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

|  |  |
| --- | --- |
| Nxxx | 2023/01/25 |
| Source | Video group |
| Title | MPAI-EVC Evidence Project report and plan |
| Target | MPAI Members |

The goal of the group is to enhance EVC (Essential Video Coding) using AI-tools to reach at least 25% improvement over the baseline profile. The group is currently working on three coding tools: Intra prediction, Super Resolution, and in-loop filtering. For each tool, in the following we describe the proposed approach and the steps of database building, learning phase and inference.

**MPAI dataset**

Th new dataset based on BVI-DVC Part 1 (University of Bristol)combined with:

* Ultravideo dataset, containing 16 4K 10-bit raw sequences, available here <http://ultravideo.fi/>
* The opensource SVT datasets (7 new plus 5 old 4K sequences), available here <https://www.svt.se/opensource/content>
* The Tencent video dataset (85 4K sequences), available here <https://multimedia.tencent.com/resources/tvd>

for a total of 350 4K sequences.

To be used in the MPAI-EVC experiments, the sequences must be pre-processed. The actions taken by the group to prepare the sequences for the training are shown in Figure 1.



Figure 1 processing workflow

We finished the coding at fixed QP (as per the Common Test Conditions): 22, 27, 32, 37, 42, 47.

The coded video sequences are in YUV 4:2:0 10-bit format, and come either in the BT.709 or BT.2020 colour spaces. Since the super-resolution network operates on PNG images, the application of the correct colour space information in the conversion process from YUV to PNG is critical to avoid introducing unwanted image deterioration and, as a consequence, losing coding efficiency due to factors other than the coding scheme itself.

Thus, two different workflows are followed, accounting for the colour space difference. The ffmpeg command lines used for the conversion of BT.709 and BT.2020 content respectively, are

ffmpeg -pix\_fmt yuv420p10le -s videosize -i inpufile.yuv -vf scale=out\_color\_matrix=bt709:flags=full\_chroma\_int+accurate\_rnd+lanczos -c:v png outputdir/%3d.png

and

ffmpeg -pix\_fmt yuv420p10le -s videosize -i inpufile.yuv -vf scale=out\_color\_matrix=bt2020:flags=full\_chroma\_int+accurate\_rnd+lanczos -c:v png outputdir/%3d.png

Other datasets, such as the Youtube UGC dataset are being investigated.

The dataset is ready, and it is available on MPAI ftp server.

Also the test sequences are available on the ftp server.

Figure 2: MPAI-EVC datasets

We now have two datasets (Figure 2): a huge training dataset with 350 4K sequences that we use for training. Moreover, we have a test dataset with few sequences, only 24 sequences, which we use for testing. The overlap between these two datasets is zero, which means we do not use test sequences in the training phase.

In addition, all the results we present are based on the test sequences. We usually call them JVET sequences, because they are in common with the MPEG JEVET group.

**Intra prediction tool**

We address the challenge of predicting an intra-coded block given its context (Intra prediction) as an image inpainting problem, i.e. recovering pixels of an image that are unavailable due to, e.g. occlusions or information loss. Masked convolutional neural networks have been recently proposed for image inpainting exploiting the apriori information from the context to recover the missing image area. The proposed method relies on masked convolutions to generate the block predictor starting from a decoded context of 64 × 64 pixels (Figure 3). For example, for each 32x32 coding unit a 64x64 context is sent to the autoencoder. The autoencoder returns to the EVC encoder a 32x32 predictor that is considered as a 6th EVC Intra predictor mode that is put into competition with the other 5 predictors. The generated bitstream is fully decodable under the assumption that the autoencoder network is also available at the decoder side.



Figure 3: context on the left and the predictor on the right

The masked autoencoder (Figure 4) is trained in a supervised manner for 1000 Epochs over a set of randomly drawn patches from about 800 images representing various types of contents by minimising the absolute error (ABS) between the network output and the original patch.



Figure 4: Procedure for training the convolutional autoencoder used to generate the Intra predictor.

We worked on the architecture of the encoder, switching to a VGG-inspired topology where the encoder is made of 5 pairs of stacked convolutional layers with 3x3 filters interleaved by subsampling for a total of 10 convolutional layers (results in Table 1).



Table 1: results with VGG-like architecture over the standard JVET test sequences for the 22-37 and 22-42 QP ranges and for 5 modes

Table 1 shows the improvement compared to the previous architecture: in green the delta wrt reference, 5 modes. The BD rate increased by an average of 1 percent from the previous architecture.



Table 2: results with the new architecture VGG-like and for 6 modes

Table 2 shows that at the moment there is no gain in adding a sixth mode (values in red), rather there is a generalized loss (small gains only for class A, more pronounced losses for class E). We are trying to explain these numbers by calculating some statistics on the predictor modes (Table 3).



Table 3: percentage of modes use: left: 5 modes, reference; middle: 5 modes, new architecture; right: 6 modes new architecture on JVET Class A

Table 3 shows the percentage of use of a particular mode on the Class A JVET sequences. With the new architecture and 5 modes, the percentage of neural predictor use increases from 51% to 62%. Unfortunately, by adding the sixth mode, the percentage drops to 56%, because with 5 modes the DC mode is replaced, while with 6 modes the DC mode reaches 25% utilisation.



Figure 5: distribution of modes use

Figure 5 shows that Class A has about 81% of the predictions in modes 0 and 1, whereas this number drops to 76% for the Class E.

Adding a 6th new intra yield no gains currently. From our analysis, it seems that NN predictor is more selected than DC but still we have no gains. This is probably due to the cost of signalling the 6th predictor. We are investigating the bit-cost for signalling and residuals.

**Super-resolution tool**

The super-resolution step is added as a post-processing tool. The picture before encoding with EVC baseline profile is downscaled and then the super-resolution network is applied to the decoded picture to get the native resolution.

We have carried out extensive preliminary experiments, to help us in understanding the different options that will make sense to carry in the final experiments using the standard training and test datasets, which will be used in all the experiments carried out in this activity. We have trained the selected deep-learning approach for super-resolution on 4 QPs (15,30,37 and 45). We have tested its performances on 8 test sequences for the case of SD to HD, and on 3 test sequences for the case of HD to 4K.

The group has worked on the computation of the BD-rate SD to HD, Table 2, showing an improvement of -3.14% when compared with the ground truth EVC.

|  |  |  |
| --- | --- | --- |
| **Sequence** | **Class** | **BD-Rate** |
| Crowd Run | Class B 1920x1080 60/50 fps, 8 bpp | -1.24% |
| Ducks Take Off | Class B 1920x1080 60/50 fps, 8 bpp | 2.12% |
| Park Joy | Class B 1920x1080 60/50 fps, 8 bpp | 1.40% |
| Diego and Owl | Class B 1920x1080 60/50 fps, 8 bpp | 8.11% |
| Rome 1 | Class B 1920x1080 60/50 fps, 8 bpp | 0.19% |
| Rome 2 | Class B 1920x1080 60/50 fps, 8 bpp | -18.81% |
| Rush Hour | Class B 1920x1080 60/50 fps, 8 bpp | 4.90% |
| Talk Show | Class B 1920x1080 60/50 fps, 8 bpp | -21.75% |
| **Average: -3.14%** | | |

Table 2: BD-rate performances on all the 8 test sequences

The HD to 4K testing phase has been finalised on all QPs (15,30,37 and 45).

Figure 6 shows the BD-rate curves for each sequence, where the network trained on QP 15 dataset has been used over the test set for all QPs. The reasoning behind it, is to learn the features information, which is intrinsic within the compressed sequences, at the highest quality possible (QP 15). This will avoid any shortcoming in the generalisation capabilities that we have faced doing separated training for each QP.



**Figure 6** BD-Rate curve for all sequences and QPs, showing the BD-rate variation (Bjontegaard) averaged over all the QPs.

The above results in the SD2HD and HD24K case were calculated by training the network on a Kaggle dataset (8-bit sources) and testing the sequences listed in the tables (8-bit sequences).

**New training Dataset set**

We decided to build a new training dataset consisting of high-quality 4K sequences (350 sequences) to improve on previous results by exploiting 10-bit sources.

We are following two paths:

* Feeding the neural network directly with the Y component from the patch extraction phase (without converting into the specific image format, avoiding any additional introduction of errors) and feeding the neural network with the Y component replicated 3 times (to follow the data input standard of the neural network)

This will be done only for the lower QP because in the preliminary experiment, as described above, we are able to guarantee the best performances while avoiding any generalisation issue.

The training datasets, patches extraction, for all the types of training listed in ‘Training Strategies’, have been initiated. However, for the SD2HD case, some issues with few sequences have been found and are under investigation. We have also started to analyse the test sequences to see if there are any potential problems so that we will be ready after training to run the test.

**New training Strategies**

We have identified different types of training strategies as follows:

1. Using the original network and its weights without any additional training
2. Starting from the configuration of point 1., we suggest performing refinement training allowing us to compensate for the limited number of sequences we may have in the standard training set. Based on the preliminary experiments this will be performed only on the lower QP (22). This will help in reducing the overall required training time. This approach has been so far found as the best solution providing good results for all QPs at the inference stage and limiting generalisation issues as shown in the previous sections.
3. Training only on the SD2HD transformation and use it also for the HD24K. This will allow us to reduce the overall required training time.
4. Training only on uncompressed data. This will be done, to see if the obtained quality can be further improved.

**Intra & super resolution combined**

The group tested the pipeline with the two combined tools Intra and Super resolution (Figure 11). So far, gains for the NNIntra the DLRN SR tools have been evaluated independently, however the underlying combined gains are expected to be strongly not linear. Therefore, we performed coding experiments where the two NN Intra and SR encoding tools were jointly activated. In the combined tests, a 4K picture is bicubic-downscaled to HD before being EVC-compressed with the NNIntra tool enabled. At the output of the decoder, the SR tool is recovers the native 4K resolution. Experiments were performed only on the first 9 frames (IBBBBBBBB GOP scheme) of the Class A 4K sequences due to complexity constraints, the QPs under consideration are 22, 27, 32, 37, 42.

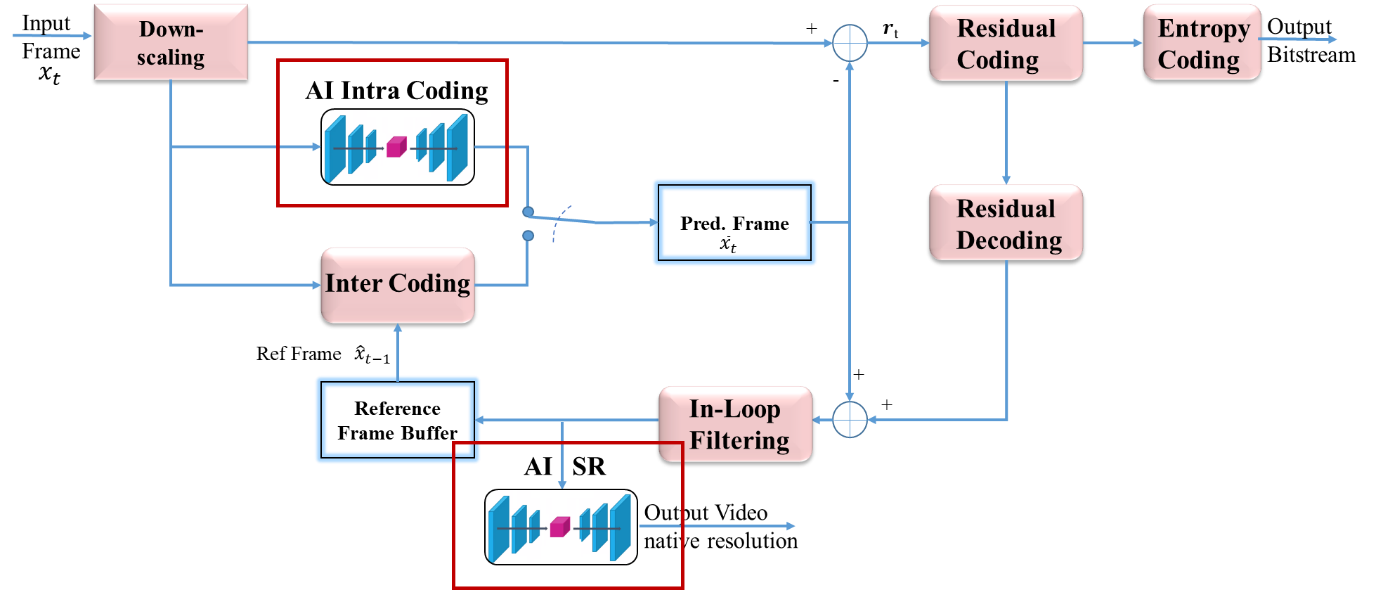


Figure 11: Setup for the combined NNIntra and Super Resolution (SR) tools

For the SR tool, we considered the pretrained DLRN network mentioned above plus the simpler the bicubic algorithm as a further reference, as illustrated below.

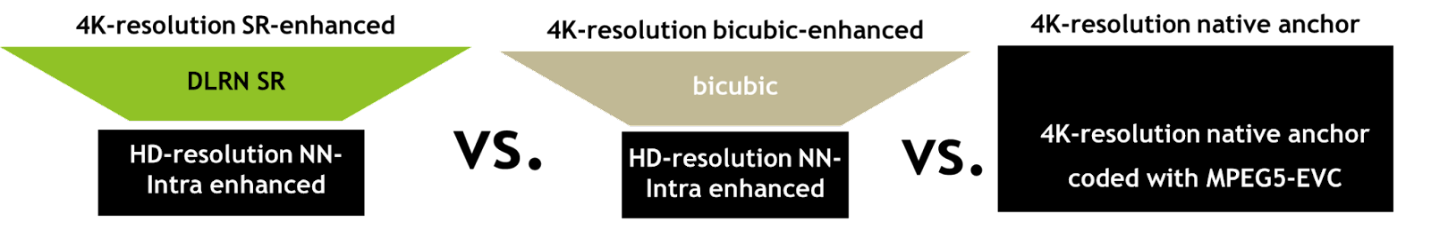


Figure 13: Joint NNIntra +SR comparison setups.

For all sequences and for all QPs we took the output of the EVC + NN intra and for each frame we considered the Y channel normalized between 0 and 255.

As the DLRN network was trained over RGB data, we decided to input into the network the Y channel repeated 3 times (Figure 14).

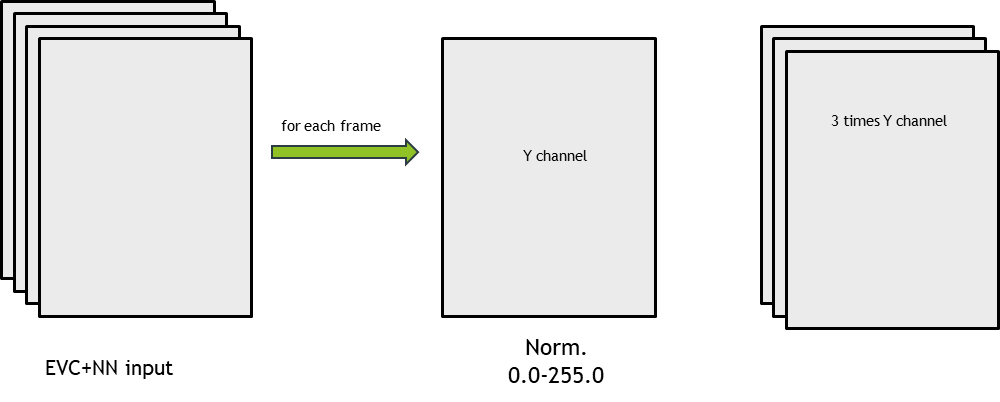


Figure 14: input to the DLRN network

The output of the Network is clipped between 0 and 255 and normalized between 0 and 1023. Finally, we take the mean of the 3 channels of the DLRN as the upsampled Y channel upon which we measure video quality (Figure 15).

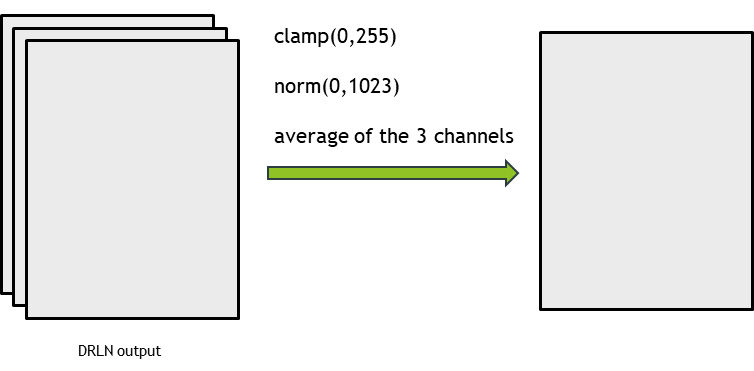
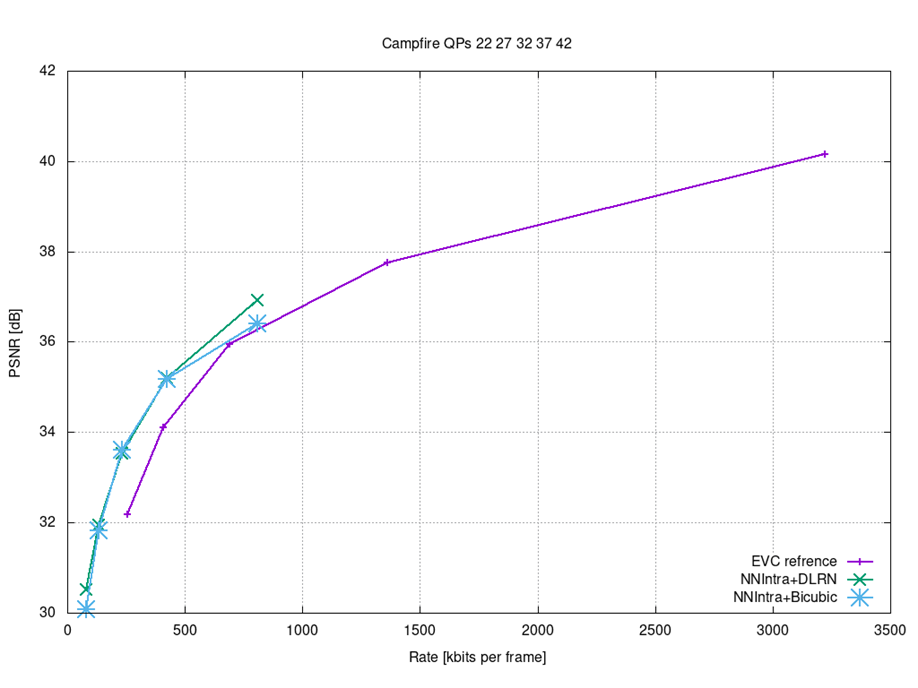
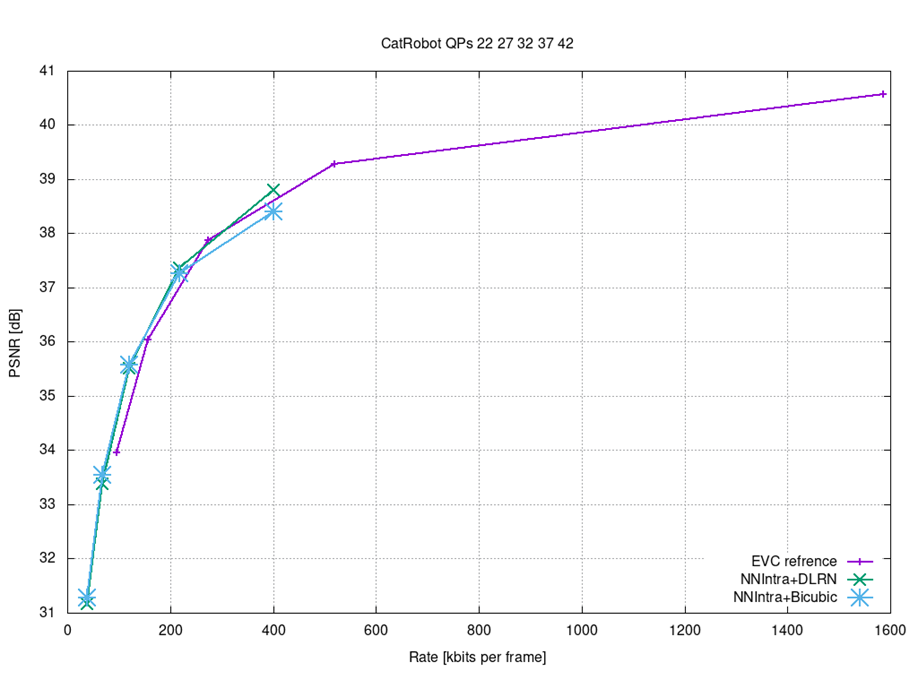
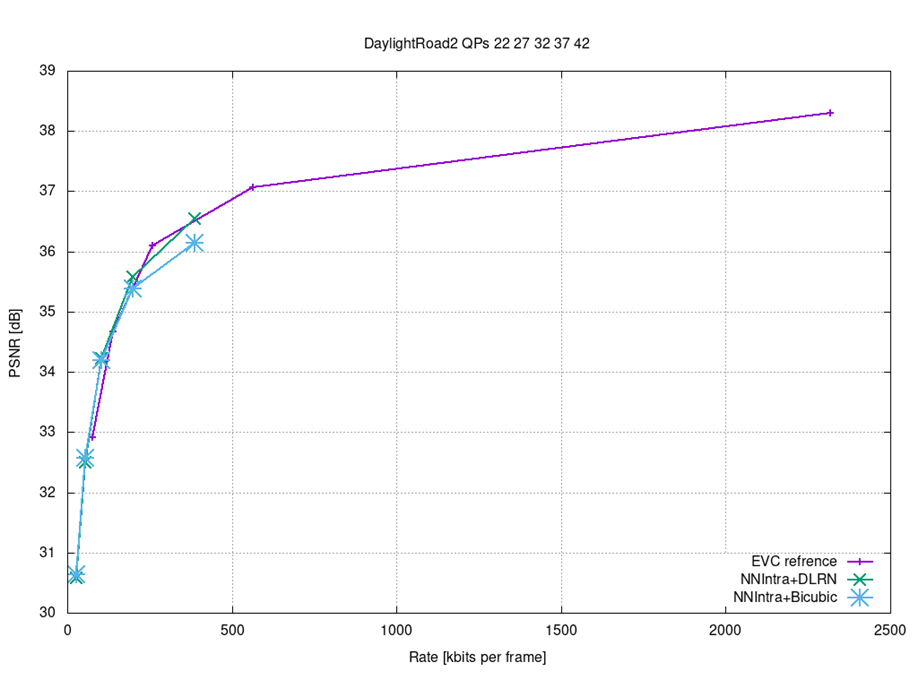
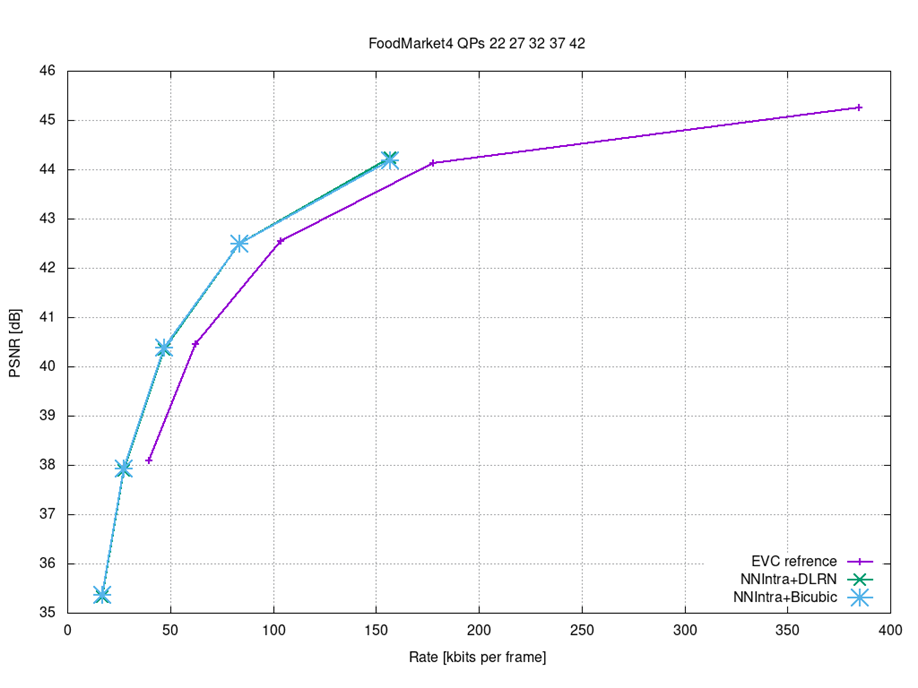


Figure 15: output of the DLRN network

Figure 16 compares the RD curves between reference 4K EVC encoding, IntraNN+ bicubic upsampling and IntraNN+DLRN upsampling for the 6 Class A 4K sequences we tested.

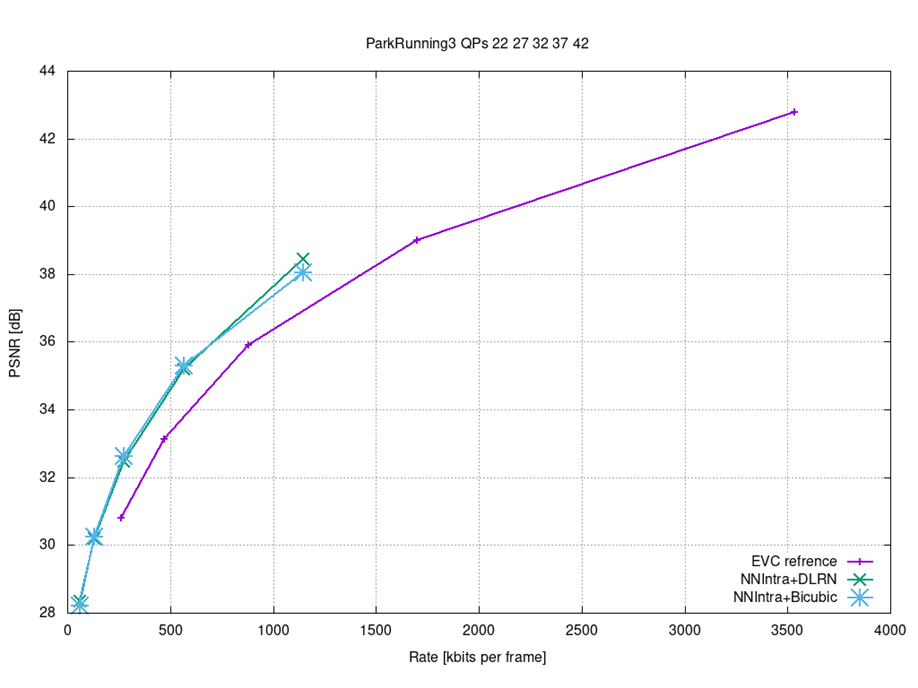
 

Figure 16: RD curves: Bicubic vs. DLRN network

The corresponding results are shown in Table 3 below and show combined BDrate gains between 17% and 18% for the bicubic and DLRN implementations of the SR tool respectively, with BDPSNR gains always above 0.5 dB.  
Weplan to retrain/refine the DLRN network over 10 bit sequences to extract additional gains.

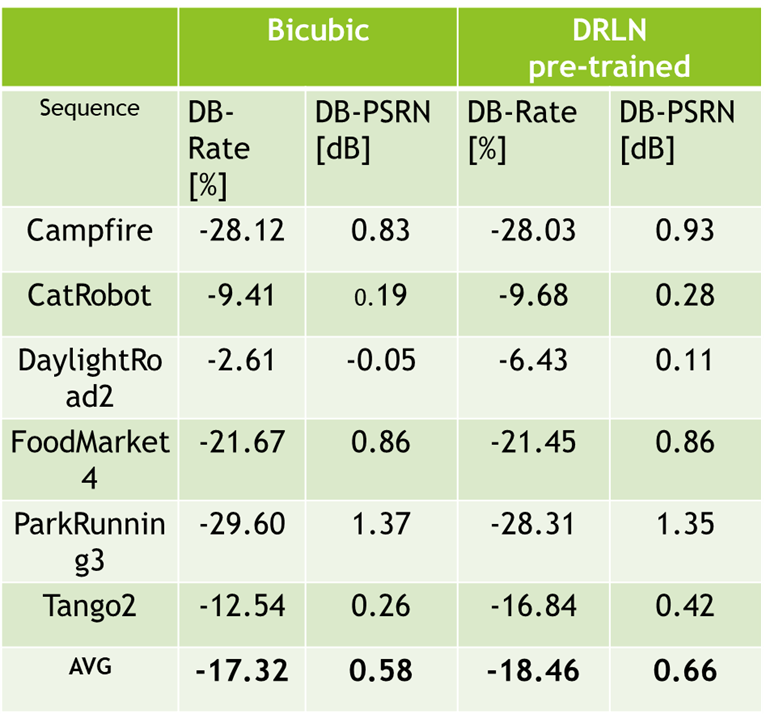


Table 3: BD rates bicubic vs. DLRN network

**In-loop filter**

Starting from the paper A Deep Learning Approach for Multi-Frame In-Loop Filter of HEVC we have started to go in-depth in the available on the git:

<https://github.com/tianyili2017/MultiFrame-InLoop-Filter>

This approach is implemented in HEVC and the plan is to port it into the EVC codec.

We reached out to one of the authors of the paper and shared insights on porting his code from HEVC to the basic EVC profile.

The preliminary results of the NN evaluation are shown in Table 4: ‘HEVC\_16.5’ is the encoding with HEVC HM 16.5 with all in-loop filters turned on (Deblocking and Sample Adaptive Offset filter). ‘HEVC CNN’ is the configuration that enhances the frame in which the neural network is always active; MIF-NET is the configuration in which a metric chooses among the best approaches. Table 4 shows that the MIF-NET configuration is the best because the PSNR is higher, with the same bitrate. CNN does a better job than HEVC filters. The assessment on the NN gives BD-rate: -9.26.

The group is working on the following topics:

* Initial investigation on the porting from HEVC to EVC
  + Issue on the partition file
  + Issue with 10 bit output from EVC

Immagine che contiene tavolo

Descrizione generata automaticamente

Table 4: Preliminary results of the assessment of the NN

**Inter prediction**

The group started the work on inter prediction with the aim of improving it by exploiting deep learning.

The group has chosen:

* Neural\_Reference\_Synthesis (NRS) for\_Inter\_Frame\_Coding

as a starting point for neural network applied to inter prediction. The code is available and we have contacted the authors to ask for support in better understanding the code.

We had a presentation by the author of the article who shared with us his findings and his thoughts on the transition from HEVC to EVC.

We invited the author to assist in integration of their NRS module into EVC. But they didn’t participate in the last 2 meeting cycles. So, we have planned to contact the authors of further shortlisted papers/journals promising inter prediction improvements.

The next steps are:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tool** | **Date** | **Topic** | **Who** |
| In-loop | 1 meeting cycle | Start porting from HEVC code to EVC | Attilio, Alessandra, Roberto |
| SR | 2 meeting cycle | New training exploiting the 10-bit training sources | Alessandro, Mattia, Gioele, Giovanni |
| Combined results | 1 meeting cycle | Testing combined results: intra + super resolutions | Alessandro, Giovanni, Attilio |
| Inter prediction | 2 meting cycle | Contact the authors of further shortlisted papers/journals promising inter prediction improvements | Gopi |

**Future Plan**

* motion compensation: improve the motion compensation using NN architecture
* quantization: uniform scalar quantization used in classical video codec standards does not conform to the characteristics of the human visual system. It is possible to use a quantization strategy based on neural networks.
* arithmetic encoder: improve the CABAC performance by leveraging NN to directly predict the probability distribution of intra modes instead of the handcraft context models