May 2023

AI-BASED LIGHT PARALLEL VIDEO ENCODER

Marwa TARCHOULI

Marc Rivière
Thomas Guionnet
Mickael Raulet

Meriem Outtas
Olivier Deforges
Wassim Hamidouche

ATEME
Captivate your audience

IETR

OUTLINE

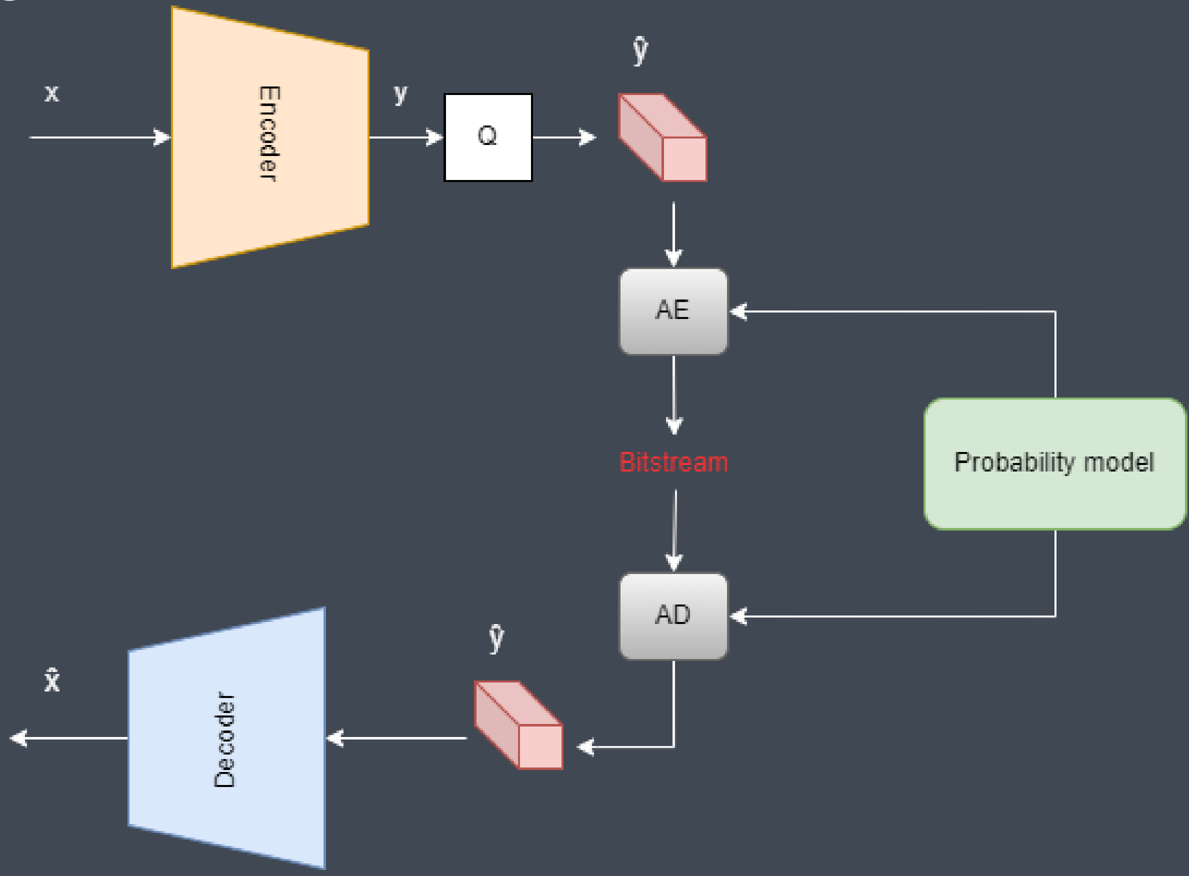
1. Introduction to learned video coding
2. Learned patch-coding using overlapping
3. Adjustment of MS-SSIM implementation
4. Parallel Neural Video Coding (PaNVC) Framework
5. Conclusion

ATEME

1 Introduction to learned video coding

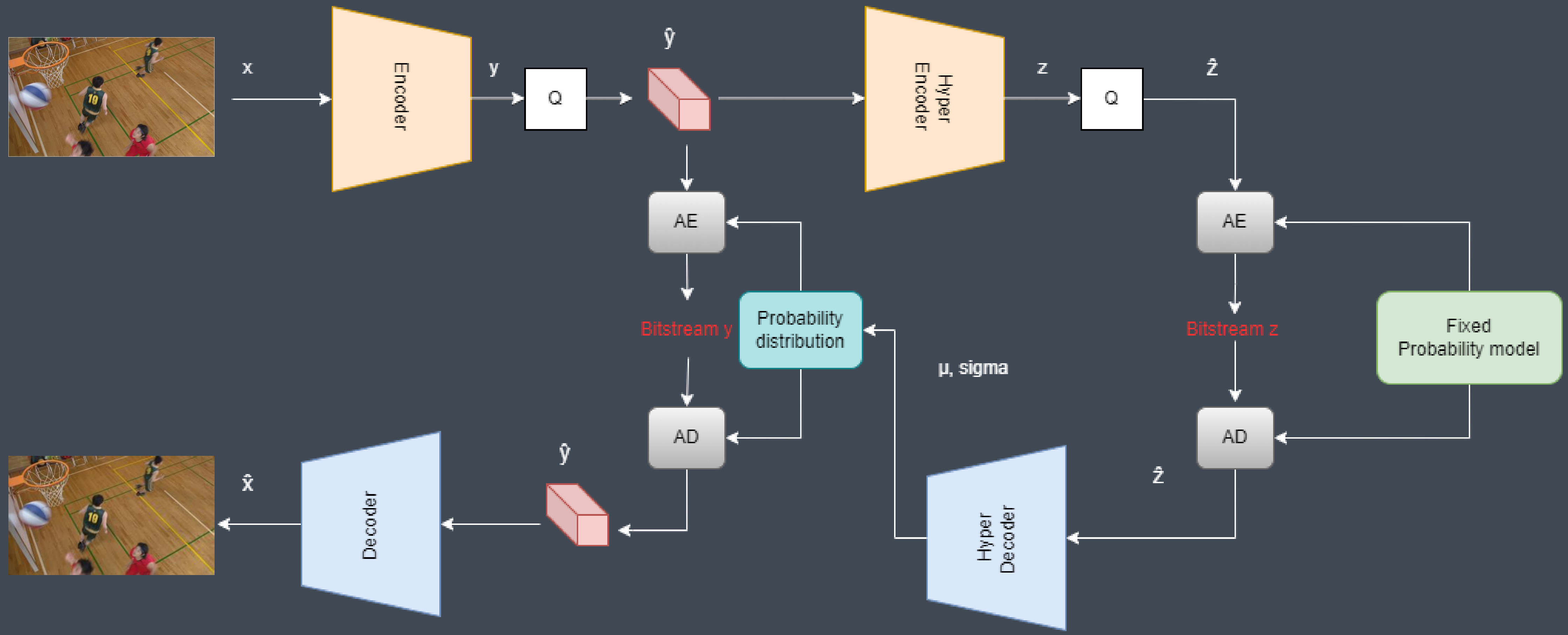
END-TO-END LEARNED IMAGE CODECS

- End-to-end learned image codecs are based on auto-encoder architecture
- Encoder transforms the input image x into a latent representation y
- Probability model estimates the probability distribution of the latent representation
- Decoder recontructs the input image from the latent representation



ATEME

END-TO-END LEARNED IMAGE CODECS



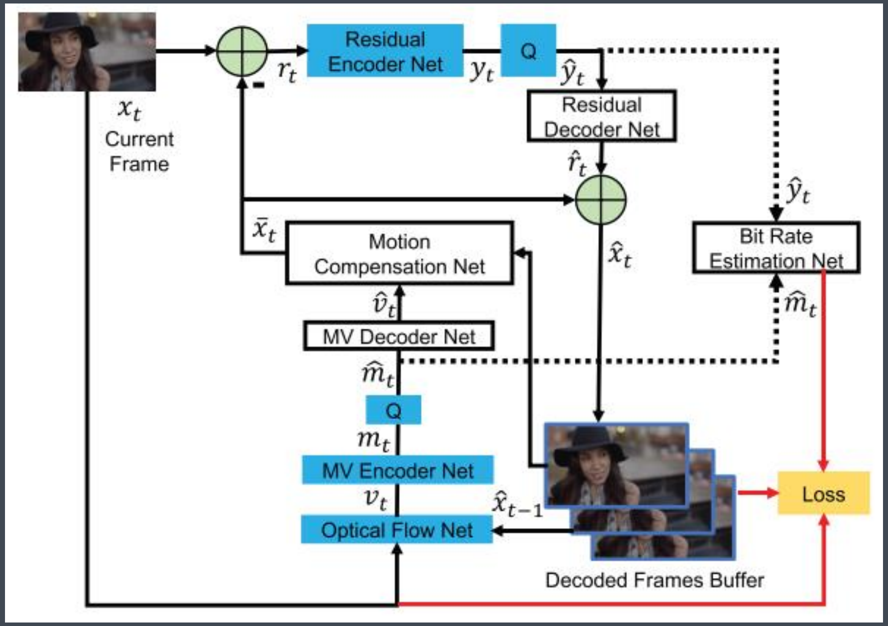
J. Balle, D. Minnen, S. Singh, and N.Johnston S.J Hwan,
"Variational image compression with a scale hyper-prior," ICLR 2018 - Conference Track Proceedings, 2018.

ATEME

END-TO-END LEARNED VIDEO CODECS

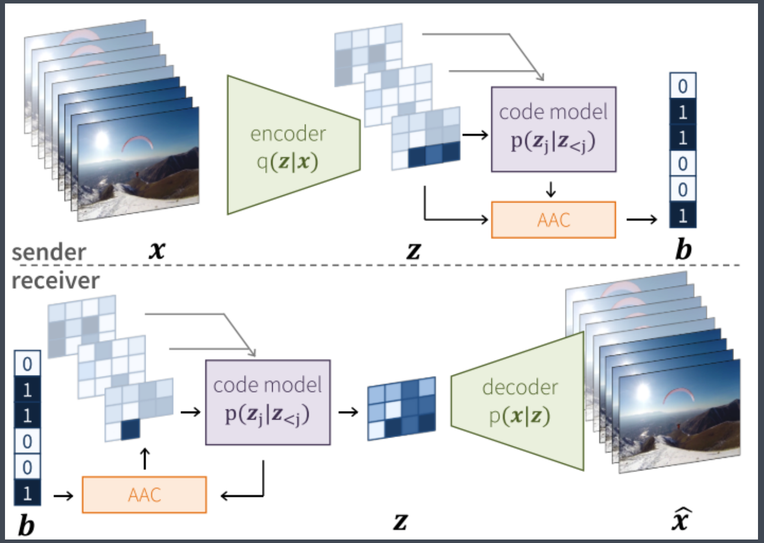
> Approaches of learned video compression :

Hybrid video coding



Lu, Guo and Ouyang, Wanli and Xu, Dong and Zhang, Xiaoyun and Cai, Chunlei and Gao, Zhiyong, "DVC: An End-To-End Deep Video Compression Framework", Proceedings of the IEEE/CVF Conference on CVPR, June, 2019

3D convolutions



Habibian, Amirhossein and Rozendaal, Ties van and Tomczak, Jakub M. and Cohen, Taco S., "Video Compression With Rate-Distortion Autoencoders", Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October, 2019

> Based on the autoencoder architecture

ATEME

END-TO-END LEARNED CODECS PRACTICAL APPLICATION

- Rate control
  - Some models need to be trained and stored for each bitrate
  - First attempts to have a more generic behavior **needs further study**

- Computing power
  - In short, deep learning is very demanding
  - To be efficient, models tend to be bigger and bigger

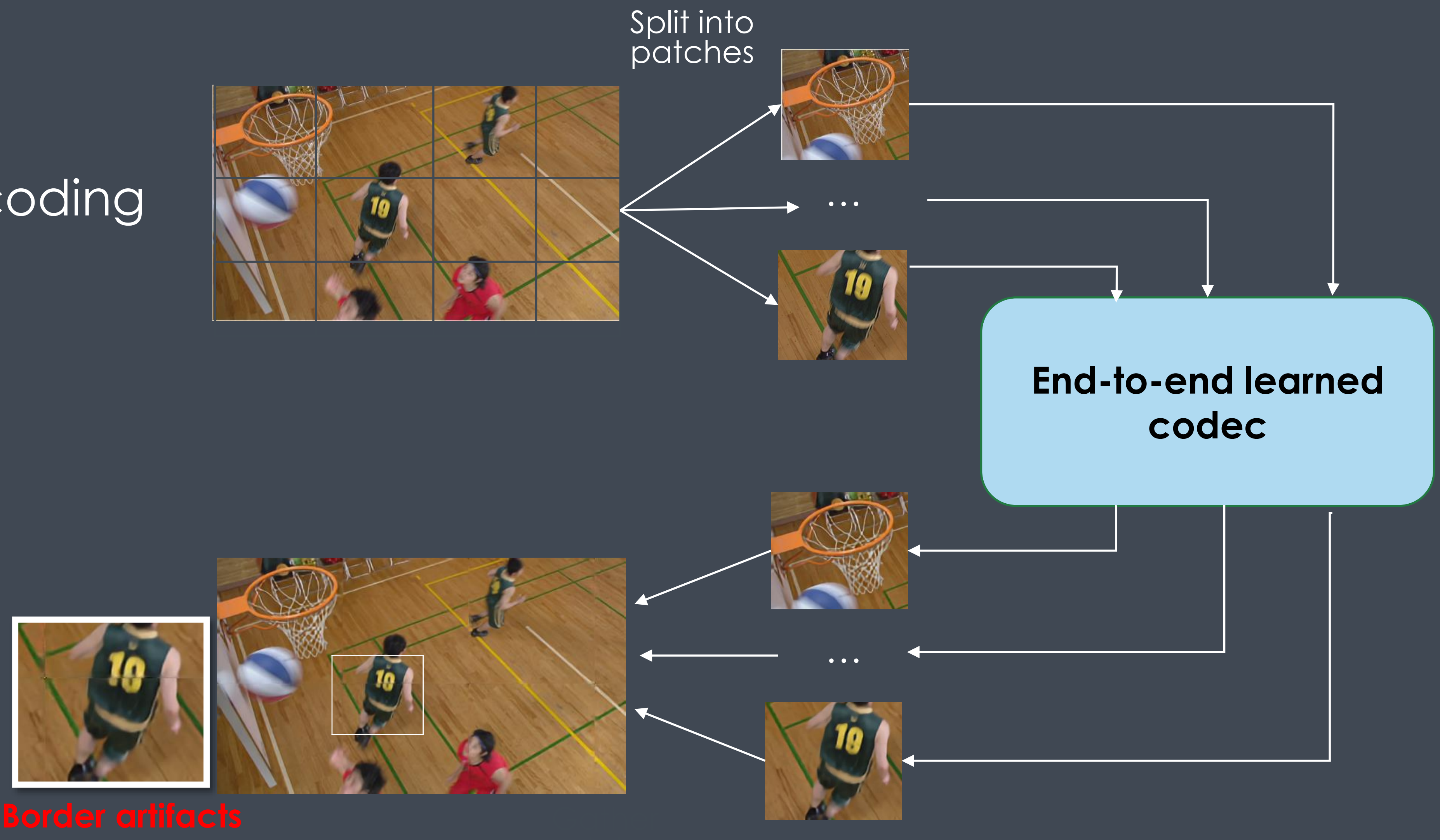
- Memory management
  - Huge model + huge frames = memory saturation

Learned patch-coding using overlapping

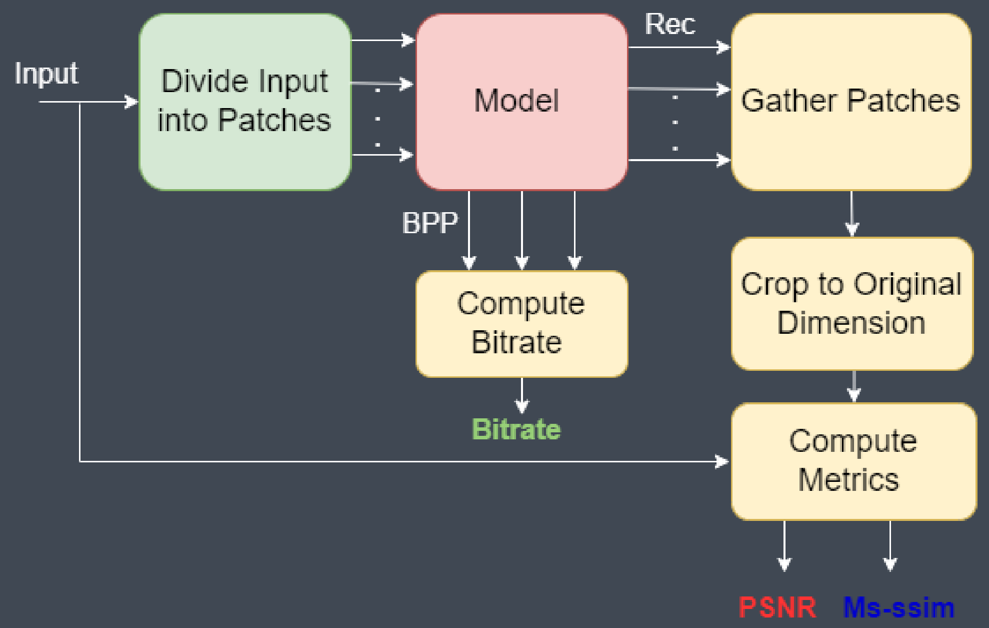
LEARNED PATCH-CODING USING OVERLAPPING

Limitation : Memory saturation while encoding big resolutions

Proposed solution : Patch-based encoding while removing artifacts with overlapping method



LEARNED PATCH-CODING USING OVERLAPPING



Steps to patch-based intra encoding coding

- Divide Input into **overlapping** patches horizontally and vertically
- End-to-end learned coding model used to encode patches **in parallel** (e.g : model based on cheng 2020[1] architecture)
- Gather patches using linear function to combine the recontructed overlapping area.

M. Tarchouli, M. Riviere, T. Guionnet, W. Hamidouche, M. Outtas, and O. Deforges. " PATCH-BASED IMAGE LEARNED CODEC USING OVERLAPPING ". Proceedings of Signal and Image Processing : An International Journal (SIPIJ), 2023

1- Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 7936–7945, 2020.

CODING TIME RESULTS

- Patch size : 256x256
- N : Number of overlapping pixels

Resolution	Method	Coding time GPU 2080ti 11GB	Coding time GPU 3090 24GB	Total Number of patches	Number of patches coded in parallel	N
1920x1080 (HD)	Full resolution coding	OOM	OOM	-	-	-
	Patch-coding, in parallel with overlapping	3.39s	2.03s	40	8	16
1280x720	Full resolution Coding	OOM	0.93s	-	-	-
	Patch-coding, in parallel with overlapping	1.91s	1.01s	15	5	16
832x480	Full resolution coding	1.06s	0.52s	-	-	-
	Patch-coding, in parallel with overlapping	1.10s	0.55s	8	8	16

Coding time Results

ATEME

RATE-DISTORTION RESULTS

Learned patch-coding using overlapping

Coding additional overlapping pixels

Potential Loss

Coding smaller patches

Getting closer to training conditions

Potential Gain

RATE-DISTORTION RESULTS

Visual results for MSE models

BD-rate (PSNR) results of patch-based coding schemes compared to full resolution coding system for CTC sequences.

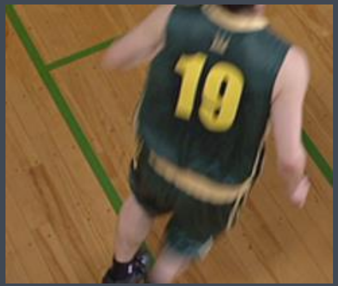
	Patch-coding w/o Overlapping	Patch-coding with Overlapping				
		N = 2	N = 4	N = 8	N = 16	N = 32
JVET CTC	+0.013	-0.025	-0.030	-0.034	-0.041	-0.035

BD-rate (MS-SSIM) results of patch-based coding schemes compared to full resolution coding system for CTC sequences.

	Patch-coding w/o Overlapping	Patch-coding with Overlapping				
		N = 2	N = 4	N = 8	N = 16	N = 32
JVET CTC	+0.611	+0.036	+0.005	-0.024	-0.040	-0.039

RATE-DISTORTION RESULTS

(a) original



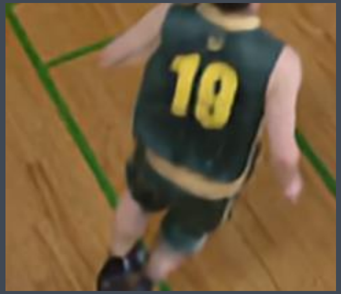
(b) full Image

(c) w/o overlap

(d) N = 2

(e) N = 4

(f) N = 8



<u>**Visual results for MS-SSIM models**</u>

BD-rate (PSNR) results of patch-based coding schemes compared to full resolution coding system for CTC sequences.

	Patch-coding w/o Overlapping	Patch-coding with Overlapping				
		N = 2	N = 4	N = 8	N = 16	N = 32
JVET CTC	+0.013	-0.025	-0.030	-0.034	-0.041	-0.035

BD-rate (MS-SSIM) results of patch-based coding schemes compared to full resolution coding system for CTC sequences.

	Patch-coding w/o Overlapping	Patch-coding with Overlapping				
		N = 2	N = 4	N = 8	N = 16	N = 32
JVET CTC	+0.611	+0.036	+0.005	-0.024	-0.040	-0.039

3

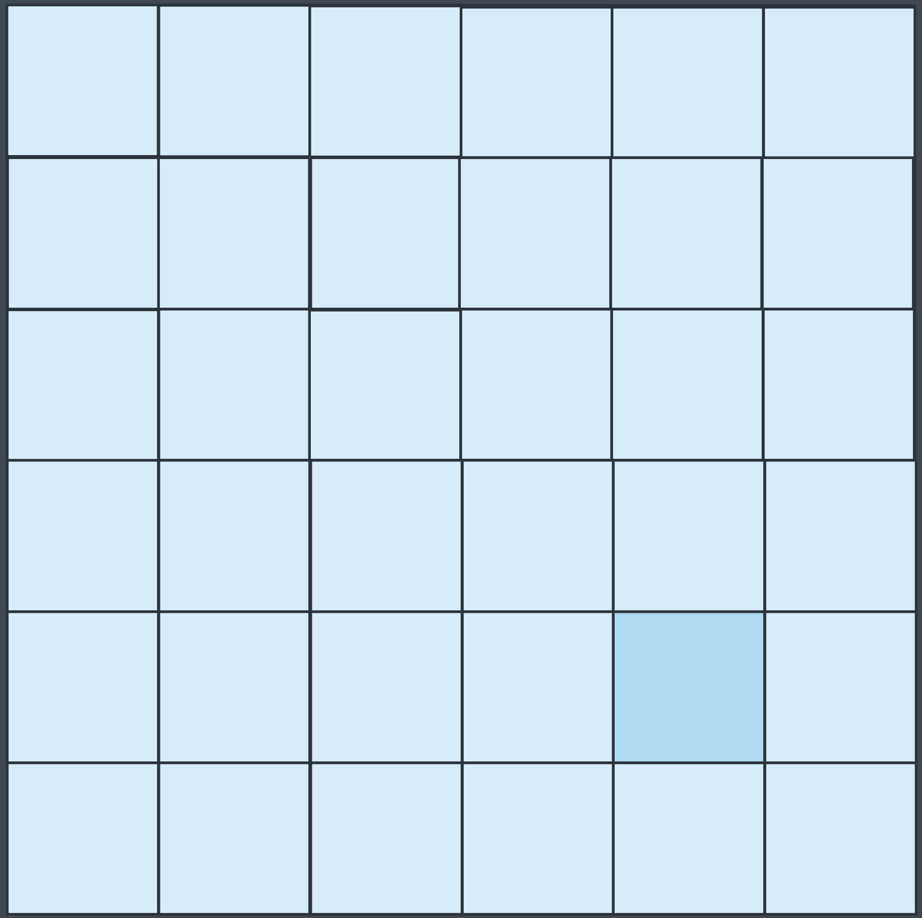
Adjustment of MS-SSIM implementation

Confidential & proprietary

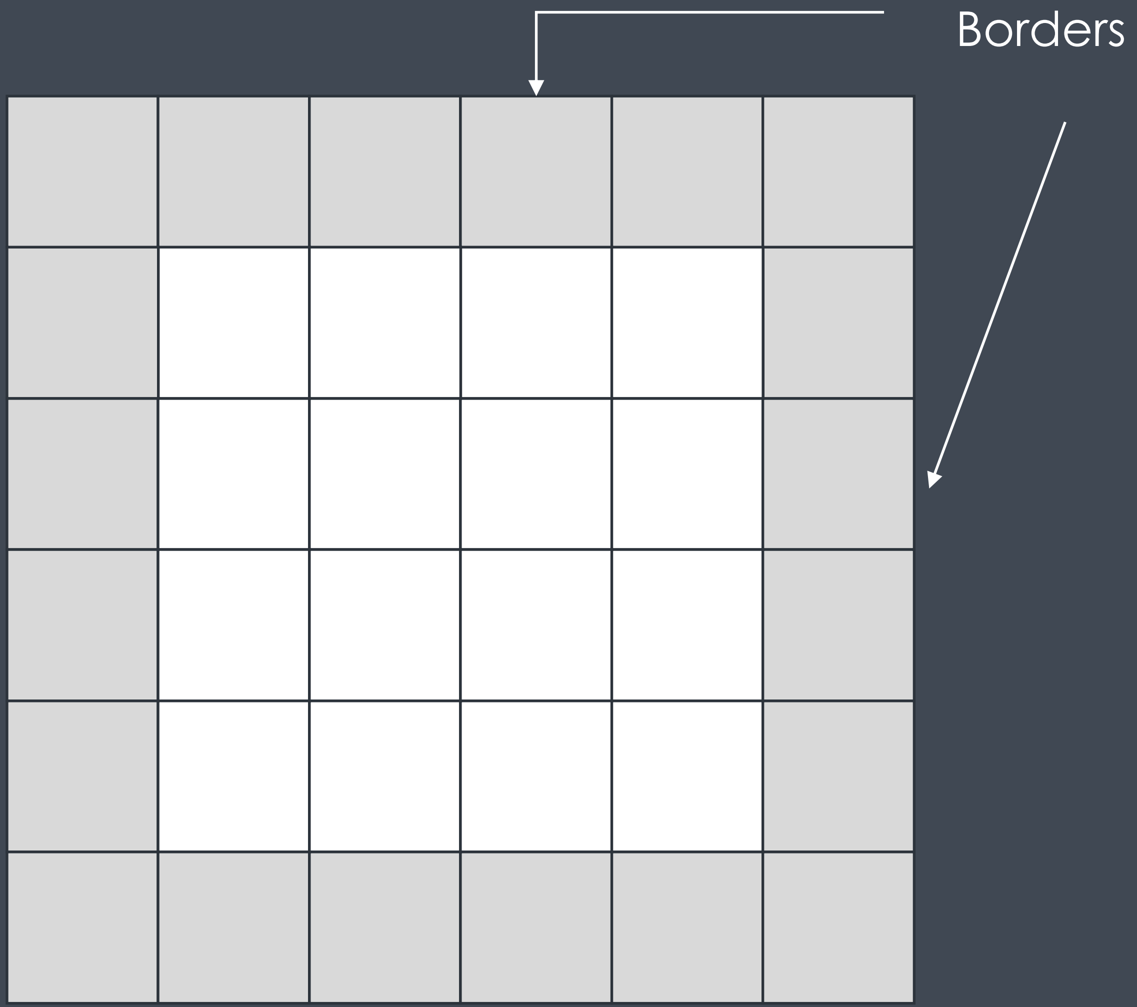
ADJUSTMENT OF MS-SSIM IMPLEMENTATION

- MS-SSIM is an objective quality metric designed to measure the similarity between two images/videos.
- This metric is commonly used in the compression video field, to assess the quality between the original input video sequence and the decoded one.
- The MS-SSIM function consists in computing the SSIM metric on multiple scales using a sliding Gaussian window

ADJUSTMENT OF MS-SSIM IMPLEMENTATION

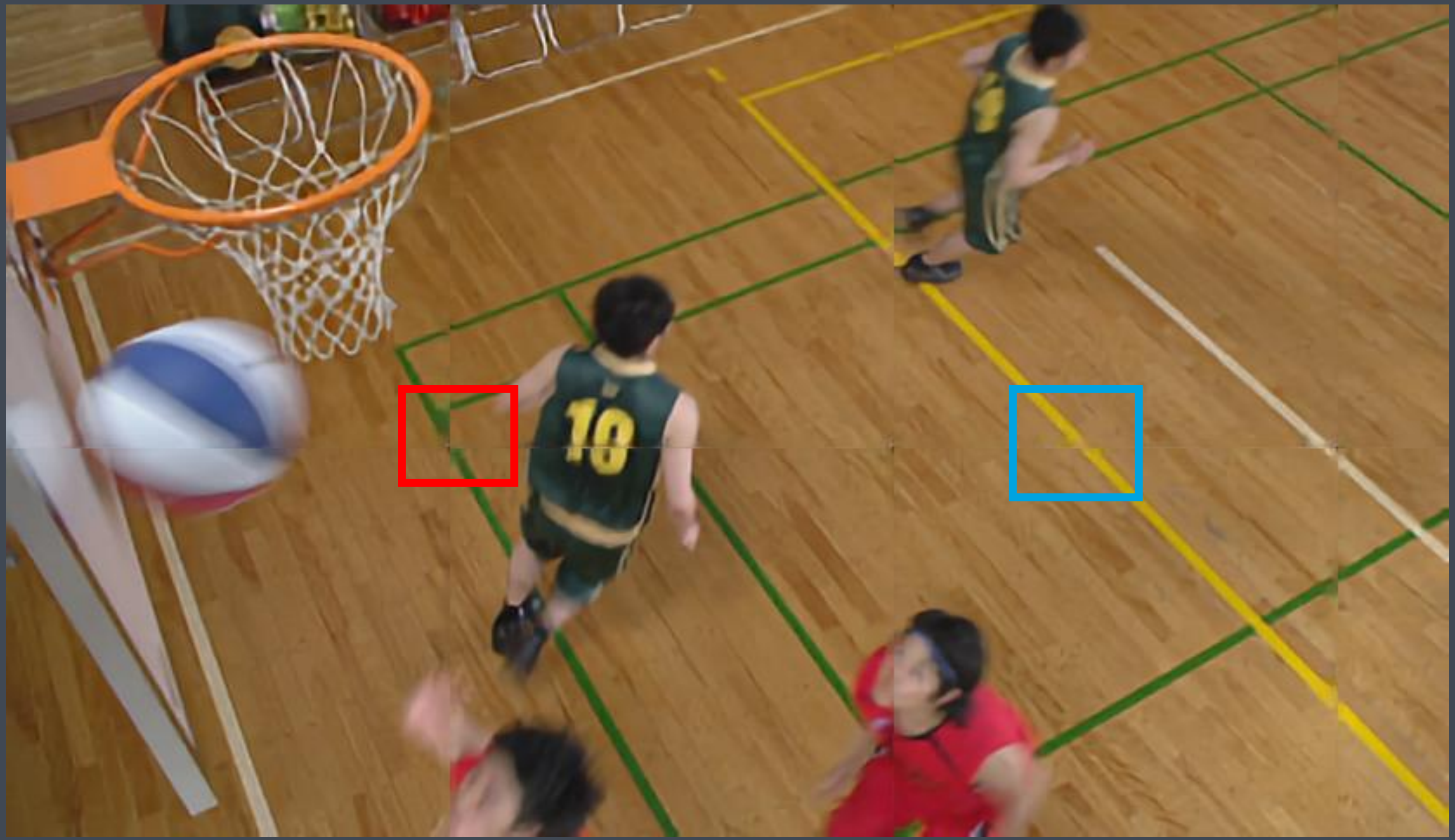


Classic Implementation of MS-SSIM



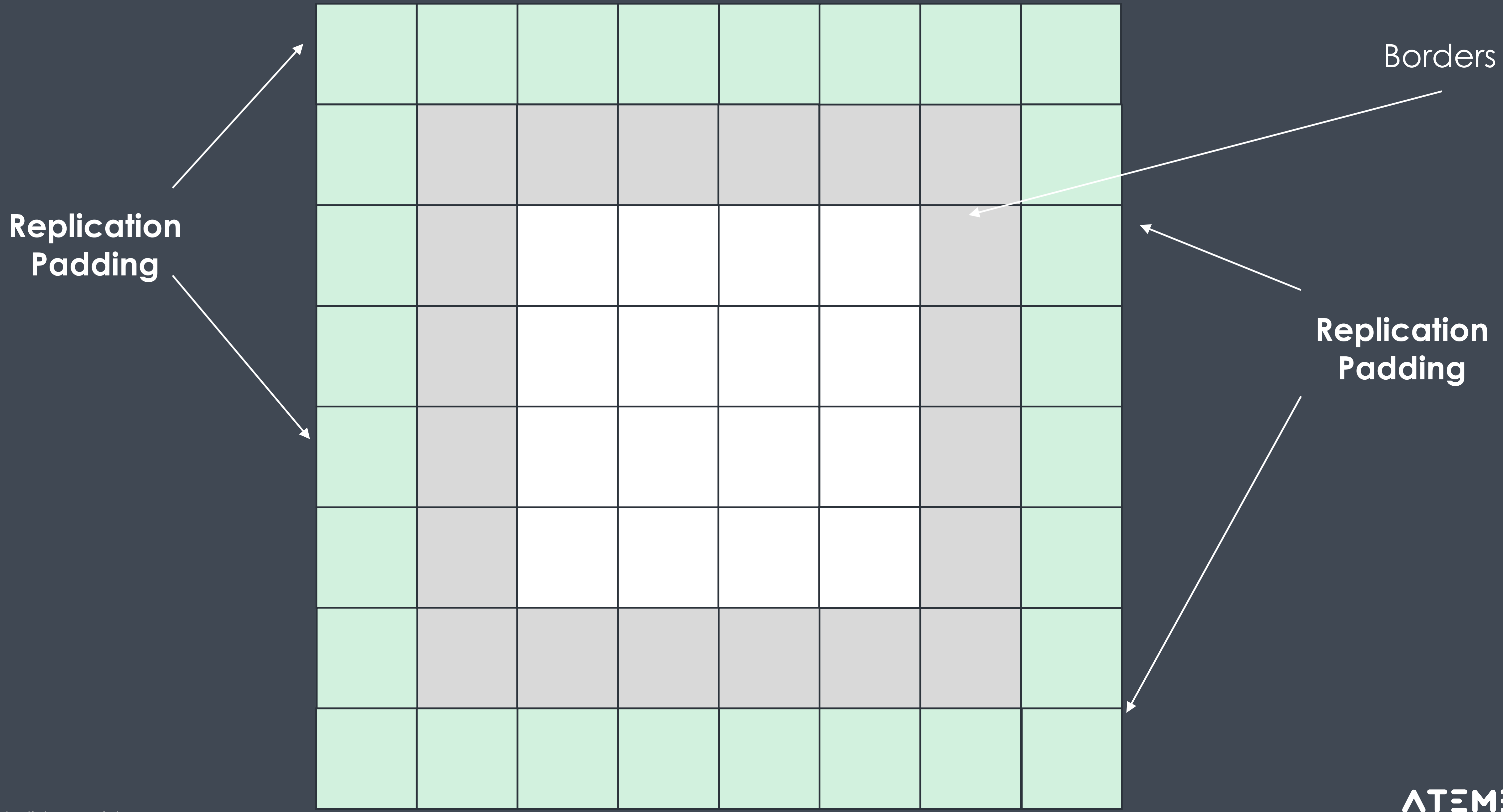
Borders not considered in the classic Implementation of MS-SSIM

ADJUSTMENT OF MS-SSIM IMPLEMENTATION



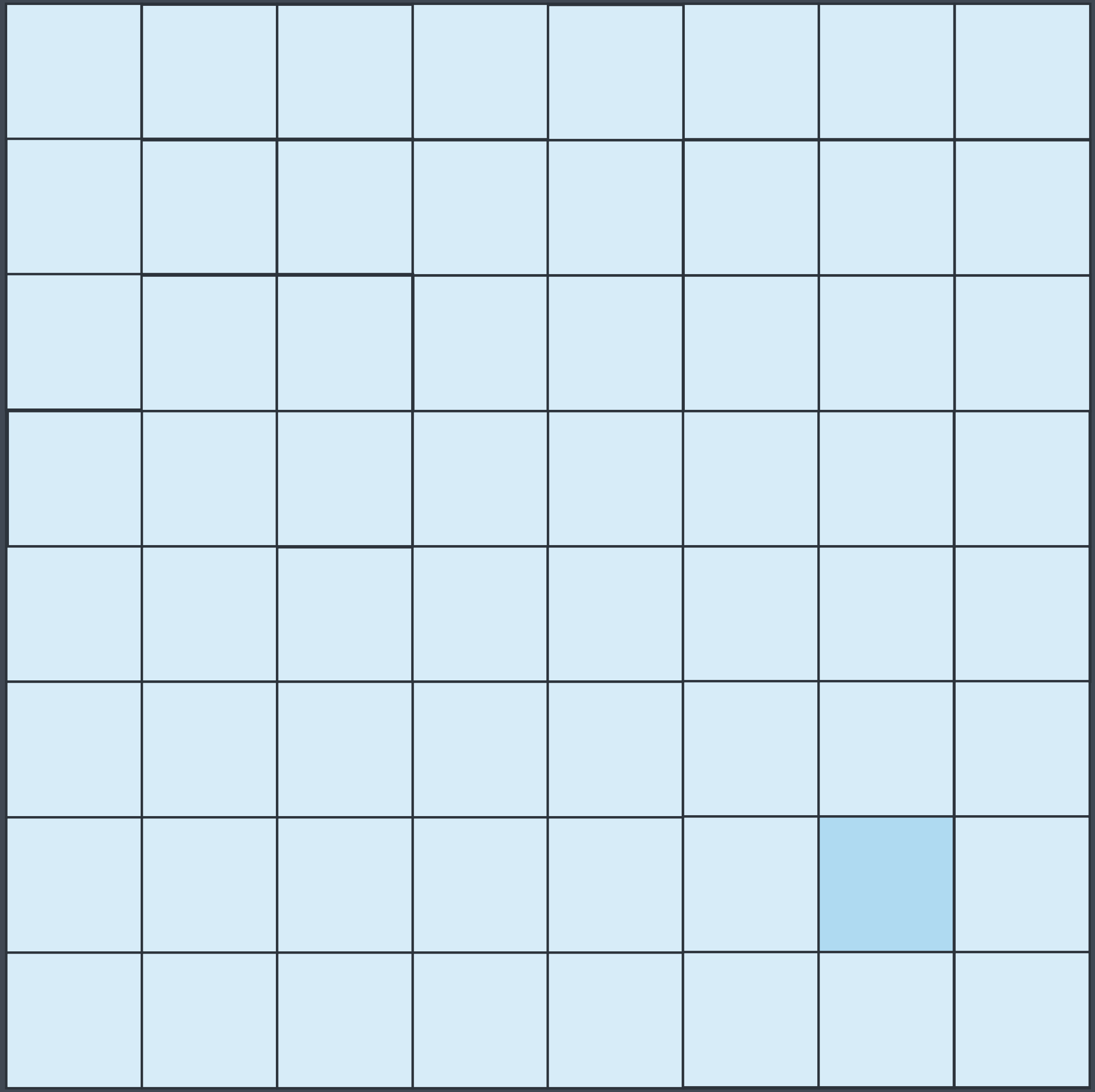
Border artifacts resulting from patch-coding using the classic implementation of MS-SSIM

ADJUSTMENT OF MS-SSIM IMPLEMENTATION



ATEME

ADJUSTMENT OF MS-SSIM IMPLEMENTATION



ATEME

ADJUSTMENT OF MS-SSIM IMPLEMENTATION

Border artifacts resulting from patch-coding using the classic implementation of MS-SSIM

Border artifacts resulting from patch-coding using the adjusted implementation of MS-SSIM

BD-rate results of patch-coding w/o overlapping compared to full frame coding system for CTC sequences, using both implementations of MS-SSIM.

Sequence	Class Classic MS-SSIM	Adjusted MS-SSIM
Class B	0.649	0.017
Class C	0.537	0.017
Class D	0.232	0.004
Class E	0.958	0.042
Class F	0.686	0.042

ATEME

RESULTS OF THE LEARNED PATCH-CODING METHOD WITH THE ADJUSTED MS-SSIM

Visual results for MS-SSIM models

BD-rate (Classic MS-SSIM) gains of patch-based coding schemes compared to full resolution coding system for CTC sequences.

	Patch-coding w/o Overlapping	**Patch-coding with Overlapping**				
		N = 2	N = 4	N = 8	N = 16	N = 32
JVET CTC	+0.611	+0.036	+0.005	-0.024	-0.040	-0.039

BD-rate (Adjusted MS-SSIM) gains of patch-based coding schemes compared to full resolution coding system for CTC sequences.

	Patch-coding w/o Overlapping	**Patch-coding with Overlapping**				
		N = 2	N = 4	N = 8	N = 16	N = 32
JVET CTC	+0.024	-0.004	-0.007	-0.016	-0.020	-0.014

4

Parallel Neural video coding (PaNVC) Framework

Confidential & proprietary

PANVC FRAMEWORK

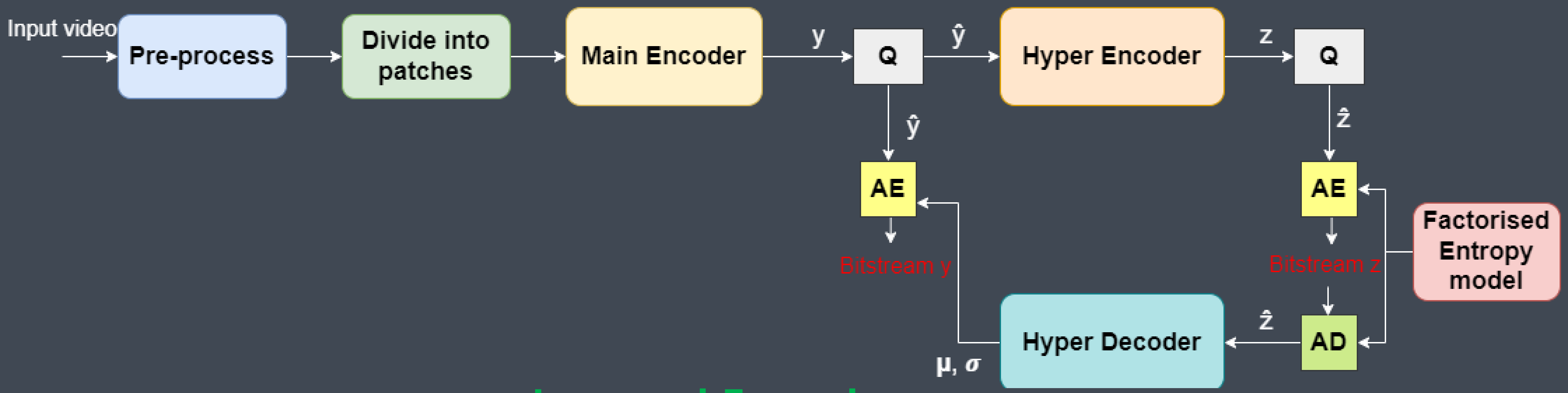
- Goal :
  - Build an end-to-end learned video coding framework, adapted to real-life applications.
  - This framework is designed to be compatible with several learned video coding models

- Contributions
  - Integrate the learned patch-coding method to manage the memory efficiently
  - Exploit parallelization
  - Train independent Encoder and Decoder
  - Generate an actual bitstream
  - Compute quality metrics (MS-SSIM and PSNR) as well as the rate of the video (theoretical bitrate and the real bitrate)

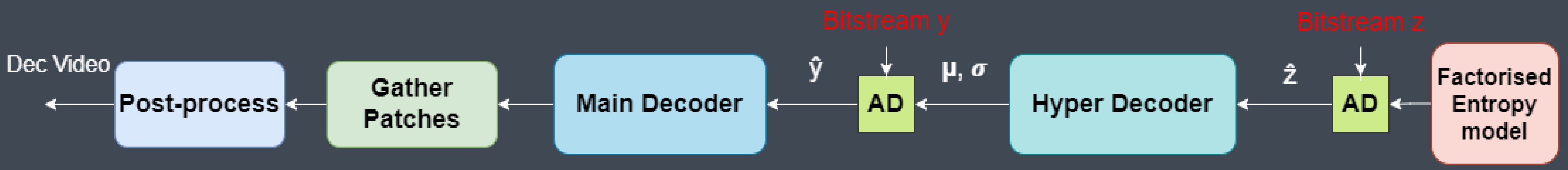
- Implementation
  - Language : Python , Framework : Pytorch

ATEME

INDEPENDENT ENCODER AND DECODER



Learned Encoder



Learned Decoder

BITSTREAM ARCHITECTURE

Wrap-up

CONCLUSION

- PaNVC is an end-to-end learned video coding framework adapted to practical application.

- PaNVC is designed to be compatible with several neural video coding models

- PaNVC includes the learned patch-coding method using overlapping, as a solution to memory saturation problem.

- The implementation of MS-SSIM function is adjusted to the patch-coding use-case.

ATEME

REFERENCES

- [1] J. Balle, D. Minnen, S. Singh, and N.Johnston S.J Hwan, "Variational image compression with a scale hyper-prior," ICLR 2018 - Conference Track Proceedings, 2018.

- [2] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Learned image compression with discretized gaussian mixture likelihoods and attention modules," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 7936–7945, 2020.

- [3] M. Tarchouli, M. Riviere, T. Guionnet, W. Hamidouche, M. Outtas, and O. Deforges. " PATCH-BASED IMAGE LEARNED CODEC USING OVERLAPPING ". Proceedings of Signal and Image Processing : An International Journal (SIPIJ), 2023

- [4] https://github.com/jorge-pessoa/pytorch-msssim

THANK YOU FOR YOUR ATTENTION