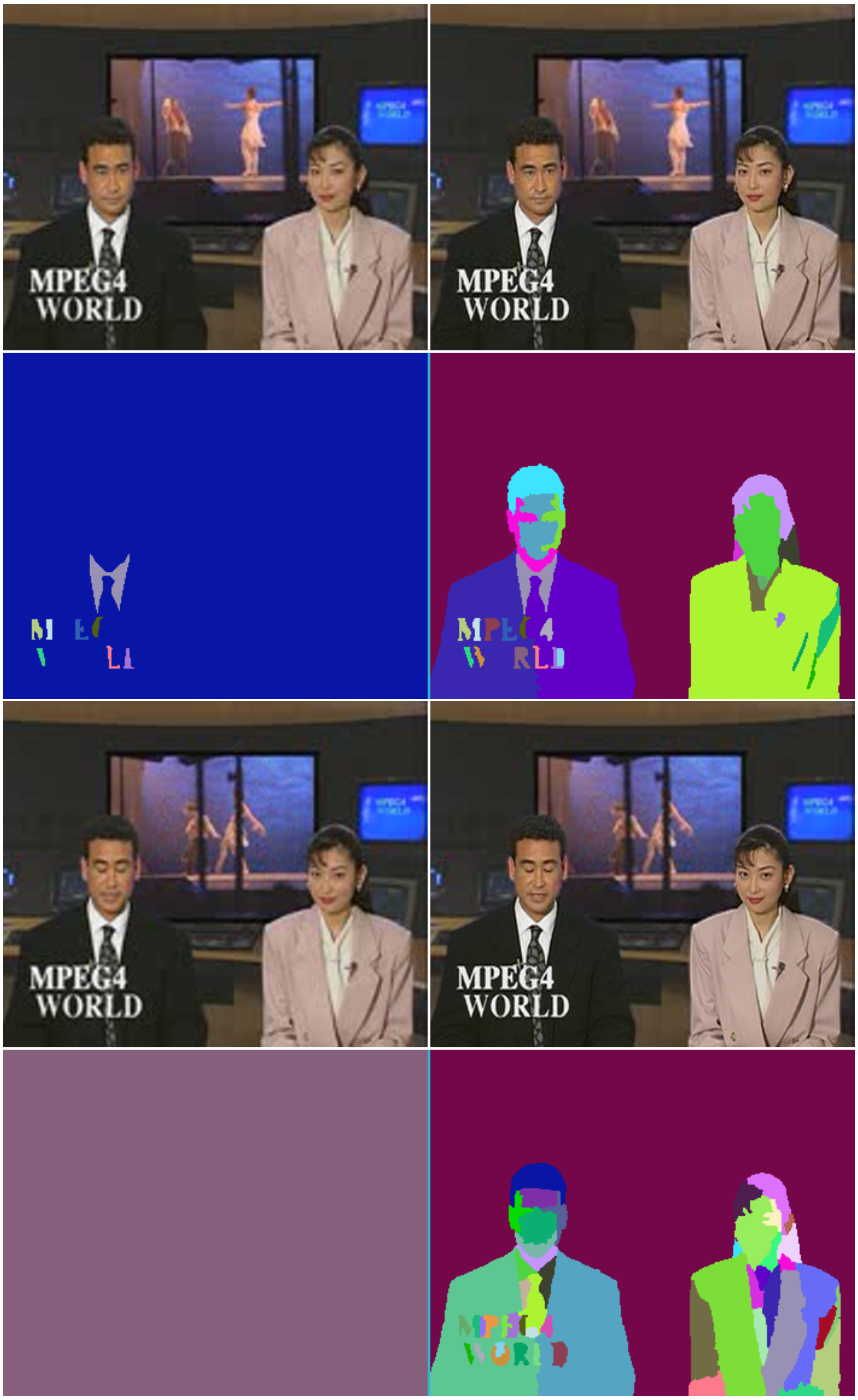


Figure 9.19: Example of Spatial scalability: News sequence (CIF) coded at 10Hz, object functionalities mode, 128 kbps (base layer, QCIF) + 384 kbps (enhancement layer, CIF). Figure shows base (up-sampled by a factor two) and enhancement layers in the first row and the corresponding partitions in the second row for frames #0 and #60

Figure 9.20: Example of Spatial scalability: News sequence (CIF) coded at 10Hz, object functionalities mode, 128 kbps (base layer, QCIF) + 384 kbps (enhancement layer, CIF). Figure shows base (up-sampled by a factor two) and enhancement layers in the first row and the corresponding partitions in the second row for frames #120 and #180

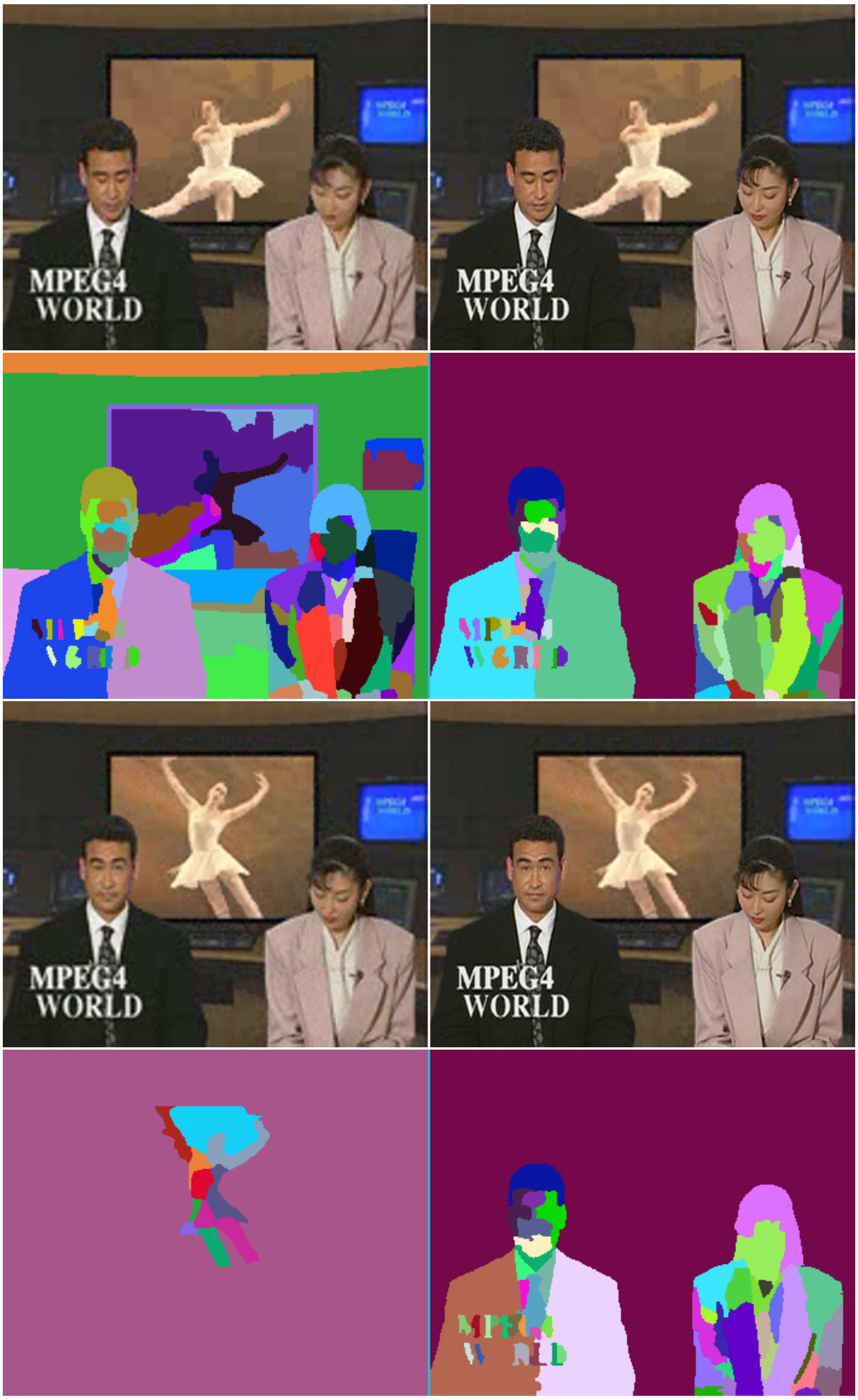
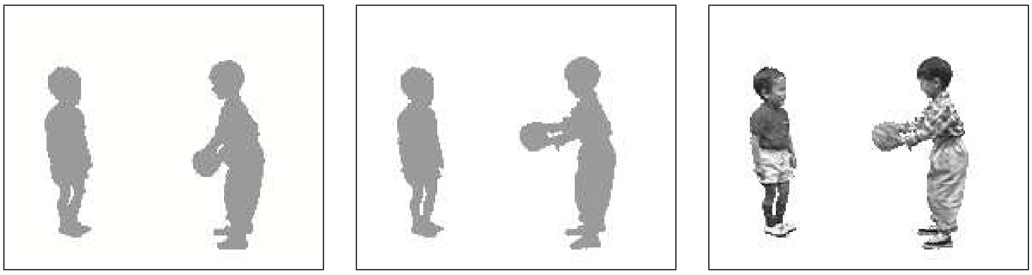
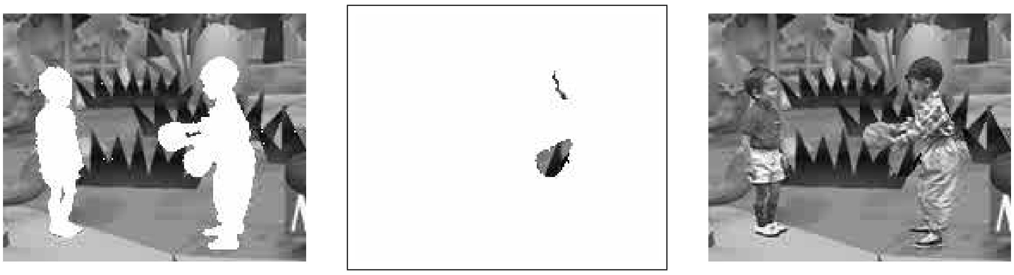


Figure 9.21: Example of Spatial scalability: News sequence (CIF) coded at 10Hz, object functionalities mode, 128 kbps (base layer, QCIF) + 384 kbps (enhancement layer, CIF). Figure shows base (up-sampled by a factor two) and enhancement layers in the first row and the corresponding partitions in the second row for frames #240 and #282



(a) Selection mask #(6) (b) Selection mask #(9) (c) Coded object



(d) Coded background (e) Coded uncovered area (f) Final result

Figure 9.22: Example of uncovered background in temporal scalability, object functionalities mode. Frames #(6) and #(9) of the *Children* sequence have been used. Both the base layer and enhancement layers have been coded at 5Hz

9.3 Temporal scalability

Uncovered background

Figure 9.22 shows the strategy adopted for uncovered areas (see Section 8.7.3) in temporal object scalability. Two frames are taken from the decoded sequence, one from the base layer (#6) and one from the enhancement layer (#9). In the figure, pictures (a) and (b) show the position of the object at the consecutive frames. In this mode, all the bit budget for the enhancement layer is used to code the selected object. The background is directly taken from the previous frame. This is illustrated in pictures (c) and (d). The first one shows the coded object at the enhancement layer (frame #9); the second one shows the background for the same frame, copied directly from frame #6. In (d) it can be seen that the motion of the object leaves an uncovered region in the background, that is, a zone for which no reference data are available. If the deformation of the object is small, these zones of uncovered background are tiny and can be filled by extrapolating neighboring pixels. If the uncovered regions are large, they are filled by motion compensating the texture of the neighboring regions in the previous frame. This approach uses a few motion vectors for the background, but avoids unpleasant results. Picture (e) shows the uncovered background reconstructed by motion compensation, and finally, picture (f) shows the final result.

Results for Temporal scalability

Performance of the temporal scalability mode is presented in Figure 9.23. The rate-distortion curves are plotted for the sum of the base and enhancement layers. As each layer is encoded at 15Hz, this results in a 30Hz sequence (in this case, the enhancement layer encodes the frames skipped at the base layer).

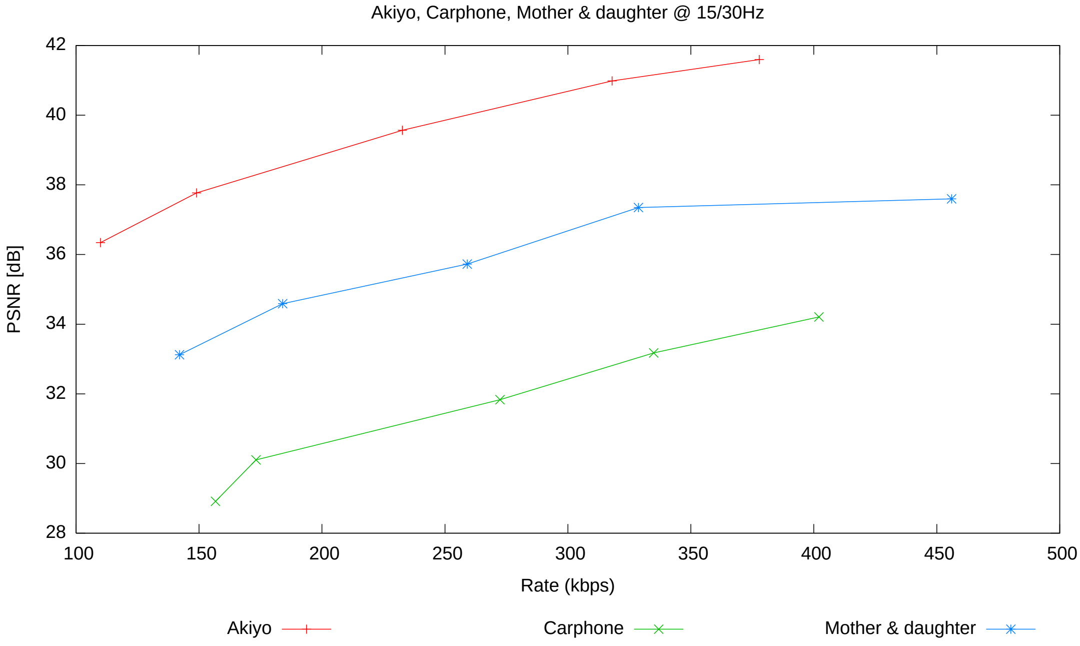


Figure 9.23: Example of temporal full frame scalability: Akiyo, Carphone and Mother and daughter sequences (QCIF). The Figure shows the result of decoding the base (15Hz) and enhancement layers (15 Hz).

Full frame mode - Summary of results: For full frame mode, details about the shape of partitions and the texture coding techniques being used in temporal scalability do not differ with the details already discussed for the non-scalable encoder (see Section 6.6).

Object functionalities mode - Summary of results: In object functionalities mode, the need to include all the contours of the object can force a large number of regions in the enhancement layer (see discussion in section 9.1). Other than this, no significant changes are observed with respect to full frame mode.

9.4 Comparison between dependent optimization and independent optimization

Here, some results are presented to show the difference between independent optimization and two-frame global or dependent optimization. We restrict the dependent problem to two frames or scalability layers because of the computational complexity which is exponential in the dependency tree. That is, in the PSNR and Spatial Scalability modes, the dependencies between the base and enhancement layers are taken into account. A global bitrate is given for both layers and the optimization algorithm selects the operating points that maximize the PSNR for the enhancement layer. An additional restriction is placed to ensure that the base layer has a minimum quality. In this case, the temporal dependencies between successive frames are not considered. In the temporal scalability case, the base and enhancement layers correspond to different time instants. In this case, temporal dependencies are considered between these two frames.

PSNR scalability:

For PSNR scalability (see Figure 9.24) the results show that the global coding is performing better than the separate optimization at all bitrates. Figure 9.24 shows the results for the enhancement layer of the *Akiyo* sequence coded at 10Hz. The average PSNR gain in the enhancement layer is between 1.2 and 1.5 dB over the independent optimization case.

In all the tests performed, the dependent optimization algorithm always assigns less bits to the base layer than the independent algorithm. Independent optimization test were forced to assign 50% of the bitrate to the base layer and 50% to the enhancement layer. The analysis of the results of test sequences at different bitrates show that the dependent optimization algorithm distributes the bit budget so that the base layer receives approximately 20% and the enhancement layer the remaining 80%. These figures are rather consistent between different test sequences and different bitrates. The number of regions is always lower (10% - 20%) in the dependent optimization case, both in the base and the enhancement layer.

Results for object functionalities mode PSNR scalability (see Figure 9.25) are similar to those in full frame mode scalability. In this case, the gain of the dependent optimization is even higher. This is because the inclusion of the object forces the presence of a larger number of regions in the enhancement layer partition. In this case, the strategy followed by the dependent algorithm consisting on using more bits in the enhancement layer is even more profitable.

In the enhancement layer there is almost no difference in the number of regions used by the dependent and the independent optimization approaches. This is because the need to include all the contours of the object does not leave as much freedom as in the full frame mode.

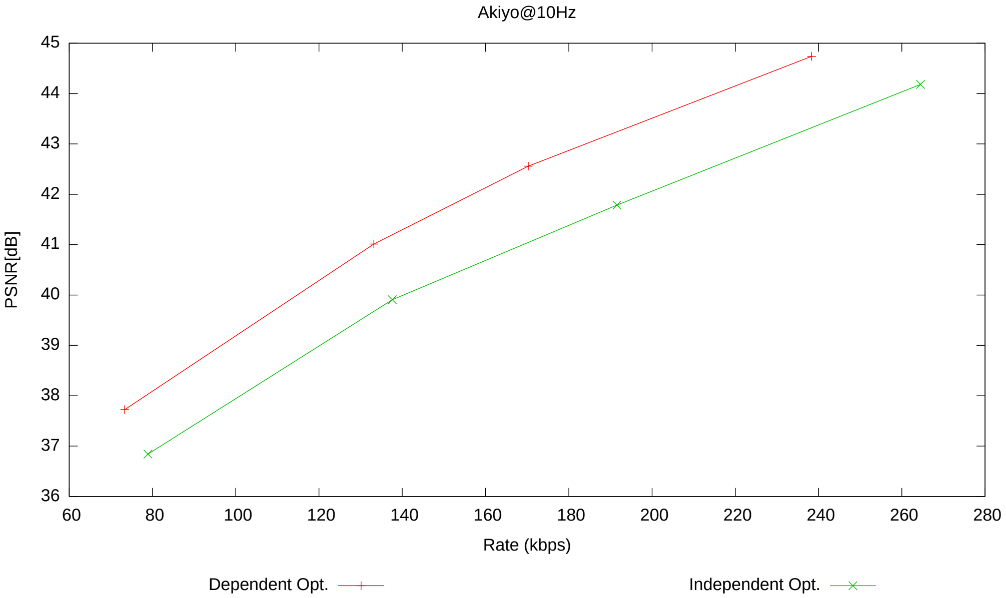


Figure 9.24: Comparison between dependent and independent optimization in full frame PSNR scalability.

Note that performance comparisons between dependent and independent optimization are difficult to make. If the comparison is performed frame to frame, the effects caused by dependencies between frames (the better quality of a given frame can improve the prediction of the following frames) can be lost. This would give results that do not fully reflect the improvement achieved by the dependent algorithm. If the comparison is performed on a full sequence basis, the meaningfulness of the approach is not clear because of the operational framework used in the optimization. Indeed, when in a given frame the partition selected by the dependent optimization algorithm is different from the one selected by the independent algorithm, the operational set of regions in the following frames will be different and either will the possible choices of the dependent algorithm. In all our tests the results for the two comparison approaches are almost the same. For the sake of simplicity, the latter approach has been used.

Spatial scalability:

For Spatial object scalability (see Figure 9.26) the results show that the global coding is performing better than the separate optimization at all bitrates. The figure shows the results for the object in the enhancement layer of the *News* sequence coded at 10Hz and 128 kbps for the base layer. The average PSNR gain in the enhancement layer is higher at low bitrates.

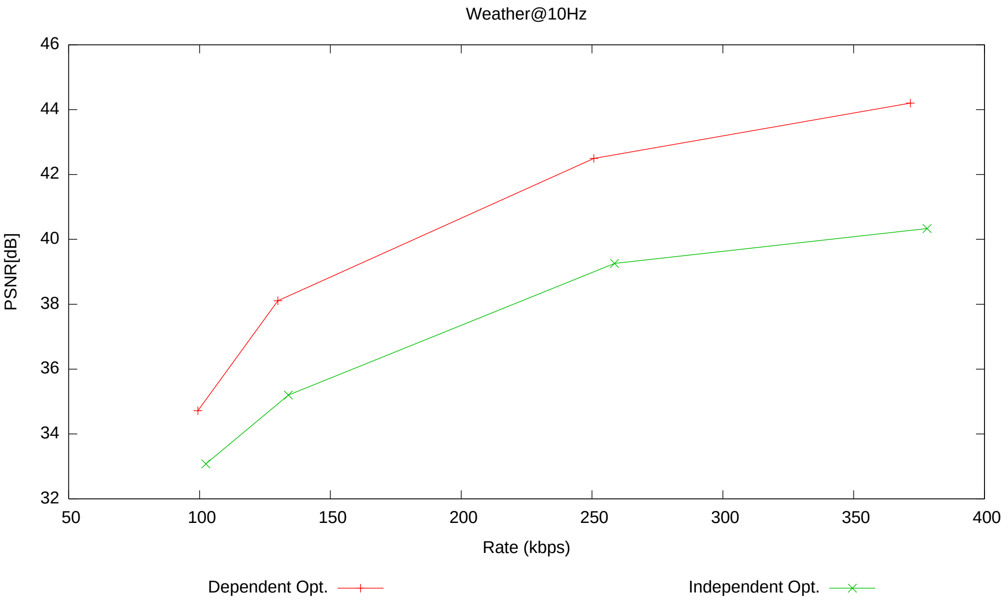


Figure 9.25: Comparison between dependent and independent optimization in object functionalities mode PSNR scalability

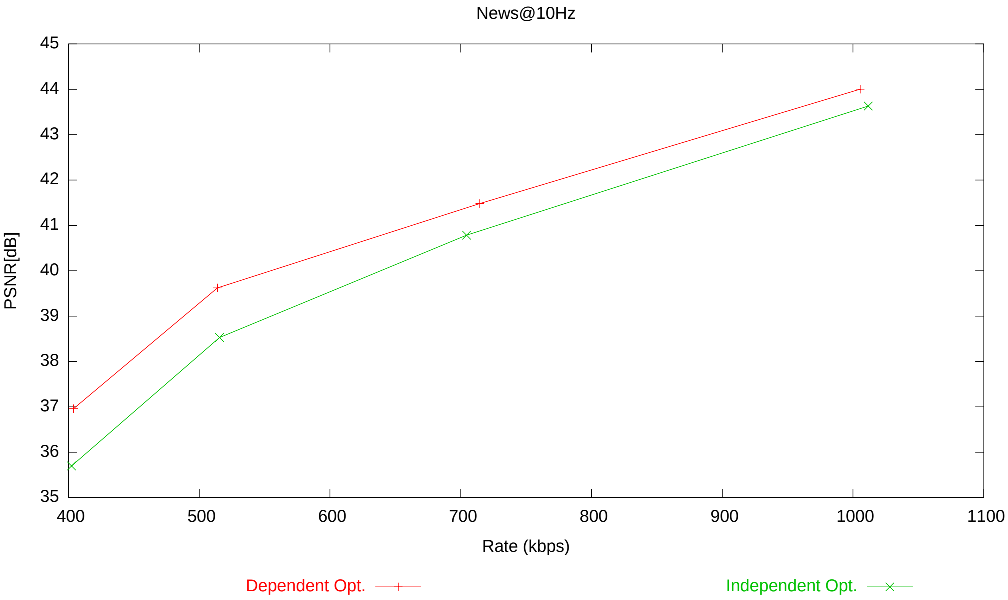


Figure 9.26: Comparison between dependent and independent optimization for the News@10Hz sequence, Spatial object scalability

Temporal scalability:

In the temporal scalability case the optimization algorithm seeks to maximize the average PSNR between the base and the enhancement layer. In this case, the results show that the bit budget is almost evenly distributed between the base and the enhancement layers. The differences in the number of regions between the dependent and the independent optimization cases are small. However, the dependent optimization algorithm performs significantly better (see Figure 9.27).

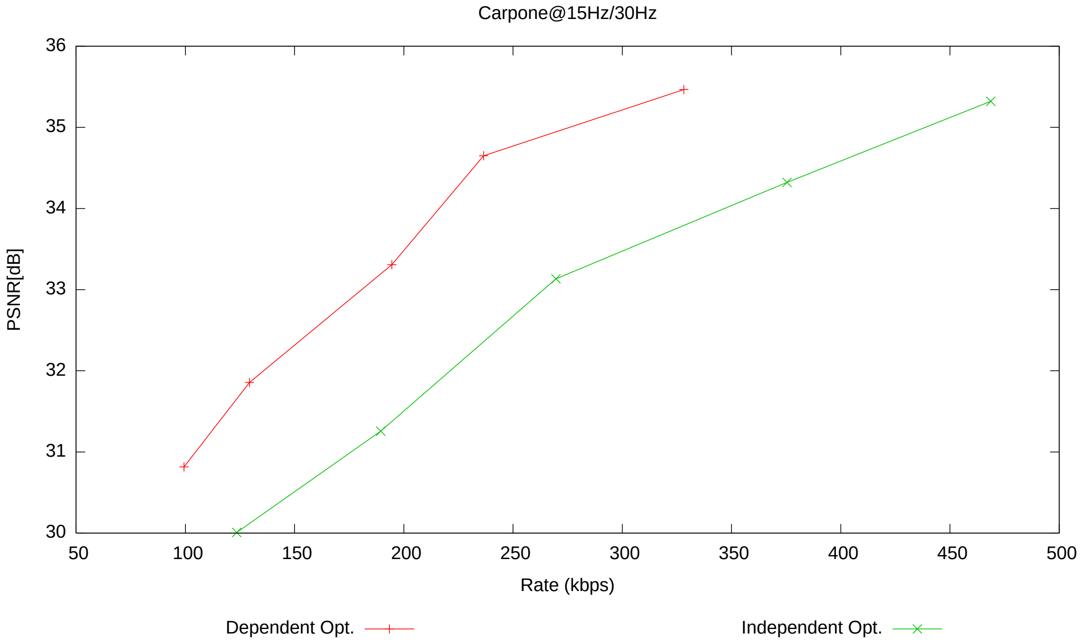


Figure 9.27: Comparison between dependent and independent optimization for the Carphone@10Hz sequence, Temporal full frame scalability

The results show that the proposed schema of dependent optimization is performing well in all the cases. The drawback of this approach is its high complexity.

Computational complexity

The computational cost of the dependent optimization algorithm is much higher than the cost of the independent one ($\tilde{2}0$ times). This larger complexity currently limits the usefulness of dependent optimization.