor achieves comparatively lower RD cost in over 50% of the cases.

|  |  |  |
| --- | --- | --- |
| **Sequence** | **BD-rate variation [%]** | **Selection rate** |
| BasketballDrive | -1.6037 | 56.15 |
| BQTerrace | 0.0020 | 73.59 |
| Cactus | -0.4680 | 82.83 |
| Kimono | -0.9291 | 50.62 |
| ParkScene | -0.5807 | 79.16 |
| AVG | -0.7159 | 68.47 |

Table 1: BD-rate gains for the EVC baseline encoder where the 32x32 mode 0 (DC) Intra predictor is replaced by the predictor enhanced by a convolutional autoencoder (QPs 22-47)

|  |  |
| --- | --- |
| **Sequence** | **BD-rate variation [%]** |
| BasketballDrive | -2.5180 |
| BQTerrace | -0.1287 |
| Cactus | -0.7256 |
| Kimono | -1.1523 |
| ParkScene | -0.8006 |
| AVG | -1.06504 |

Table 2: BD-rate gains for the EVC baseline encoder where the 32x32 mode 0 (DC) Intra predictor is replaced by the predictor enhanced by a convolutional autoencoder (QPs 32-47)

We recall that in these preliminary experiments, only 32x32 CUs intra predictors were enhanced with the proposed method, so further gains can be expected when the proposed method is extended to smaller CUs (16x16 and below), which represent the large bulk of CUs actually chosen by the encoder RDO mechanism.

**Super resolution**

In this section we describe our experiments on the super resolution step to improve the overall EVC decoding system performances. The super resolution step was added after the Post-loop filters (see Figure 1, block 7). In order to achieve this goal, a preliminary analysis phase was carried out, which involved studying the state of the art and then testing different neural models for the super resolution task including: **SRGAN** 'Ledig et al (x)' , **ESRGAN** 'Wang et al (x)' and **DRLN** 'Anwar et al (x)'. This phase of study and experimentation showed that, of these, the best performing model was the **DRLN**, which was adopted for subsequent trials.   
The **D**ensely **R**esidual **L**aplacian **N**etwork model proposes a new type of architecture based on cascading over residual on the residual architecture, which can assist in training deep networks. These new features help in bypassing enough low-frequency information to learn more accurate representations (Figure 4 - DRLN architecture).

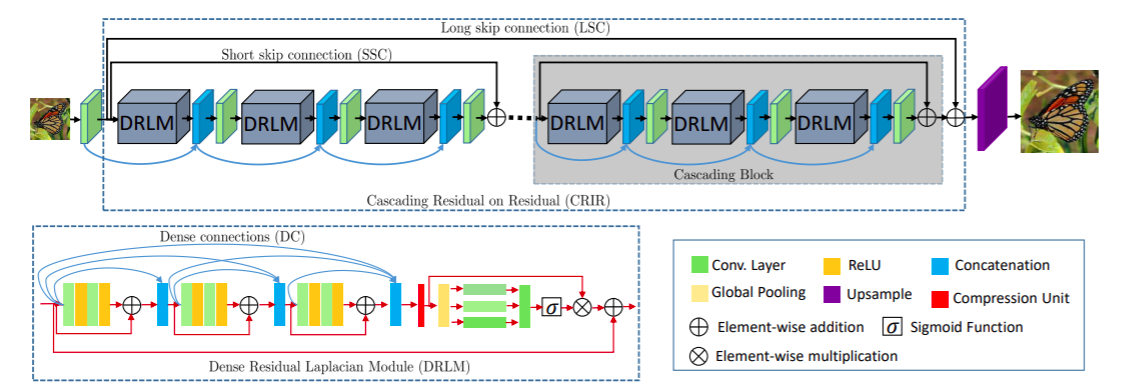


Figure 4 - DRLN architecture, Anwar et al (x)

Furthermore, Laplacian attention mechanism has been introduced, which has a two-fold purpose: to learn the features at multiple sub-band frequencies and to adaptively rescale features and model feature dependencies.   
The state-of-the-art methods model the features equally on a limited scale, ignoring the abundant rich frequency representation at other scales (Figure 5 - Laplacian Attention Mechanism).

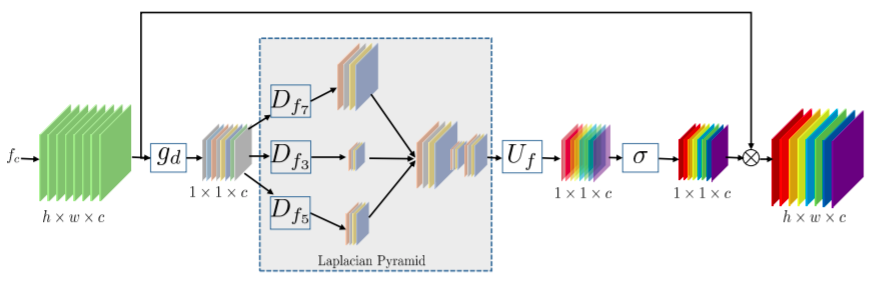


Figure 5 - DRLN Laplacian Mechanism, Anwar et al (x)

Therefore they lack discriminative learning capability and capacity across channels.

DRLN addresses these issues and utilizes the dense connection between the residual blocks to use the previously computed features. At the same time, using Laplacian pyramid attention it is possible to weight the features at multiple scales and according to their importance.

This architecture is employed as an up-sampler whenever the input sequence, into the decoding system, has been downsampled. Due to the complexity, in terms of memory and computational costs, of the used deep-learning approach we have designed an efficient training strategy. This has been solved by subdividing the input frame in crops and developing suitable training and validation sets based on the cropping strategy adopted. To achieve this, we have employed two strategies, both based on the entropy information of the input frame. This is calculated by estimating at each pixel position (i,j) the entropy of the pixel-values within a 2-dim region centred at (i,j). The first strategy uses a random crop if, and only if, its average entropy exceeds a given threshold. The second strategy selects n crops, of the same size, from the total crops available in each frame. This is based on the importance sampling technique applied on the entropy values distribution of all crops in each frame. A particular attention needs to be given to the right combination of the crop and batch sizes as in such a way a tradeoff with respect to GPU memory consumptions can be achieved. Table 3 shows the tested combination:

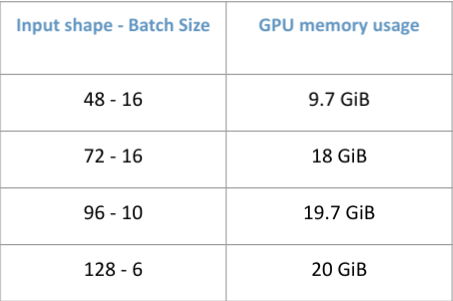


Table 3: Comparison of different combinations of crop and batch size

in terms of GPU memory usage.

All the experiments have been conducted on two uses cases, e.g., SD to HD and HD to 4K, with in mind to extend them in the future to SD to 4K. The chosen quantizer parameter (QP) values have been 15, 30 and 45, with activated and not activated the in-loop filter (Figure 1, block 6), which in our case is a deblocking filtering. Based on the above possible configurations we have built a dataset to train the super resolution network: 2000 pictures (KAGGLE DATASET 4K standard resolution images (2057 files) <https://www.kaggle.com/evgeniumakov/images4k>).

We have experimented the performances of the trained network on 8 sequences of 500 frames each, and in the following the graphs with the results.

Figure 6: sequence park joy, QPs(15,30,45)

Figure 7: sequence Crowd run, QPs(15,30,45)

Figure 8: sequence diego and the owl, QPs(15,30,45)

Figure 8: sequence ducks take off, QPs(15,30,45)

Figure 9: sequence Rome 1, QPs(15,30,45)

Figure 10: sequence Rome 2, QPs(15,30,45)

Figure 11: sequence Rush hour, QPs(15,30,45)

Figure 12: sequence Talk show, QPs(15,30,45)

**Inloop filter**

Loop filtering module is first introduced into video coding standard since MPEG-2. Inspired by the success of NN on image/video restoration filed, many of NN based loop filters are designed

to remove compression artifacts.

The group has started the work on the ‘out of the loop’ filtering as a first step. During the super resolution training phase, the EVC deblocking filter has not been activated. The neural network was trained to learn how to improve the quality of the decoded picture.

**Dataset building**

Currently MPAI has created the following dataset for training/test:

* Intra prediction track:
  + - Training dataset of 32x32 intra prediction block created, 1.5M pictures
    - Training dataset 16x16: 5.5M pictures.
* Super resolution track:
  + We have built a dataset to train the super resolution network: 2000 pictures (KAGGLE DATASET 4K standard resolution images (2057 files) <https://www.kaggle.com/evgeniumakov/images4k> )
    - 3 use cases:
      * SD to HD
      * HD to 4K
      * SD to 4K
    - 3 resolutions:
      * 4k - 3840x2160
      * HD - 1920x1080
      * SD - 960x540 (actually 544…)
    - 3 QP values:
      * 15
      * 30
      * 45
    - Two options:
      * Deblocking enabled
      * Deblocking disabled
    - 3 resolutions, 3 quality values, 2 coding tool sets (options)
      * 18 versions, 170 GB dataset

Test dataset:

* Crowd run, Ducks take off, Park joy, Diego and the owl, Rome1, Rome2, Rush Hour, Talk Show, (500 frames per sequence)
* 3 resolutions:
  + 4k - 3840x2160
  + HD - 1920x1080
  + SD - 960x540
* 3 QP values:
  + 15
  + 30
  + 45
* Two options:
  + Deblocking enabled
  + Deblocking disabled

**References**

[pat2016] Pathak, Deepak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. "Context encoders: Feature learning by inpainting." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2536-2544. 2016.

[wang2019] Wang, Li, Attilio Fiandrotti, Andrei Purica, Giuseppe Valenzise, and Marco Cagnazzo. "Enhancing HEVC spatial prediction by context-based learning." In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4035-4039. IEEE, 2019.

[arod] Schwarz, Katharina, Patrick Wieschollek, and Hendrik PA Lensch. "Will people like your image? learning the aesthetic space." In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 2048-2057. IEEE, 2018.