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

**Public document**

|  |  |
| --- | --- |
| N1007 | 2022/12/21 |
| 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 method we propose 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 con 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 training of 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 generalization 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).

**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 path:

* conversion from YUV to RGB (PNG) format and then applying the patch selection for each sequence as part of the original compressed dataset.
* Real-time extraction of the Y component (without writing to hard disk in image format) and feeding the neural network with the Y component replicated 3 times (to fill the 3 channels 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 generalization 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 analyze the test sequences to see if there are any potential problems, so that we will be ready after training to run the test.

**Training Strategies**

We have identified different types of training strategies as follow:

1. Without training, so using only the weights of the original DRLN approach. This approach has been tested. However, the DLRN network has been modified to remove an intermediate layer that it was not used but was introducing memory management issues. This has the effect that using the original weights of the network will not map to the new DRLN network producing results that are not usable. This makes comparison no longer possible.
2. Training only of the lower QP. 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 generalization 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.

**Training results for the HD24K case**

We have carried out extensive training of the DRLN approach for super-resolution using the new datasets as defined in the Common Test Conditions, where the sequences were encoded at QPs: 22, 27, 32, 37, 42, 47. Based on the preliminary results, as shown in the previous sections, the training has been carried out using patches from the lowest QP, in this case 22. The number of epochs used were 50, as in the preliminary experiments.

The results of the training for the HD24K are shown in the figure 7 and figure 8, representing the loss function and the PSNR of the training and validation respectively:

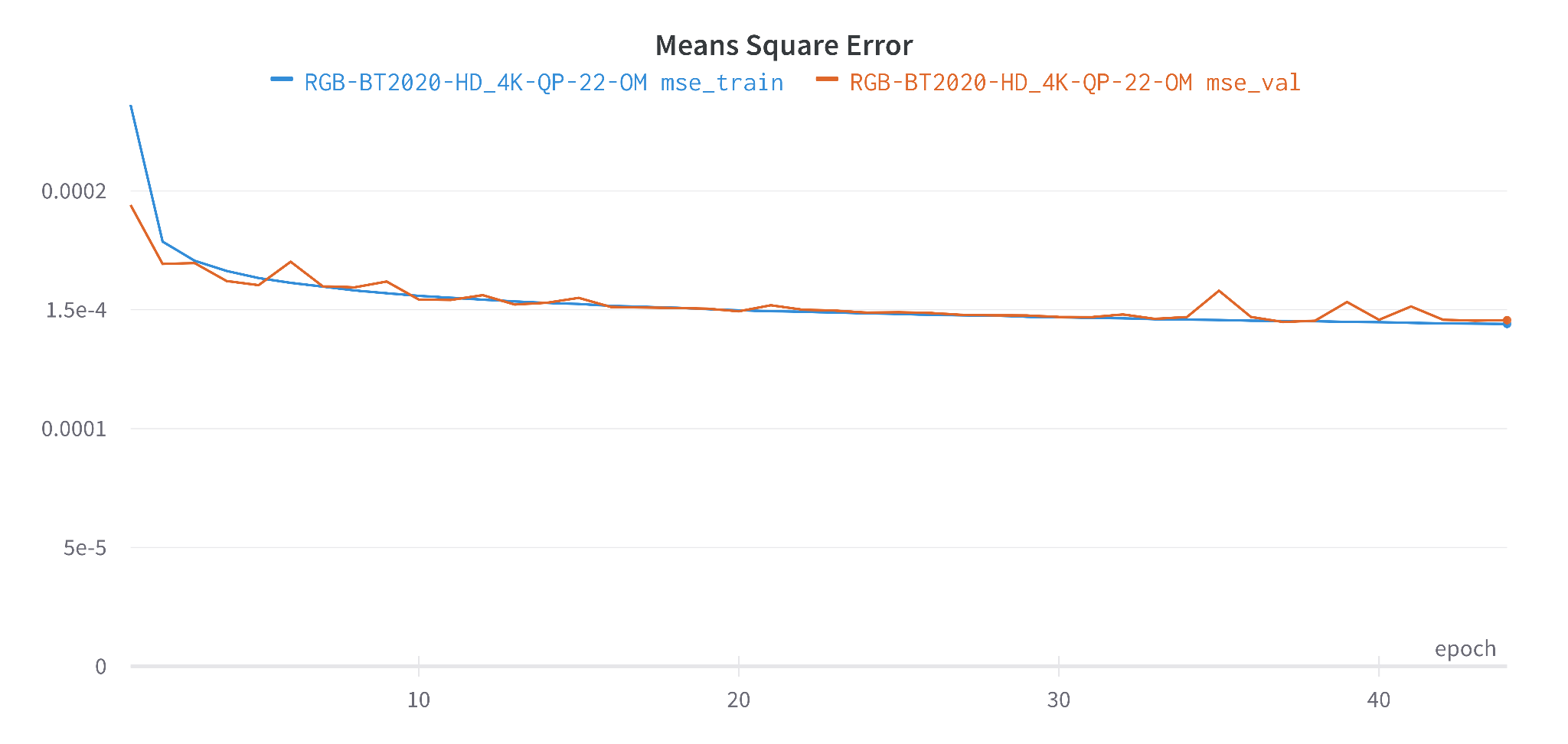


Figure 7: Loss function: Training and validation results of the HD24K case.

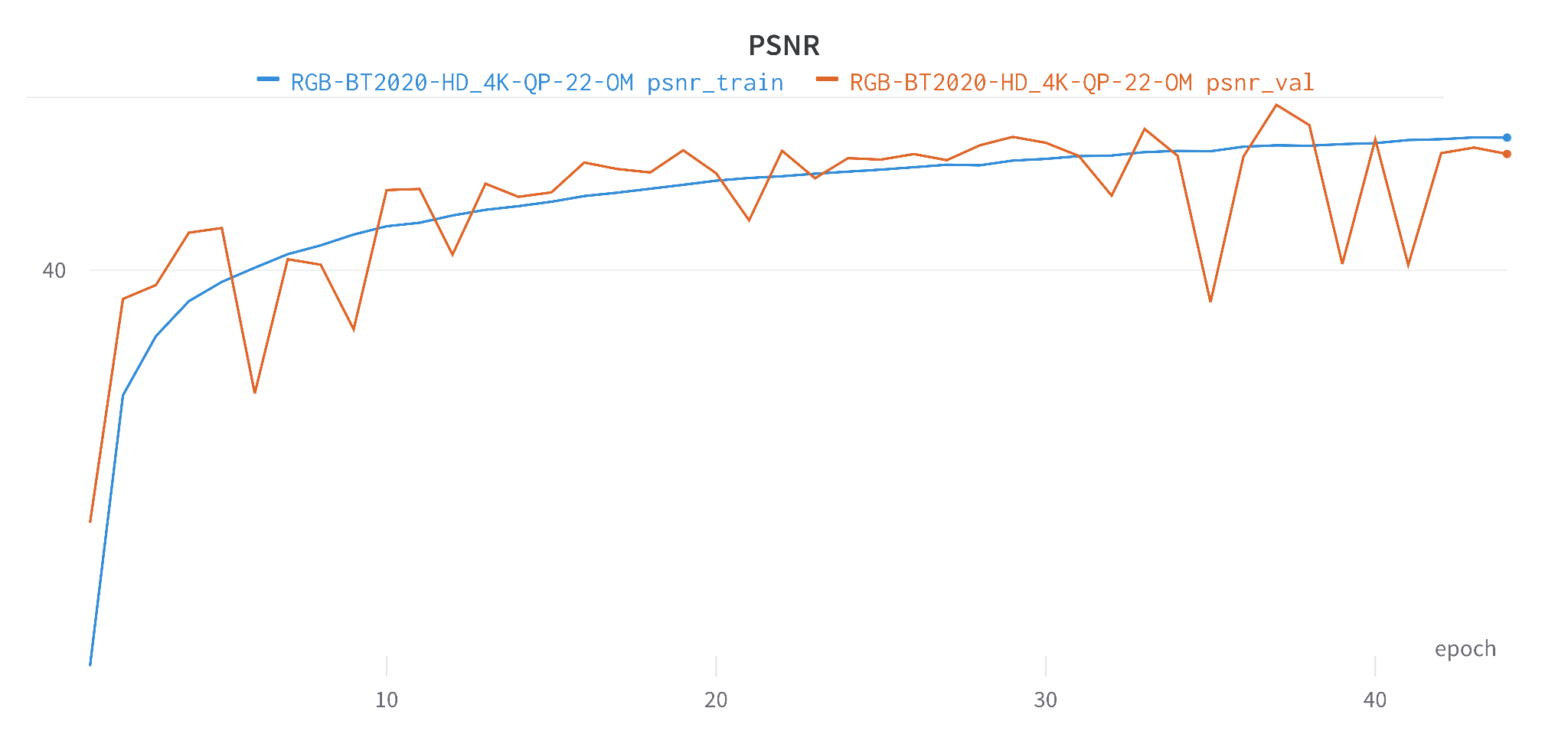


Figure 8: PSNR: Training and validation results of the HD24K case

**Training results for the SD2HD case**

We finished the training of the SR neural network on the task SD to HD. As in the previous section, training was carried out using the lowest QP patches (QP 22). Figures 9 and 10 show the loss function and PSNR of training and validation respectively up to epoch 50.

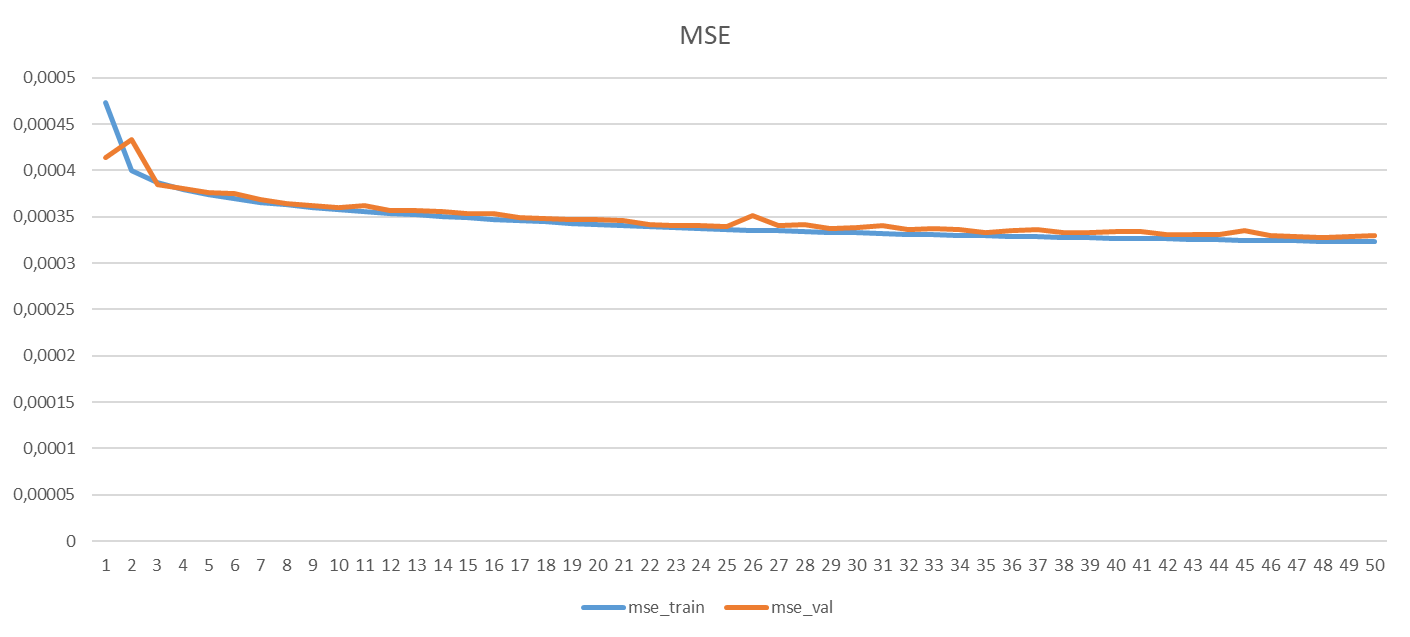


Figure 9: Mean Square Error (MSE) loss for training and validation of the SD2HD case

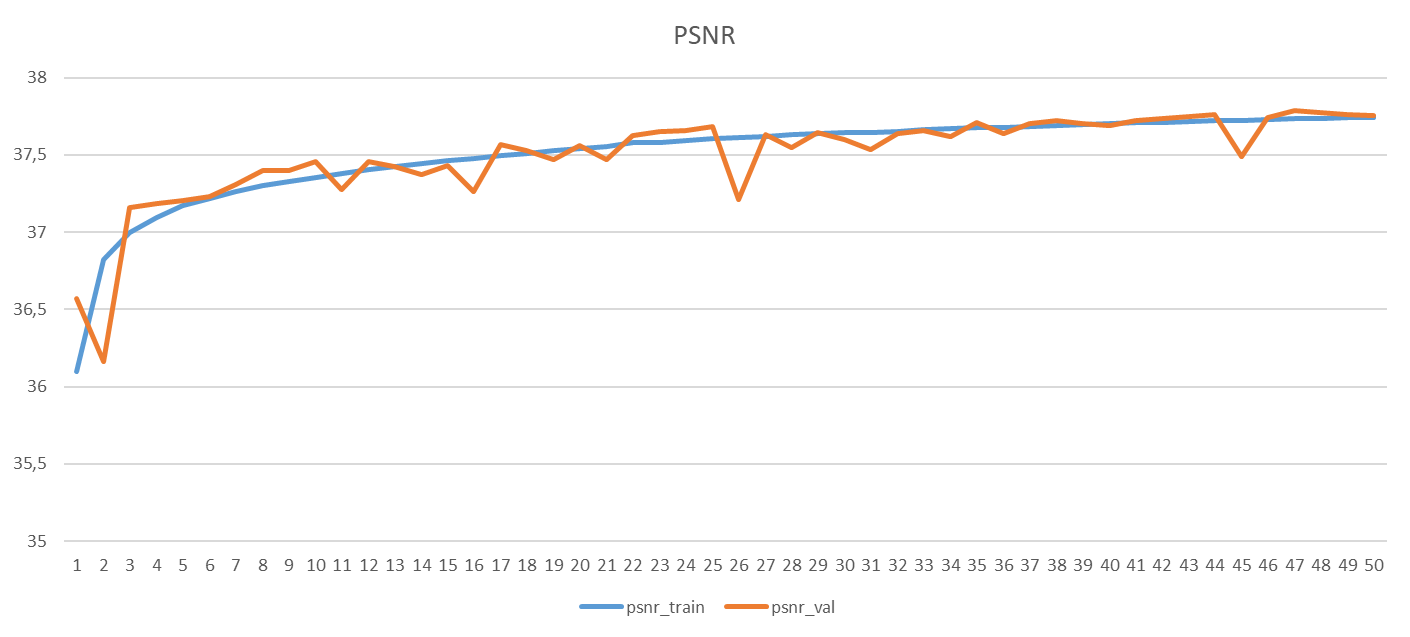


Figure 10: PSNR for training and validation of the SD2HD case

**Inference on the new dataset**

The inference phase on the new dataset is still ongoing. Tests have been carried out but we are checking their validity.

**Intra & super resolution combined**

The group started to test the pipeline with the two combined tools: Intra and Super resolution (Figure 11). So far, the percentages available for the Intra tool and the super resolution tool have been summed linearly, but the underlying process is strongly not linear. For this reason it was necessary to have a combined result to really calculate the actual coding gain.

In the combined tests, the picture is downscaled before being compressed with EVC Intra enhanced. At the output of the decoder the SR tool is applied to get the native full resolution. We are currently working only on the first frame of the sequence. The QPs under consideration are: 22, 27, 32, 37, 42.

The comparison is between the original full resolution and the upscaled picture.

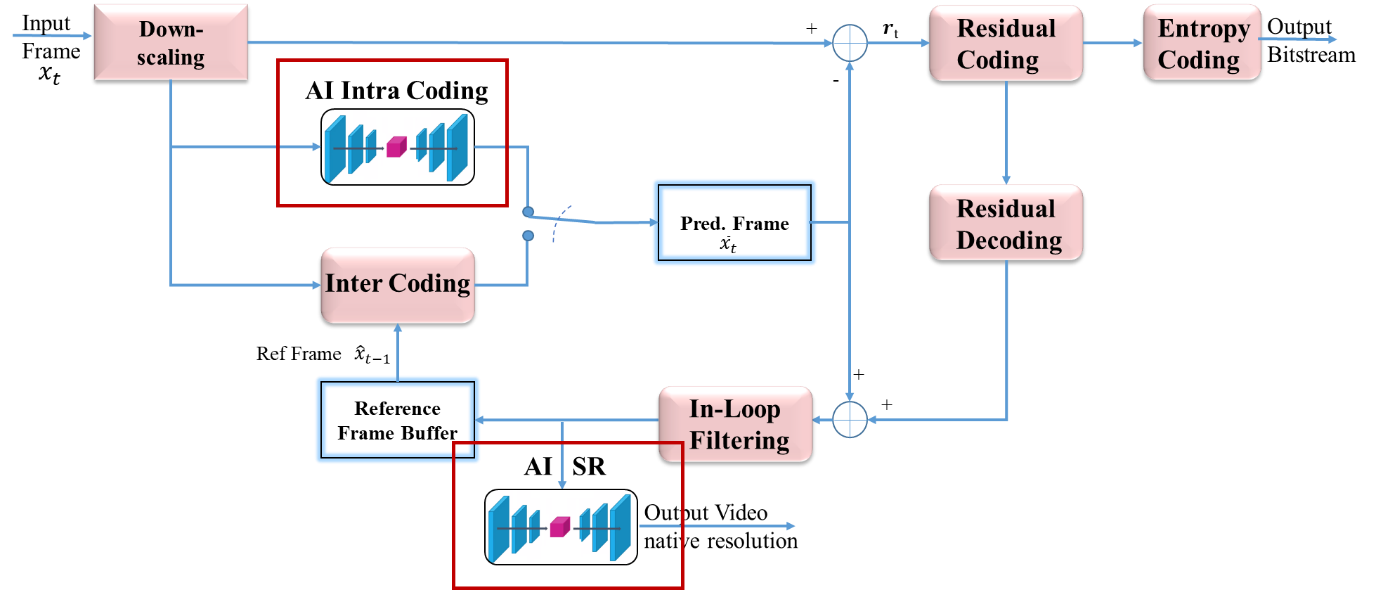
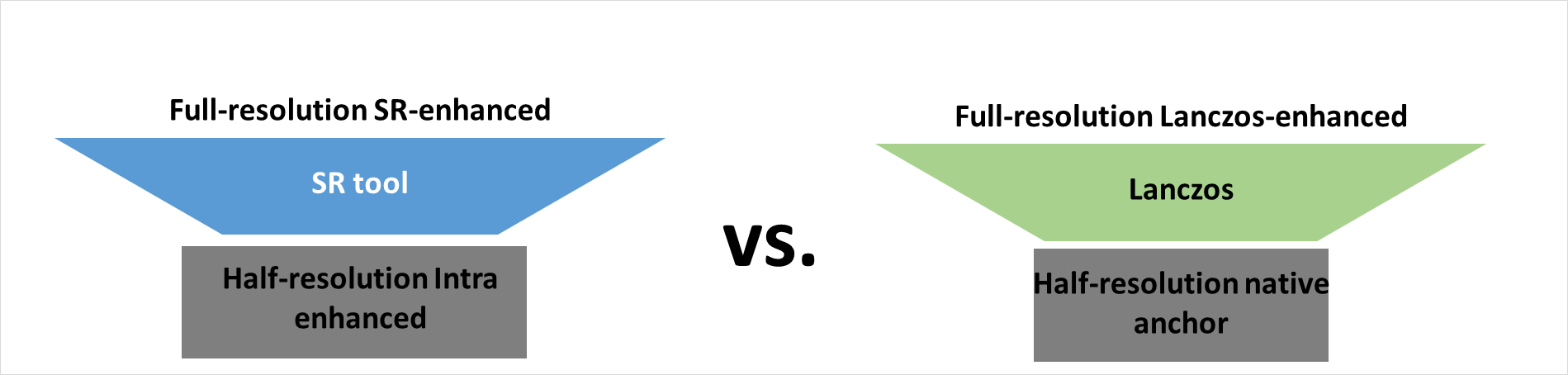


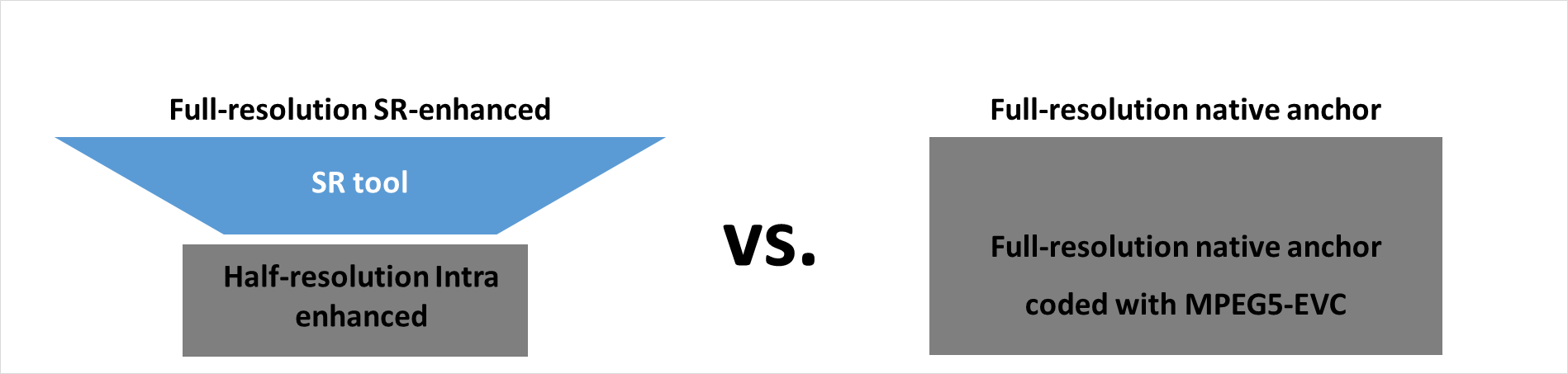
Figure 11 : synergy plan between Intra and Super resolution tools

We planned to perform the following tests:

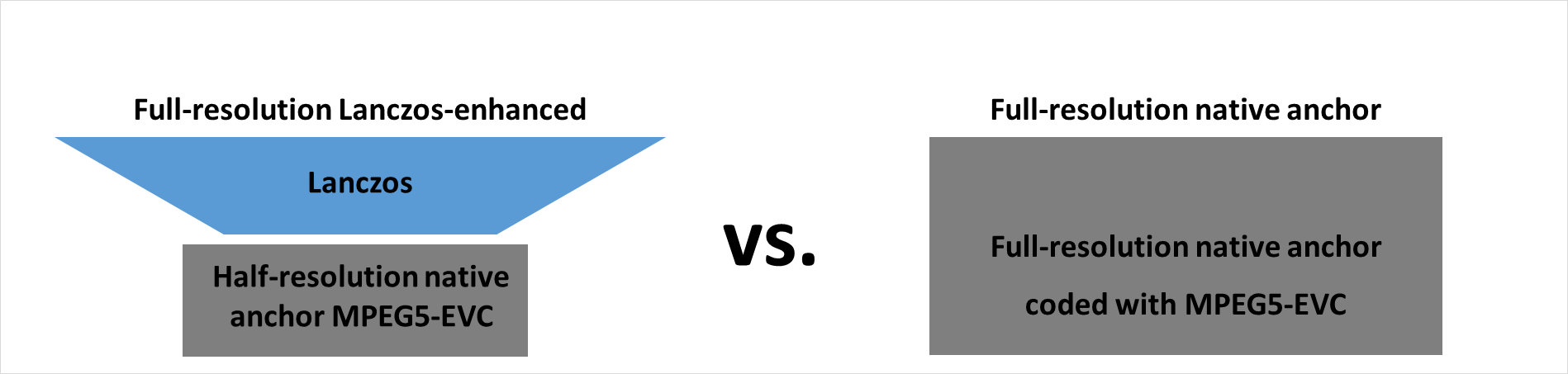
* 1. Test 1:
     + HD Intra enhanced (with deep learning) EVC to 4K upsampled with SR tool
     + HD MPEG5-EVC upsampled to 4K with Lanczos



* 1. Test 2:
     + HD Intra enhanced (with deep learning) EVC to 4K upsampled with SR tool
     + 4K native full resolution coded with MPEG5-EVC



* 1. Test 3:
     + HD MPEG5-EVC upsampled to 4K with Lanczos
     + 4K native full resolution coded with MPEG5-EVC



We have started the HD to 4K testing phase on all QPs (15,30,37 and 45).

Figure 12 shows the BD-rate curves for each of the test sequences, where the network trained on QP 15 dataset has been used over the test set for all QPs, as already explained for the previous experiments.

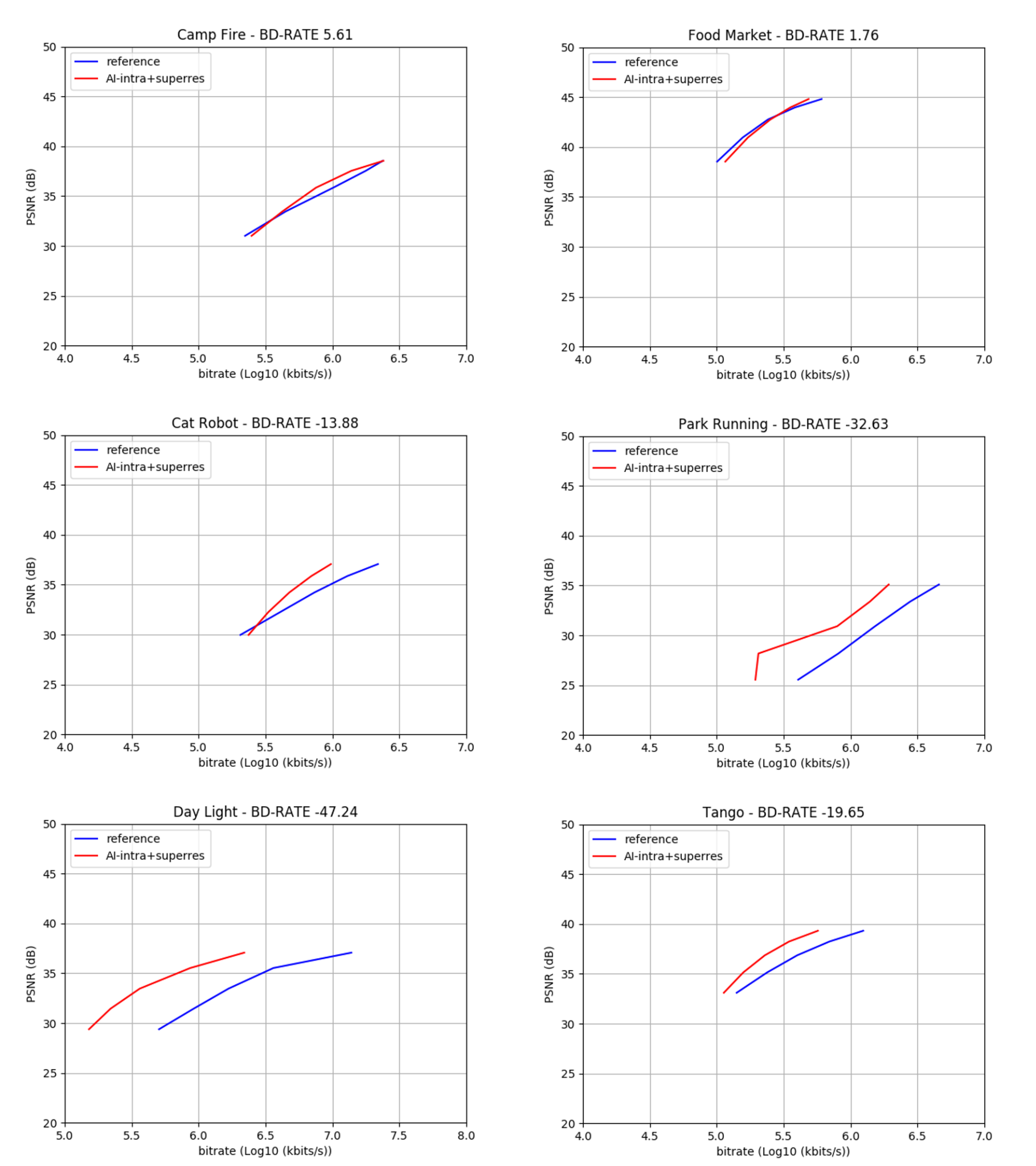


Figure 12: results of preliminary BD-Rate evaluations

The group discussed further testing the combination of the Intra and the SR tools for a random access configuration. The group agreed upon the configuration settings, including the GOP structure (9 frames, IBBBBBBBB scheme). The encodings on the JVET Class A (4K) for the Intra-augmented setup are finished.

Due to the long SR training time, these experiments are not expected to yield until the end of November at the earliest. For this reason, we performed preliminary experiments where Class A sequences downsampled to HD and encoded using the NN-Intra toolwere upsampled using the Lanczos filter, restoring the original 4K resolution. We compared this scheme with a reference EVC codec where Class A sequences were encoded at full 4K resolution as shown in (Figure 13). The corresponding results are shown in Table 3 and show BDrate gains around 16%: comparing to NN-Intra standalone gains in Table 1, this experiment brings an additional 10% BDrate gain atop of that. These preliminary experiments show that combining NN-based Intra prediction with the Lanczos upsampling filter yields significant gains on 4K sequences and suggest that further gains are expected when the SR tool takes the place of Lanczos. Once the SR tool training is completed, the decoded sequences will be upsampled to the original 4K using the SR tool rather than Lanzos and final RD curves will be plotted and BDRate values calculated to quantify such additional gains.

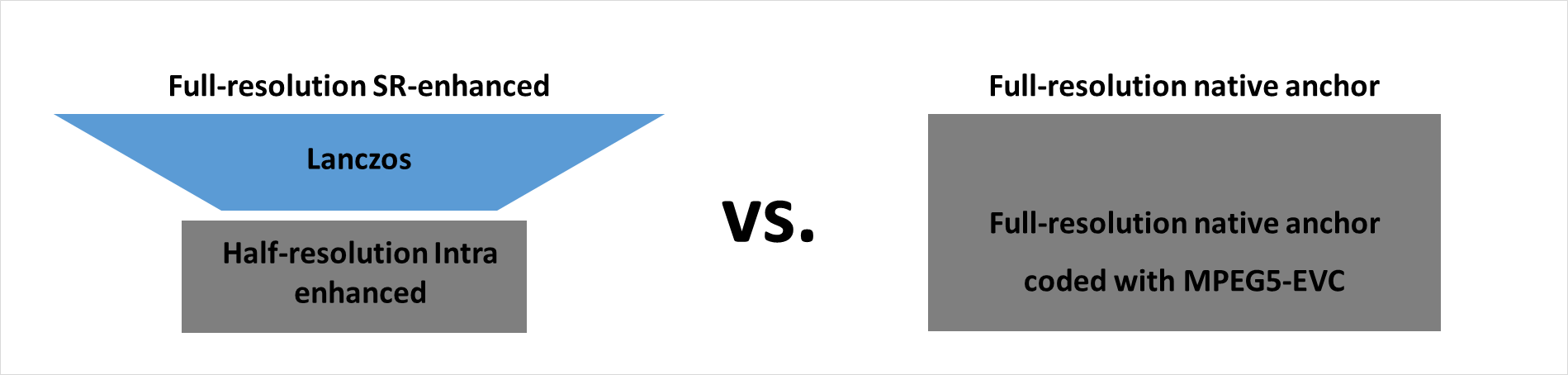


Figure 13: intermediate experiments

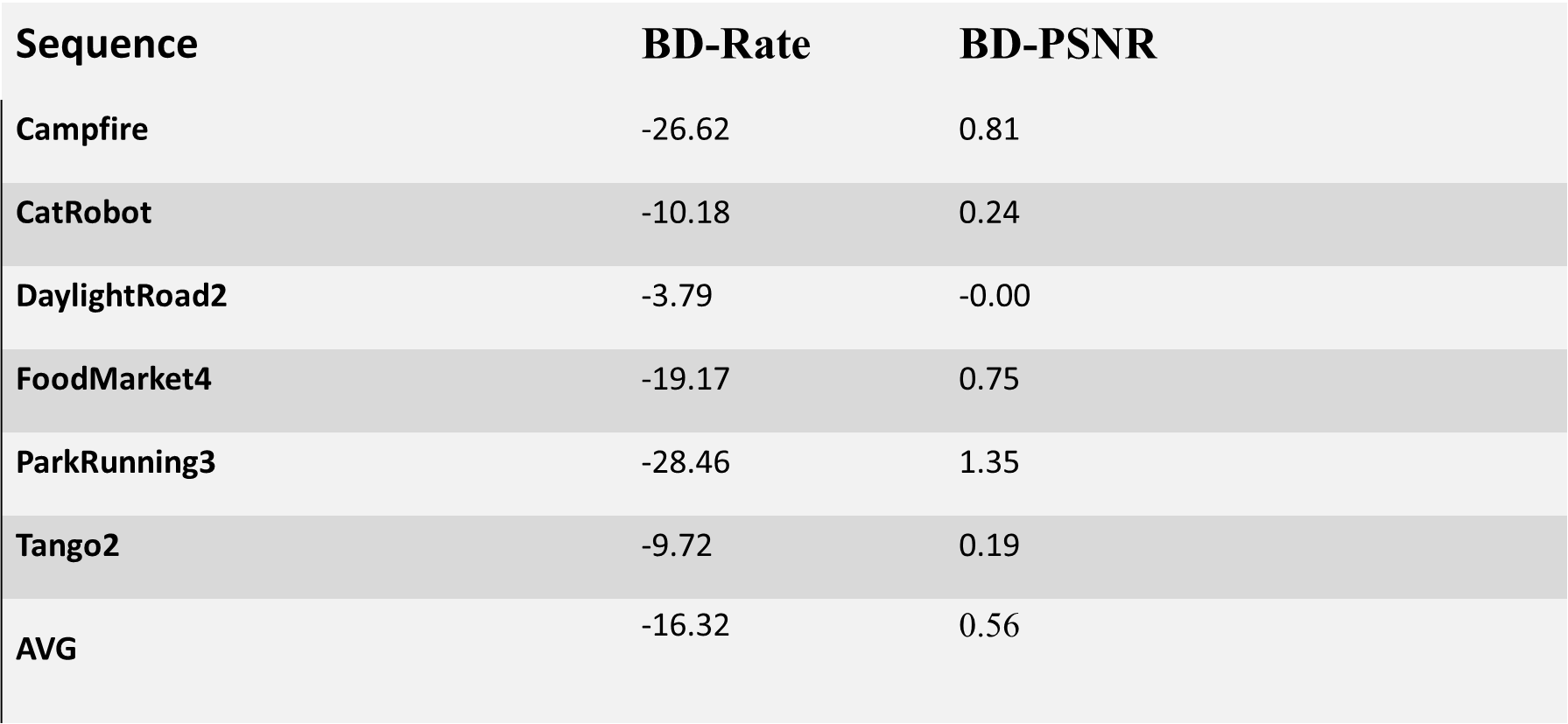
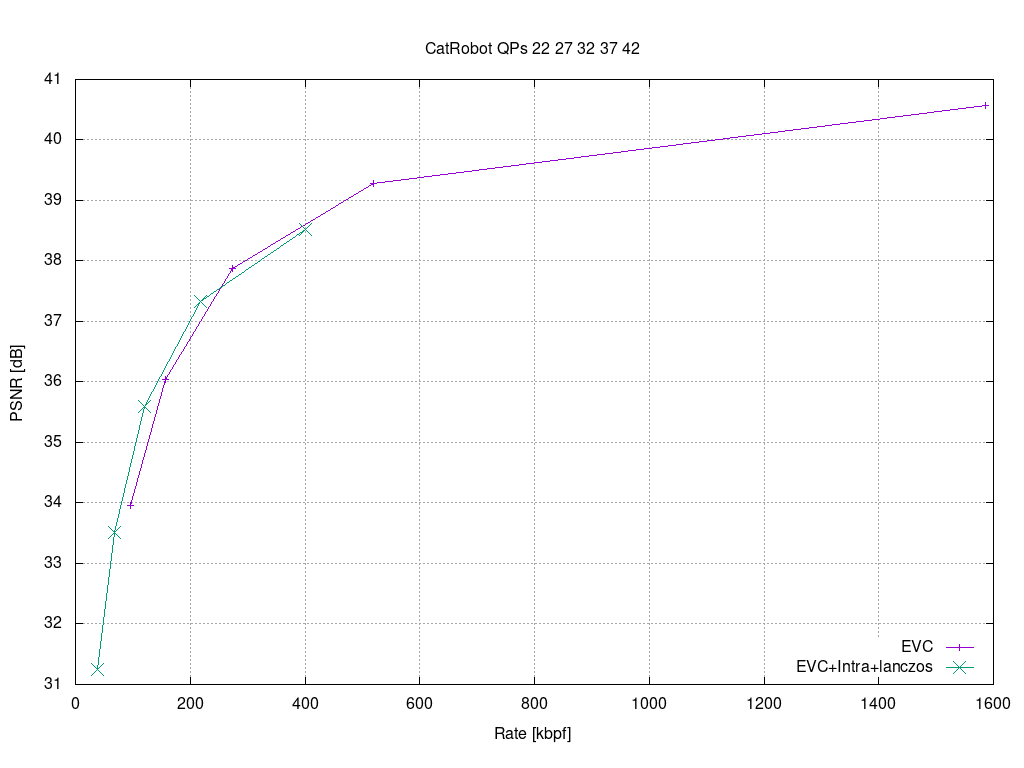
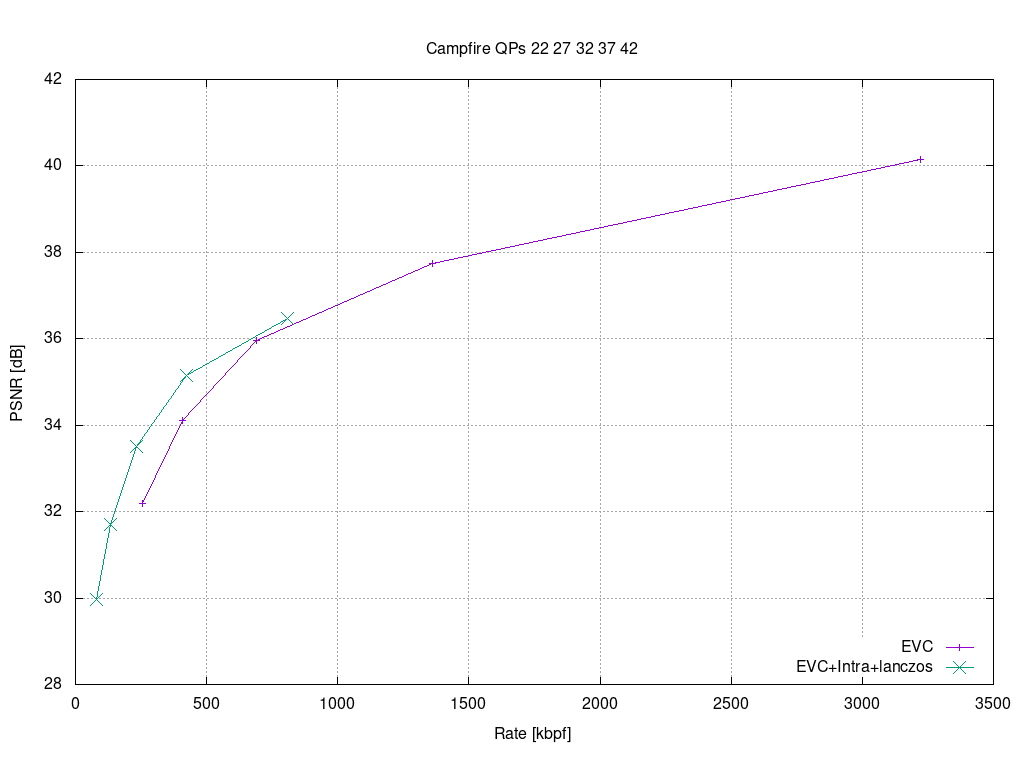
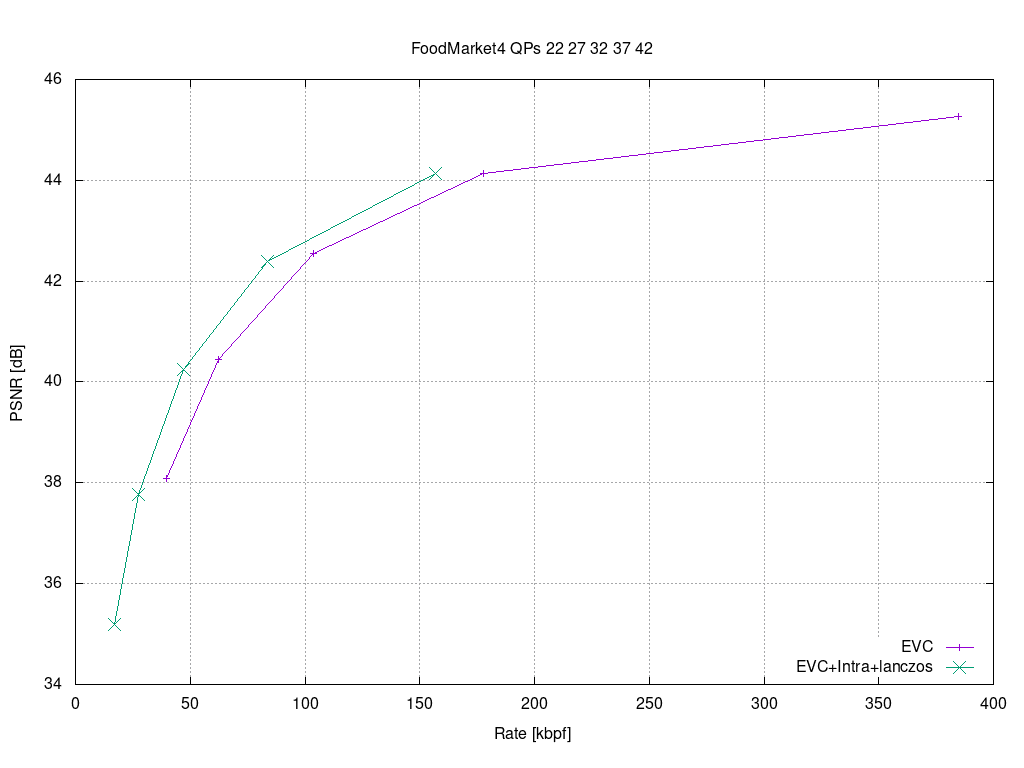
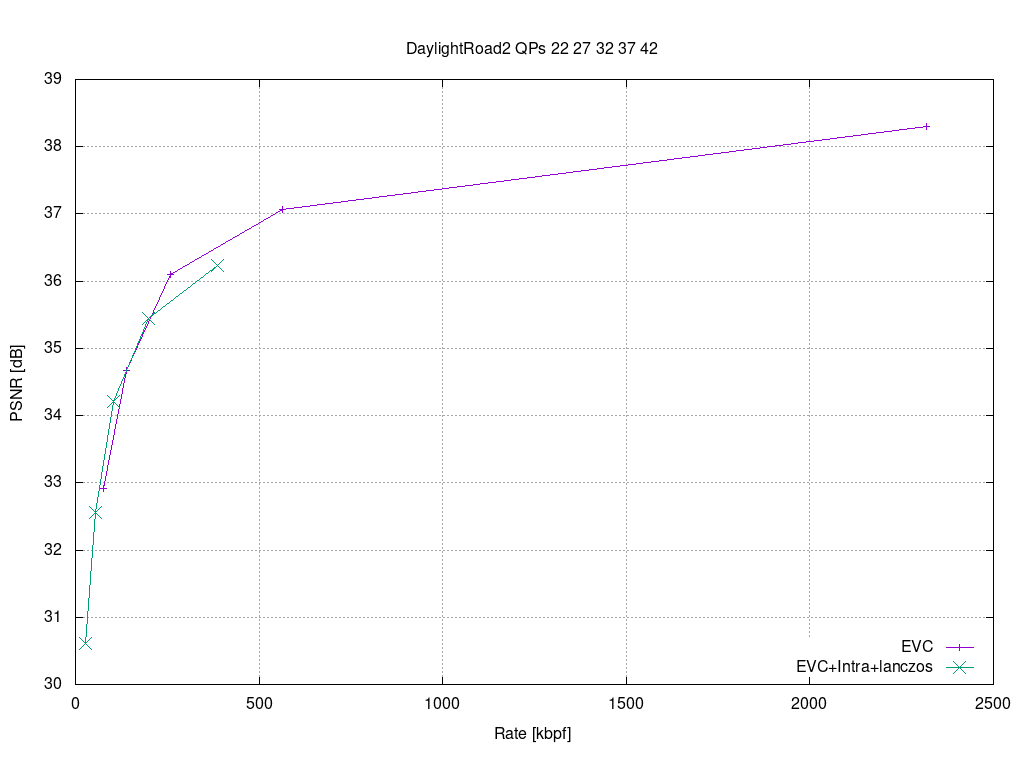


Table 3: Intra enhancement + upscaling with Lanczos





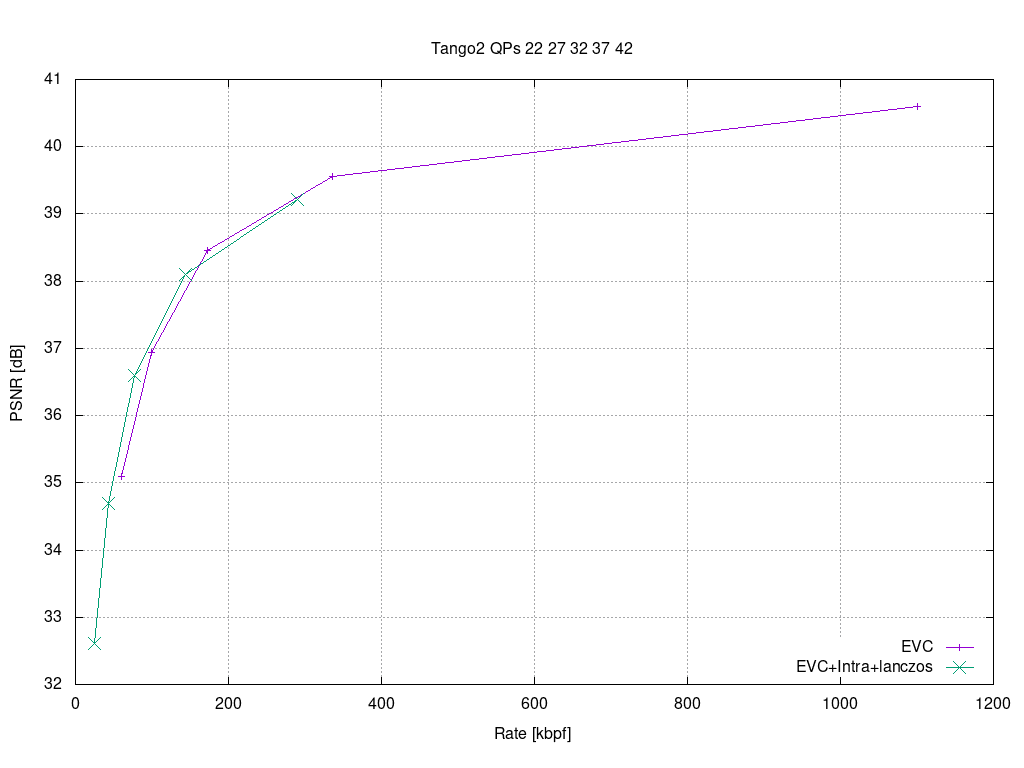
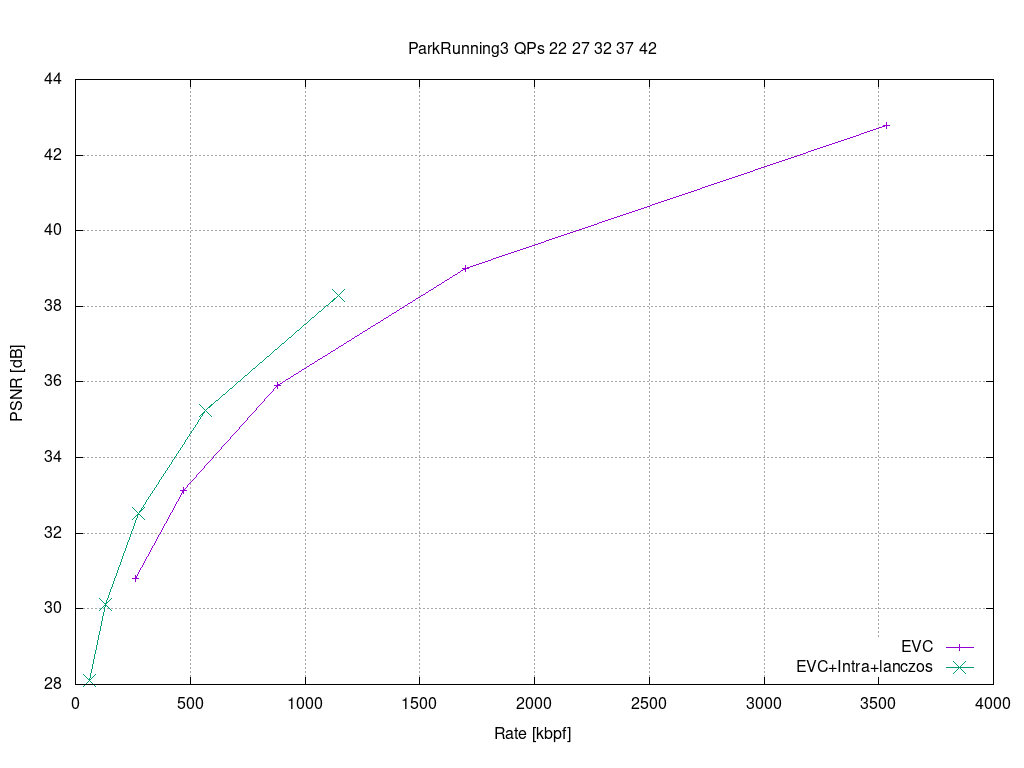


Figure 14: R-D curves for the Intra enhancement + upscaling with Lanczos

**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 group is working on the following topics:

* Initial investigation on the porting from HEVC to EVC
* The communication between HEVC and the neural network
* Drive the neural network without HEVC code
* Assessment of the NN

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.

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\_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 HVC to EVC.

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 authors of ‘Neural\_Reference\_Synthesis\_for\_Inter\_Frame\_Coding’ | 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