Conditional Coding for Flexible Learned Video Compression

Théo Ladune[1,2]

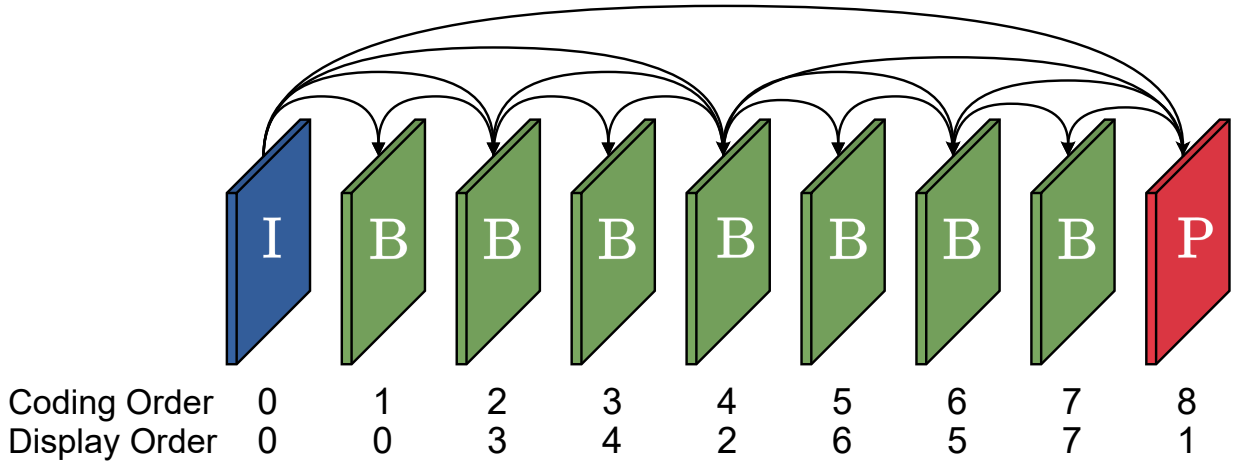
Pierrick Philippe[1], Olivier Déforges[2], Wassim Hamidouche[2], Lu Zhang[2]

[1]Orange — [2]INSA Rennes
theo.ladune@orange.com

ICLR 2021 – Neural Compression Workshop, 7 May 2021

Context & Contributions

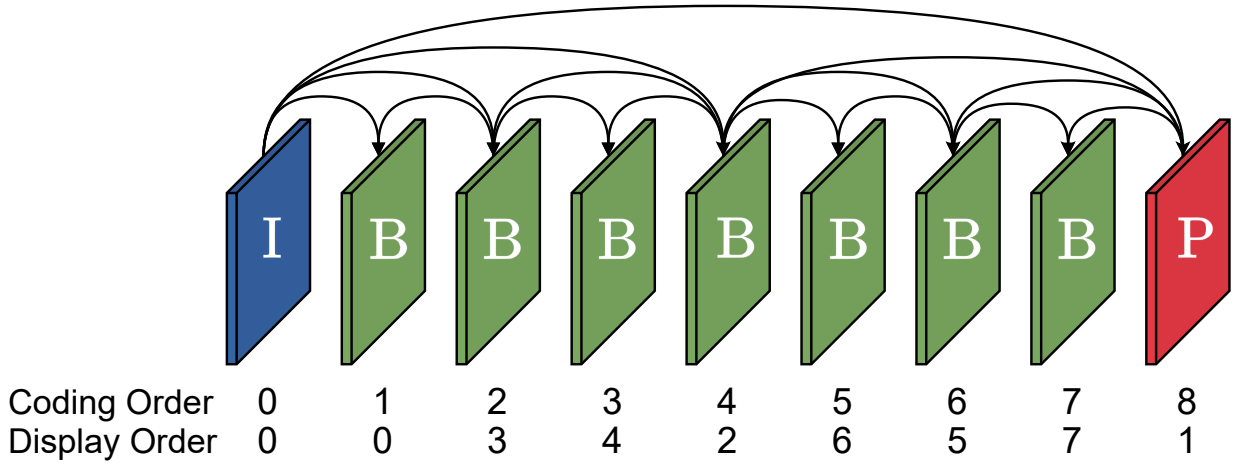
- **Learned** Video Coding



- 2 types of frame $\begin{cases} \text{Intra} & : \text{0 reference (I)} \\ \text{Inter} & : \text{1 ref. (P) or 2 ref. (B)} \end{cases}$

Context & Contributions

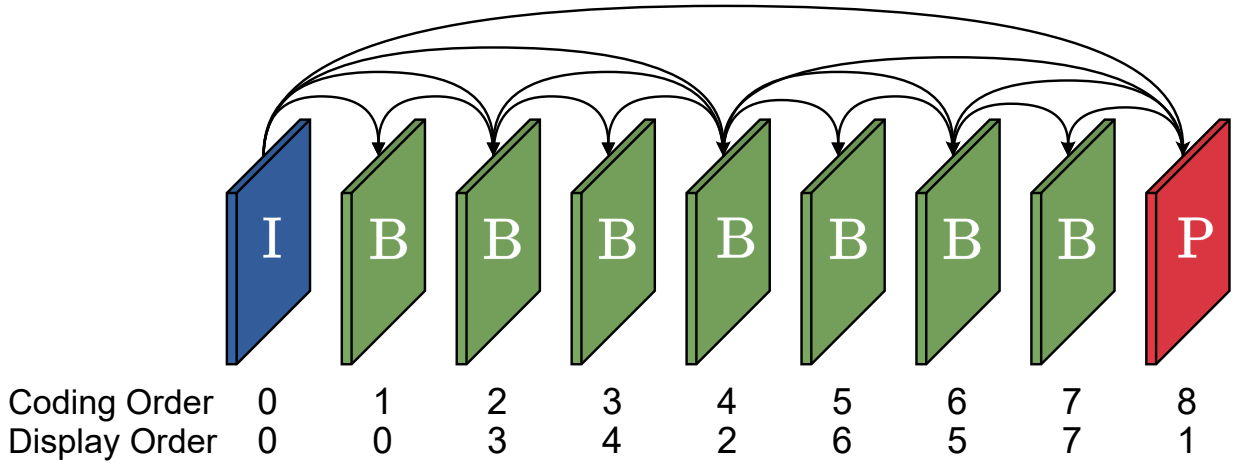
- **Learned** Video Coding



- 2 types of frame $\begin{cases} \text{Intra} & : 0 \text{ reference (I)} \\ \text{Inter} & : 1 \text{ ref. (P) or } 2 \text{ ref. (B)} \end{cases}$
- Previous work: **2 separate coders** for intra and inter-frames

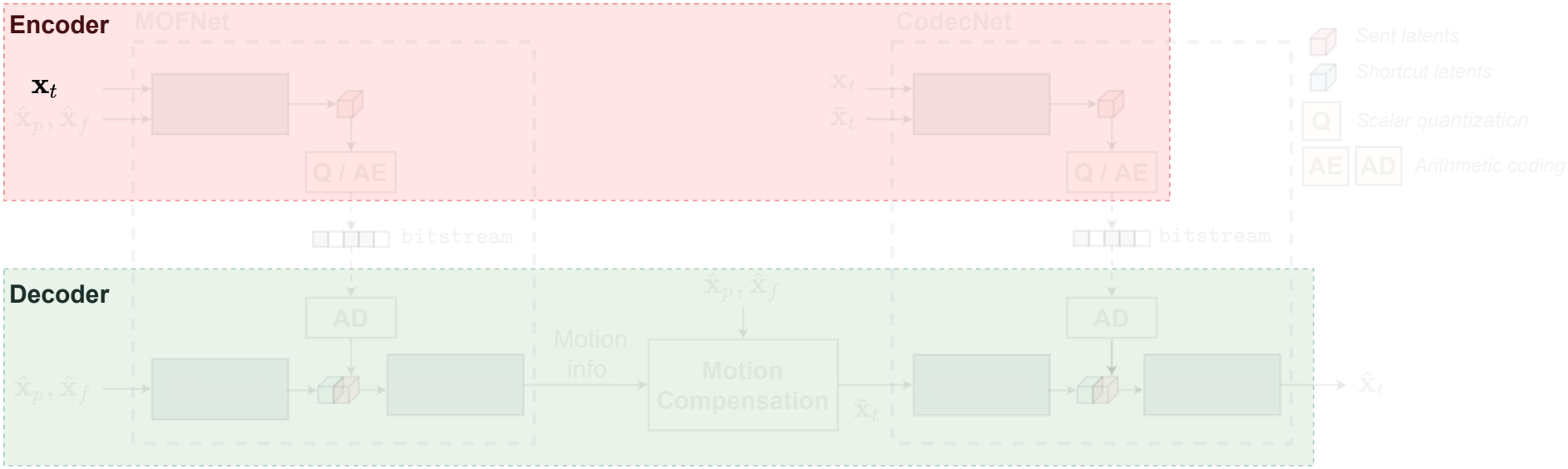
Context & Contributions

- **Learned** Video Coding



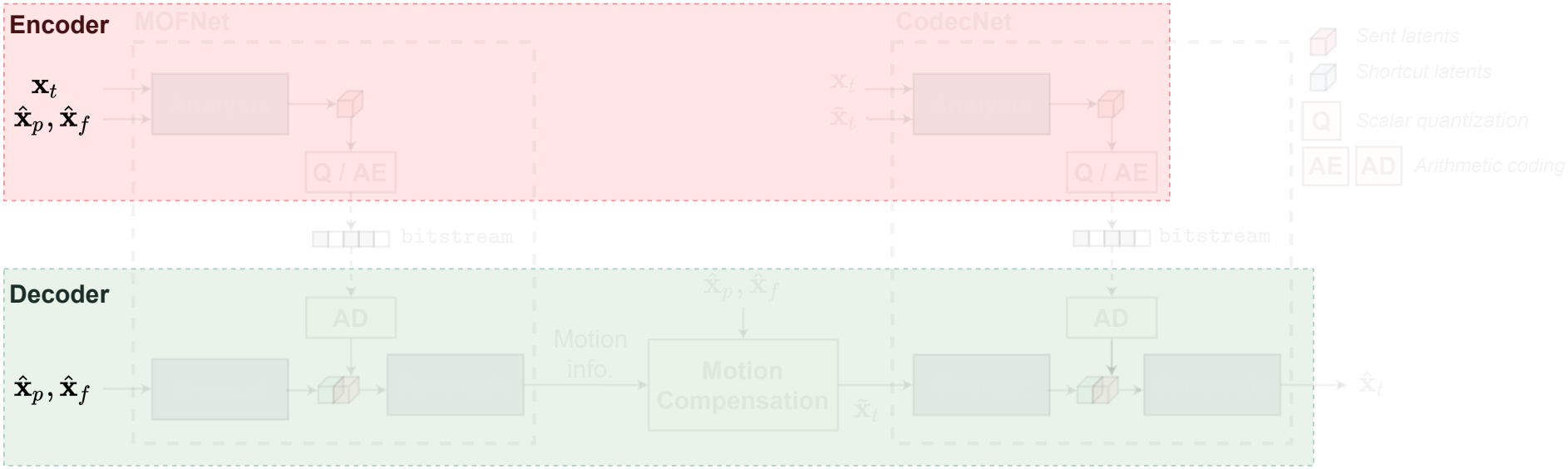
- 2 types of frame $\begin{cases} \text{Intra} & : 0 \text{ reference (I)} \\ \text{Inter} & : 1 \text{ ref. (P) or } 2 \text{ ref. (B)} \end{cases}$
- Previous work: **2 separate coders** for intra and inter-frames
- Contribution: **conditional coding**
  - Leverage 0, 1 or 2 ref. with a **unique** coder
  - **Flexible** coder on par with HEVC

Coding Scheme Overview



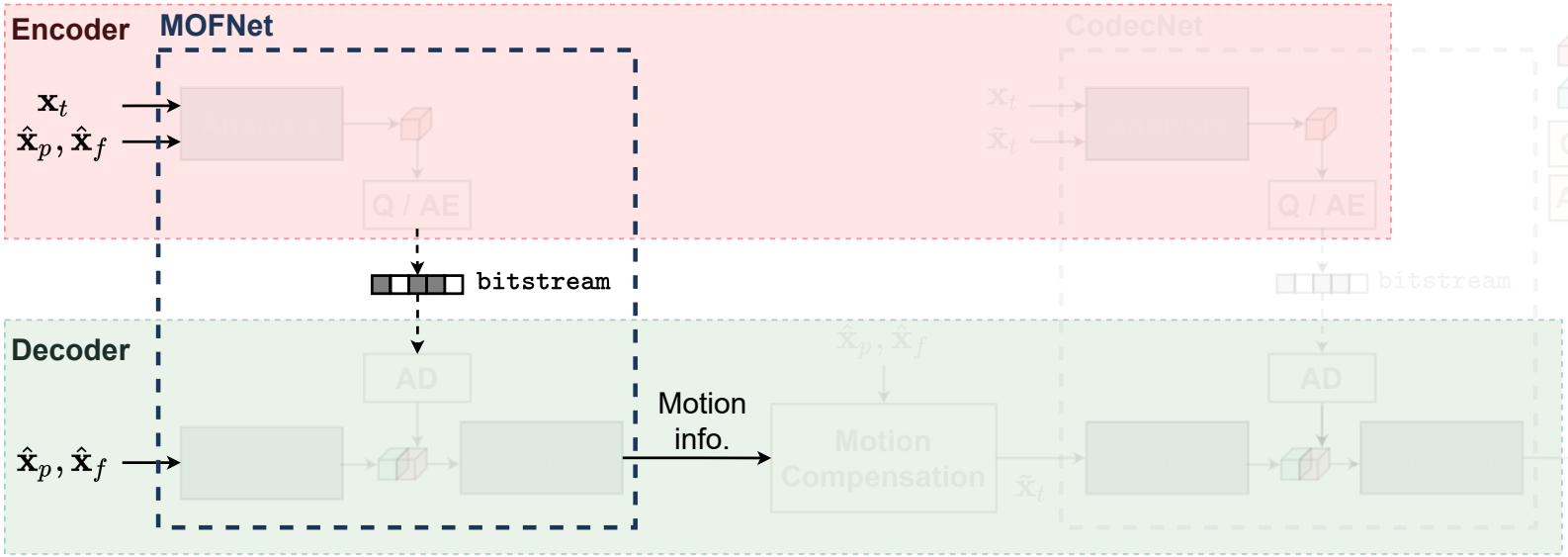
- Coding $\mathbf{x}_t$

Coding Scheme Overview



- Coding $\mathbf{x}_t$ knowing **up to 2** references $\hat{\mathbf{x}}_p$, $\hat{\mathbf{x}}_f$

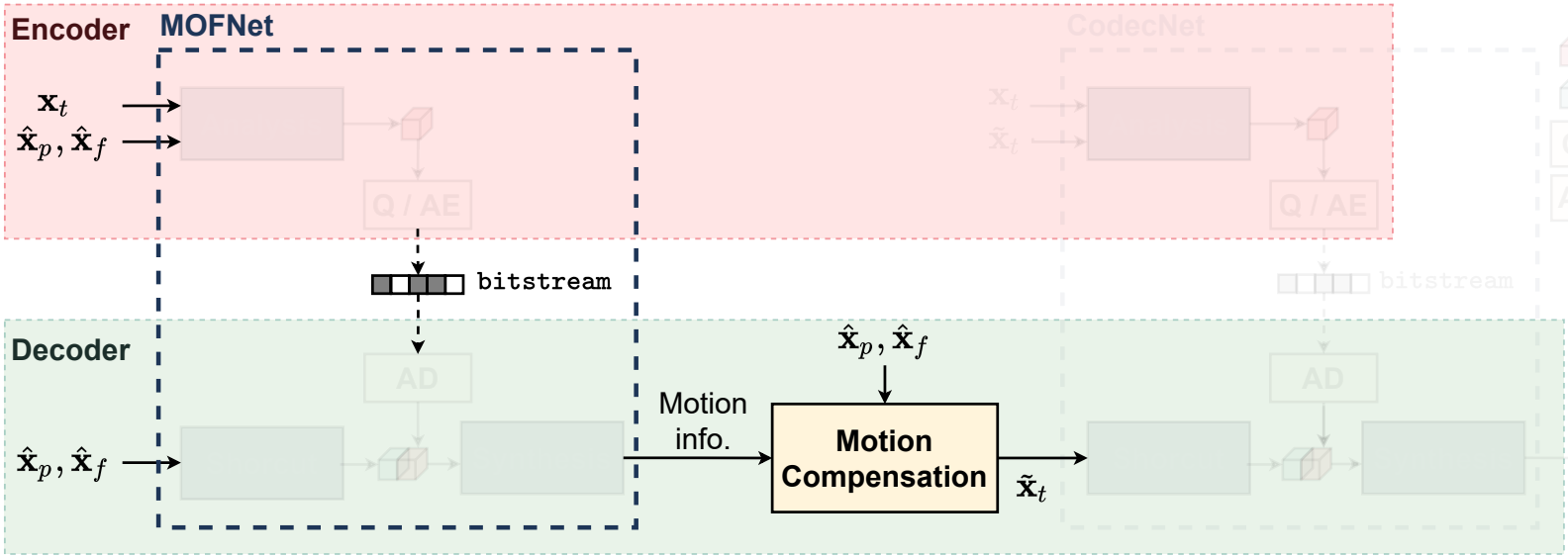
Coding Scheme Overview



- Coding $\mathbf{x}_t$ knowing **up to 2** references $\hat{\mathbf{x}}_p$, $\hat{\mathbf{x}}_f$
- MOFNet transmits **motion info**

Optical flow for motion compensation

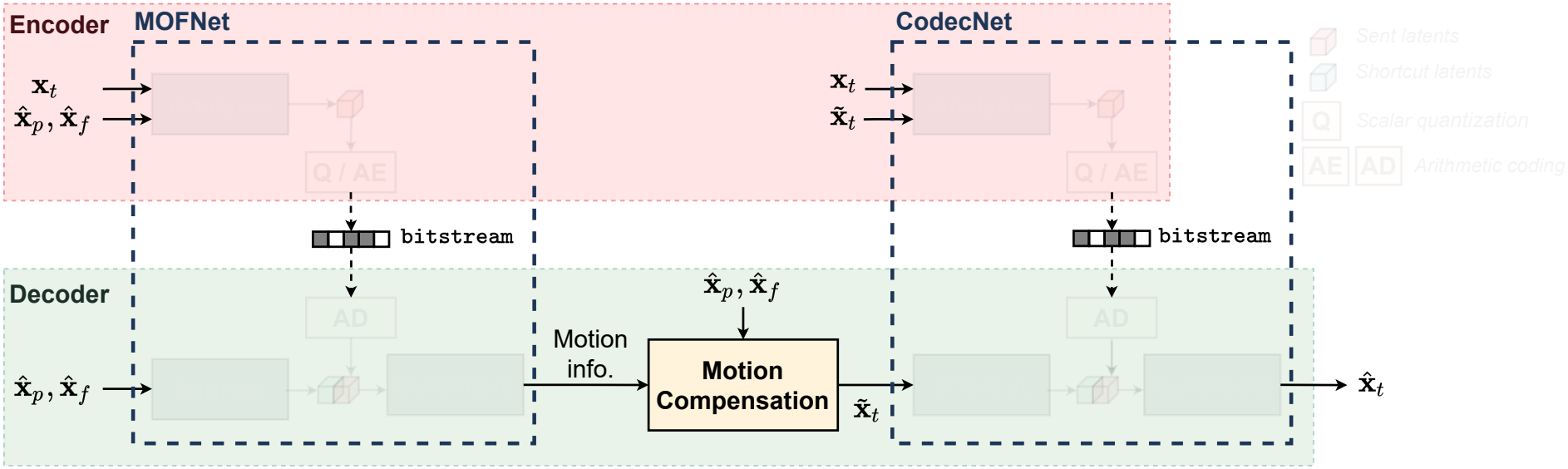
Coding Scheme Overview



- Coding $\mathbf{x}_t$ knowing **up to 2** references $\hat{\mathbf{x}}_p$, $\hat{\mathbf{x}}_f$
- MOFNet transmits **motion info** to compute the prediction $\tilde{\mathbf{x}}_t$

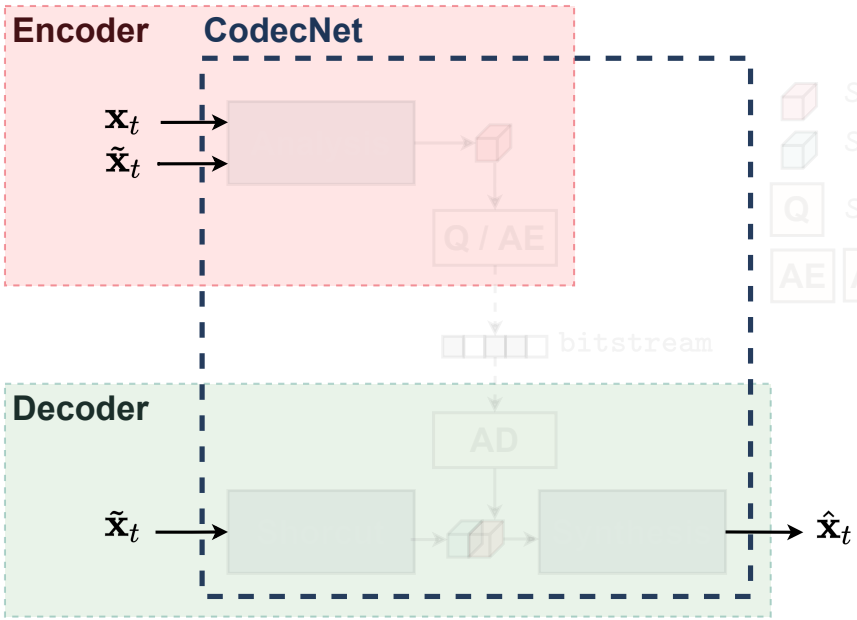
Optical flow for motion compensation

Coding Scheme Overview

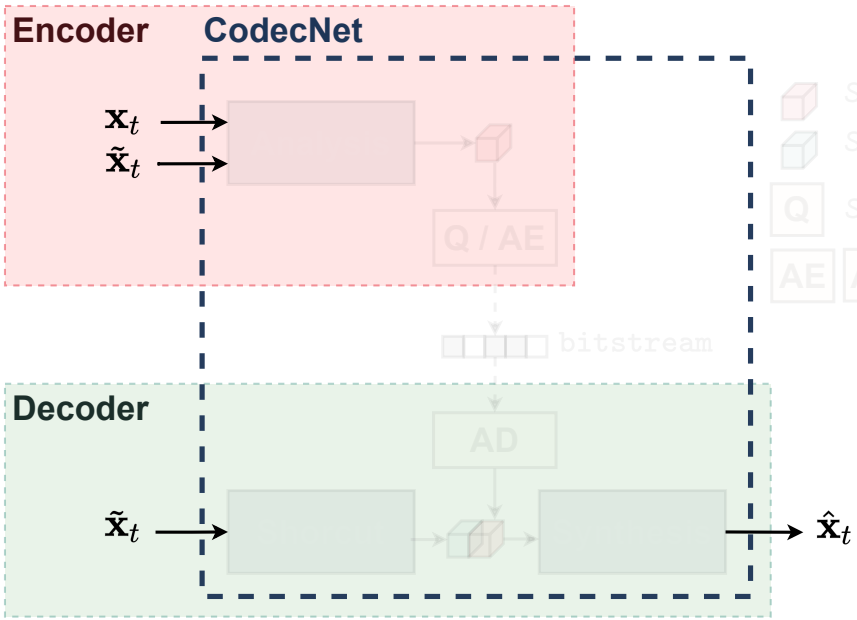


- Coding $\mathbf{x}_t$ knowing **up to 2** references $\hat{\mathbf{x}}_p$, $\hat{\mathbf{x}}_f$
- MOFNet transmits **motion info**
- CodecNet sends the **unpredictable** part to get the decoded frame $\hat{\mathbf{x}}_t$

Conditional Coding

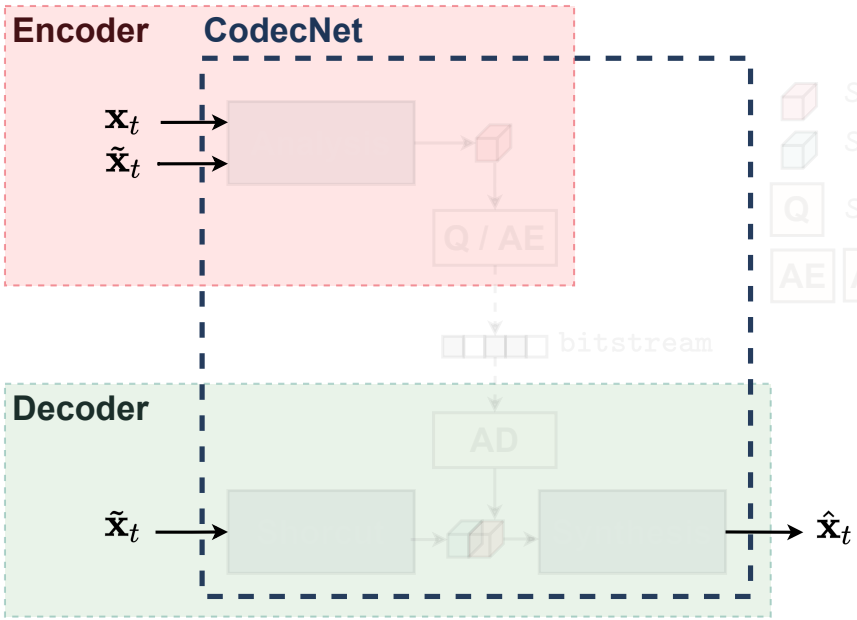


Conditional Coding



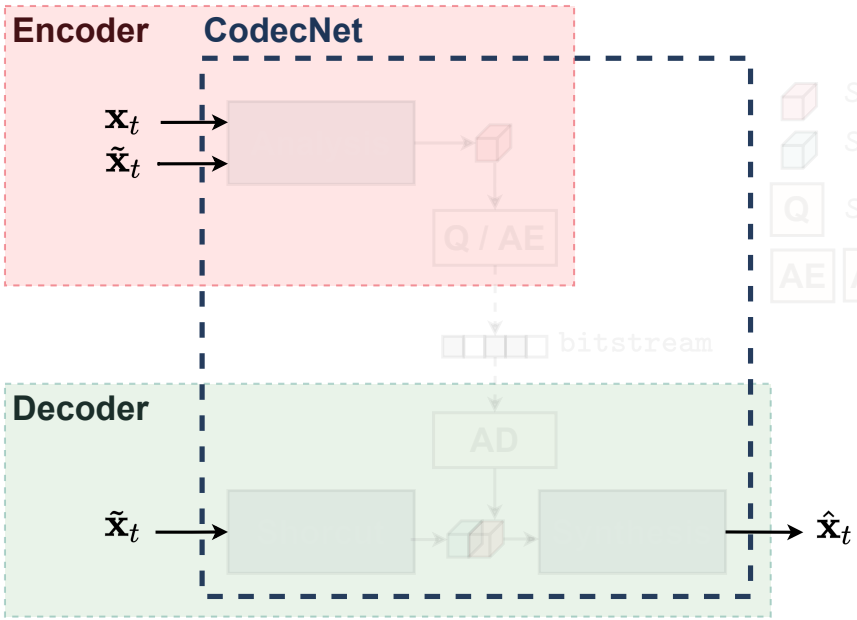
- Transmit $\mathbf{x}_t$ knowing $\tilde{\mathbf{x}}_t$
- Previous work: **residual** coding
  - **2 types** of signal: image (intra) & residue (inter) $\rightarrow$ **2 coders**

Conditional Coding



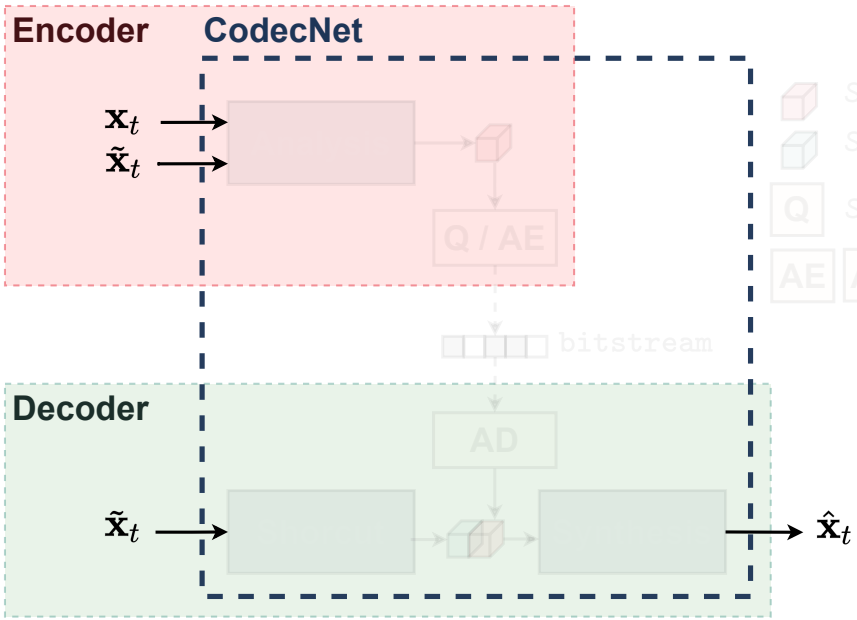
- Transmit $\mathbf{x}_t$ knowing $\tilde{\mathbf{x}}_t$
- Previous work: **residual** coding
  - **2 types** of signal: image (intra) & residue (inter) $\rightarrow$ **2 coders**
- Conditional coding: **generalization** of residual coding

Conditional Coding



- Transmit $\mathbf{x}_t$ knowing $\tilde{\mathbf{x}}_t$
- Previous work: **residual** coding
  - **2 types** of signal: image (intra) & residue (inter) $\rightarrow$ **2 coders**
- Conditional coding: **generalization** of residual coding
  - Always process image-domain inputs $\rightarrow$ **single coder**

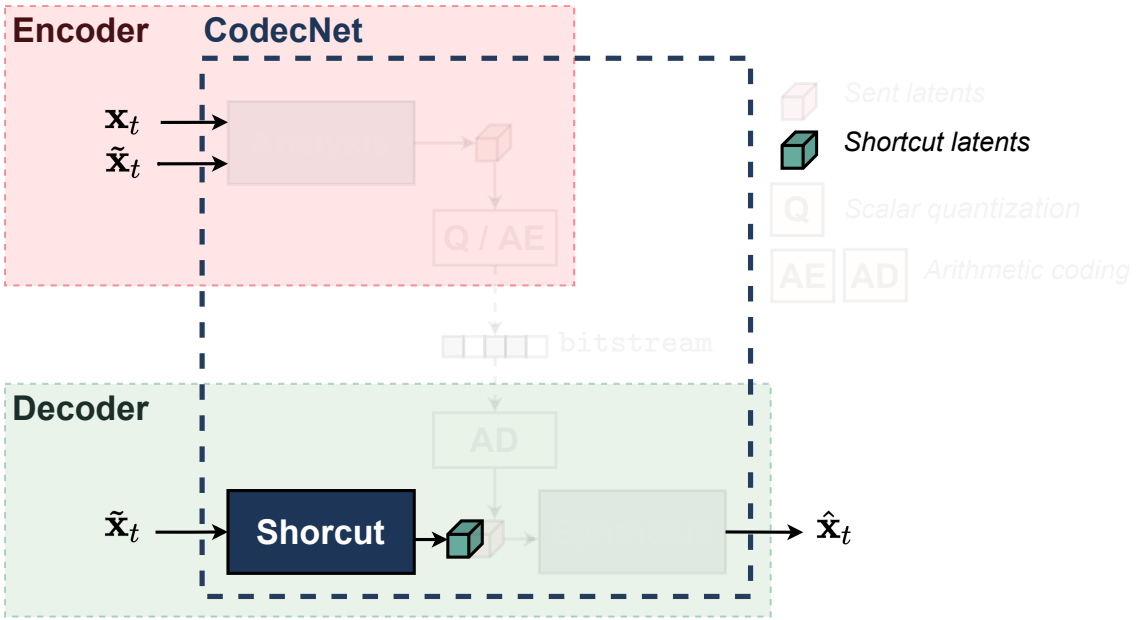
Conditional Coding



- Transmit $\mathbf{x}_t$ knowing $\tilde{\mathbf{x}}_t$
- Previous work: **residual** coding
  - **2 types** of signal: image (intra) & residue (inter) $\rightarrow$ **2 coders**
- Conditional coding: **generalization** of residual coding
  - Always process image-domain inputs $\rightarrow$ **single coder**
  - Offer better performance[1]

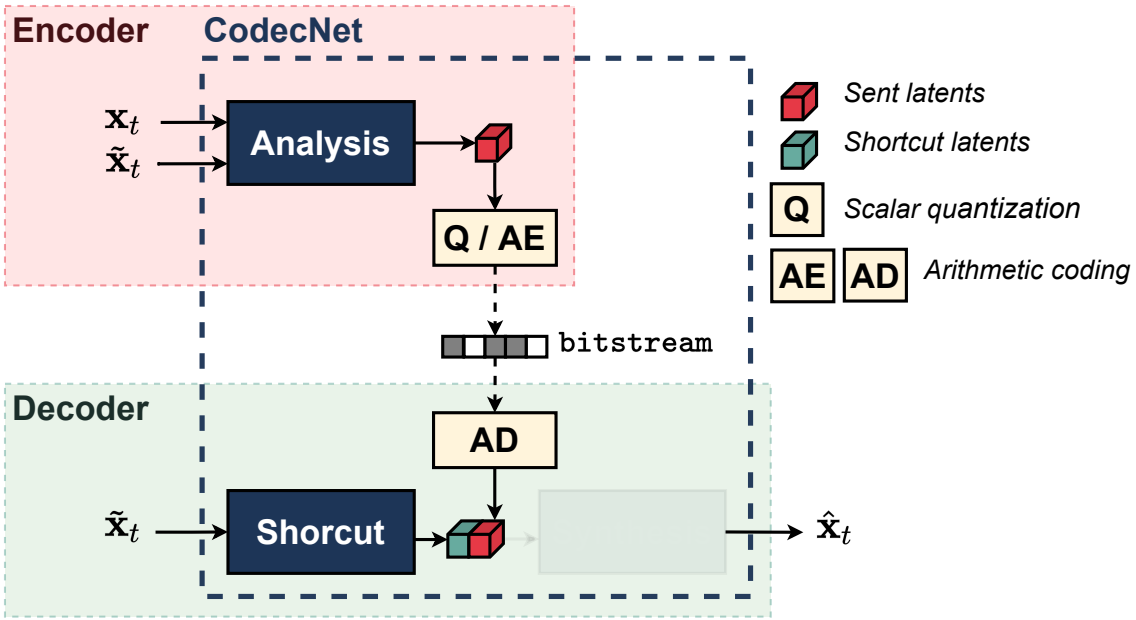
[1] Ladune, *Optical Flow and Mode Selection for Learning-based Video Coding*, MMSP 2020

Conditional Coding



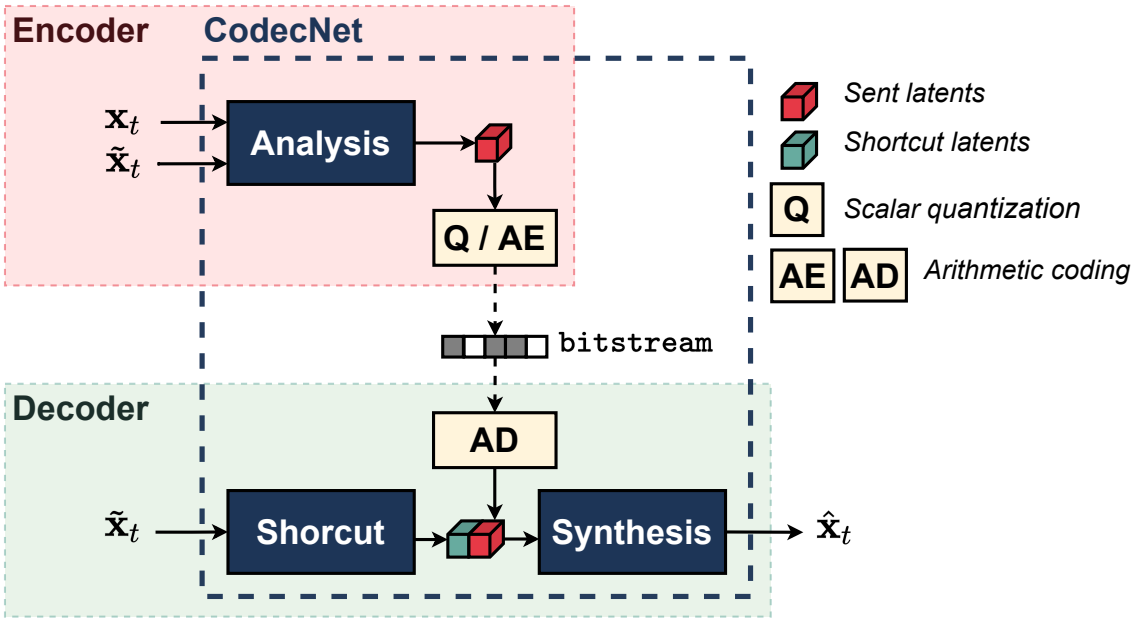
- Shortcut: exploit decoder information

Conditional Coding



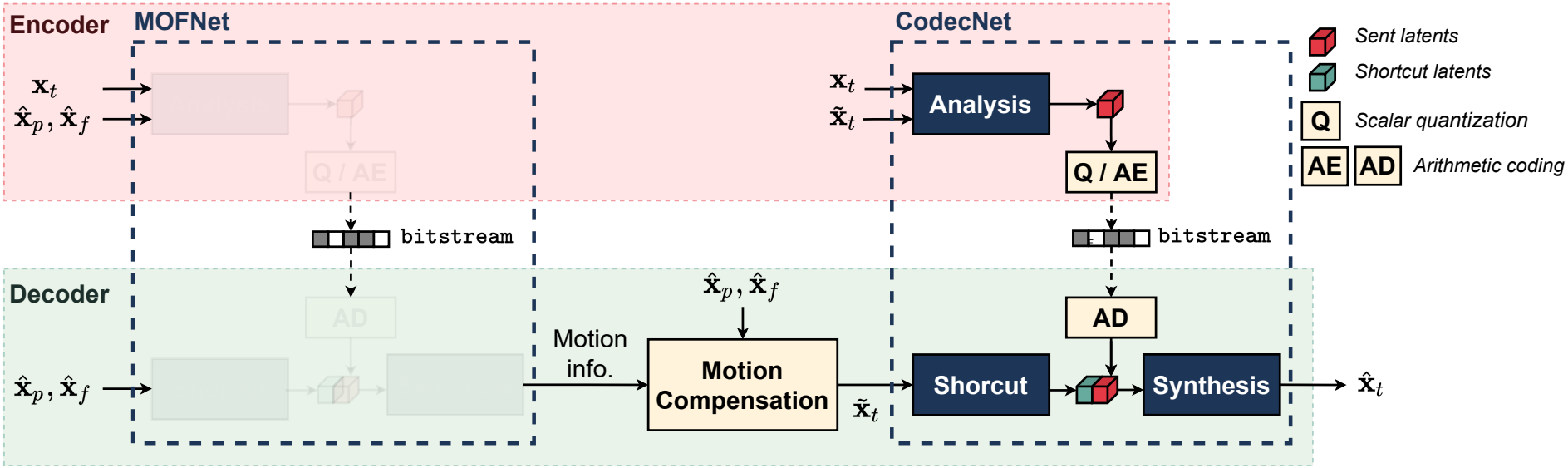
- Shortcut: exploit decoder information
- Analysis: transmit information not present at the decoder

Conditional Coding



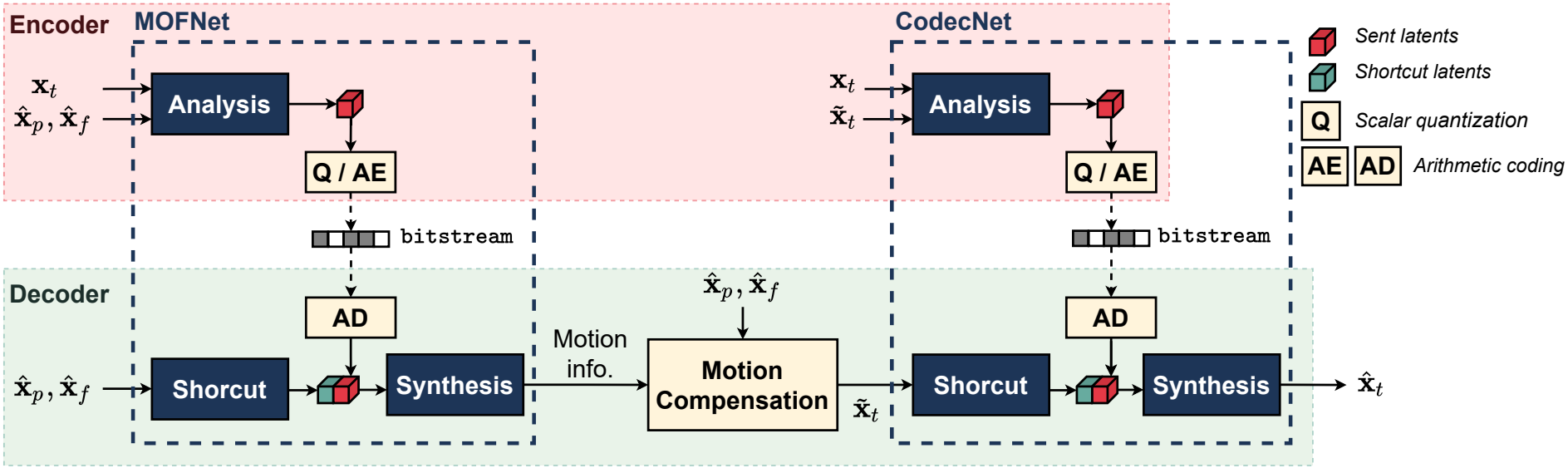
- Shortcut: exploit decoder information
- Analysis: transmit information not present at the decoder
- Synthesis: process both shortcut and analysis latents

Coding Scheme Overview



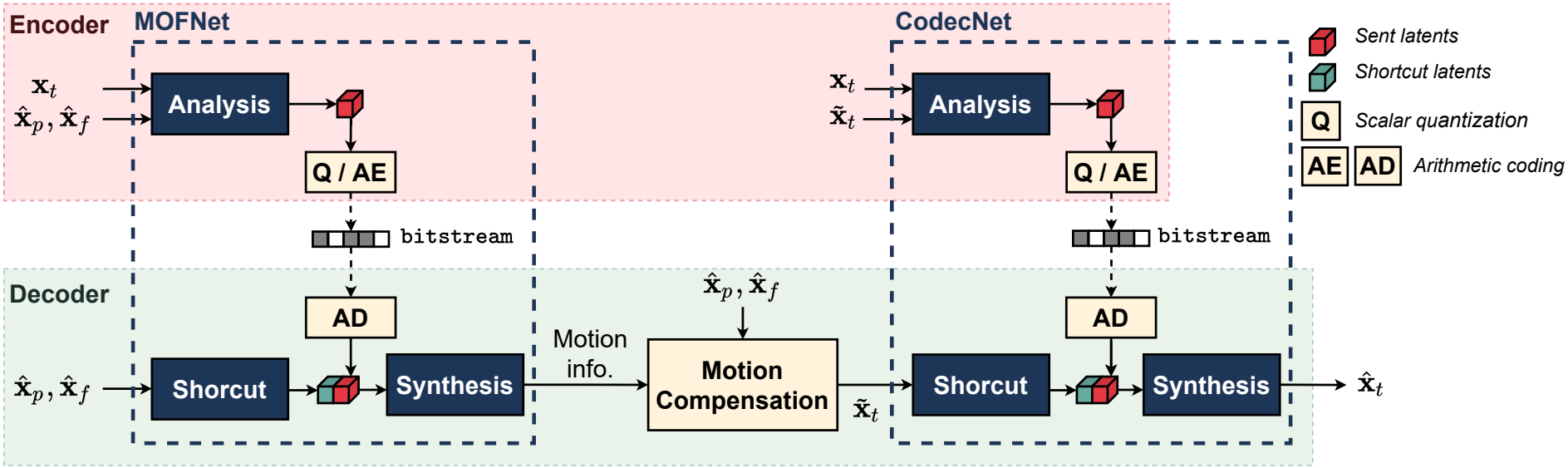
- Conditional Coding for CodecNet

Coding Scheme Overview



- Conditional Coding for CodecNet and MOFNet

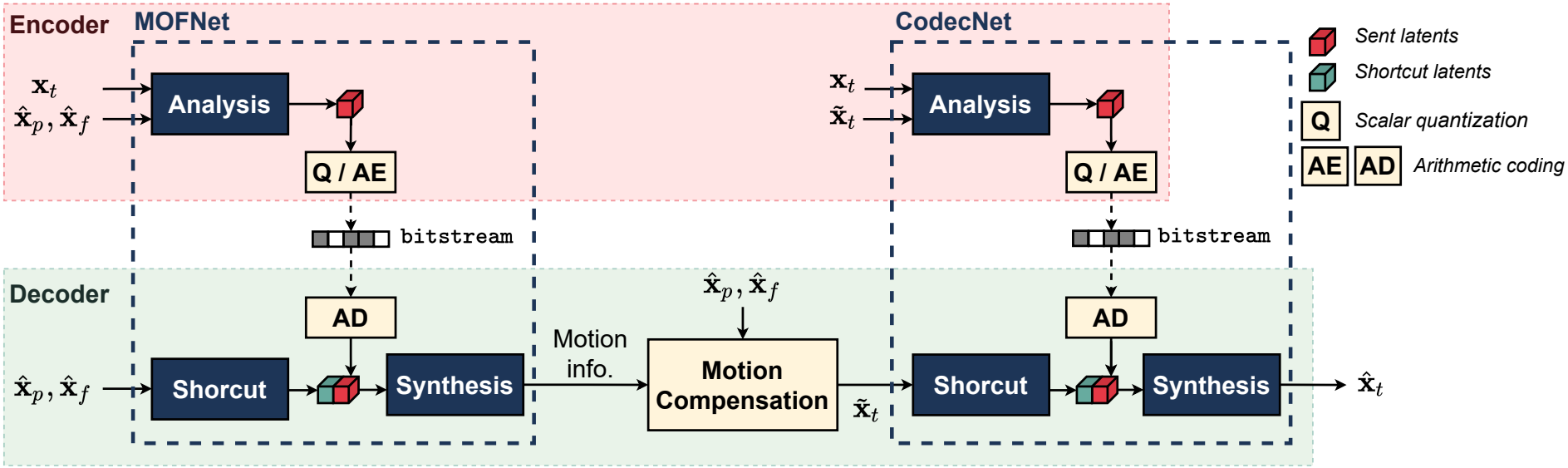
Coding Scheme Overview



- Conditional Coding for CodecNet and MOFNet
  - **Retrieve motion information** from the references

Motion from the shortcut

Coding Scheme Overview

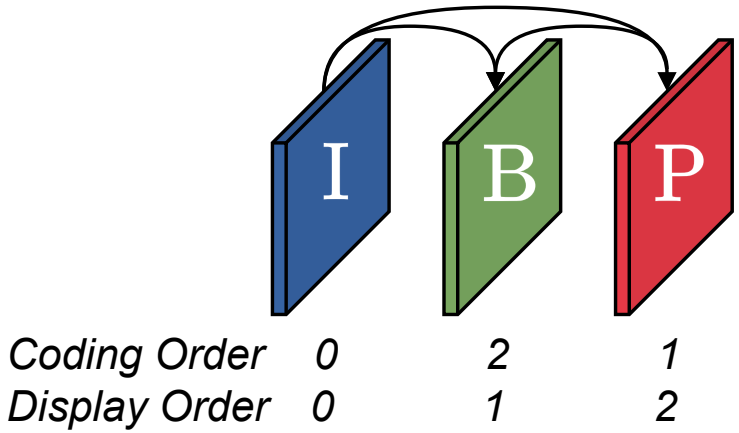


- Conditional Coding for CodecNet and MOFNet
  - **Retrieve motion information** from the references
  - The analysis transform sends only a **small** correction

Motion from the shortcut

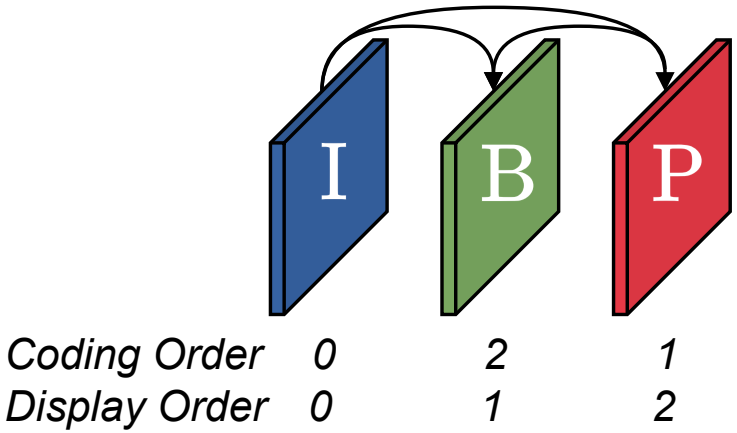
Motion from the analysis

Training process



- The system is trained to code an **entire GOP** (+ the first I)
  - Our system learns to code I, P & B-frames

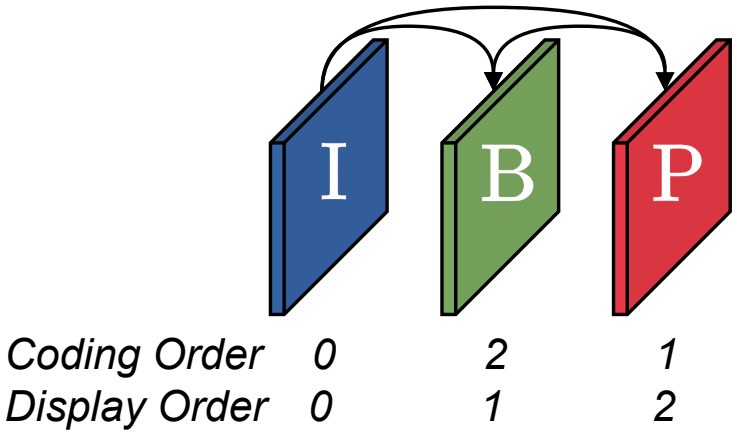
Training process



- The system is trained to code an **entire GOP** (+ the first I)
  - Our system learns to code I, P & B-frames
- **End-to-end** training **from scratch** through a **RD-loss**

$$\mathcal{L}_\lambda = \sum_t D(\mathbf{x}_t, \hat{\mathbf{x}}_t) + \lambda R(\hat{\mathbf{x}}_t)$$

Training process

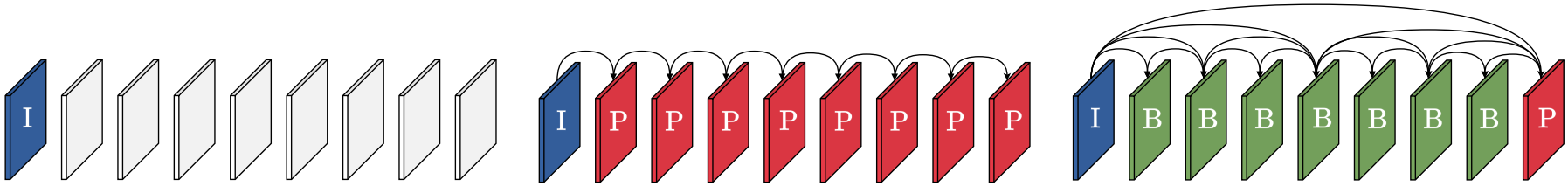


- The system is trained to code an **entire GOP** (+ the first I)
  - Our system learns to code I, P & B-frames
- **End-to-end** training **from scratch** through a **RD-loss**

$$\mathcal{L}_\lambda = \sum_t D(\mathbf{x}_t, \hat{\mathbf{x}}_t) + \lambda R(\hat{\mathbf{x}}_t)$$

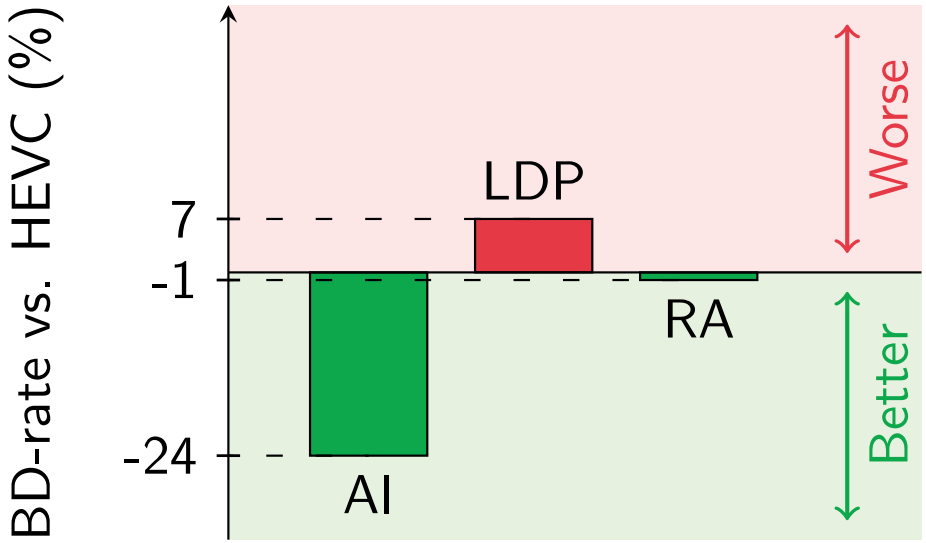
- **Single** backward for all the frames: model compression **noise propagation**

Experimental Results



All Intra (AI) *Low-delay P (LDP)* *Random Access (RA)*

- Evaluation on 3 coding configurations against HEVC



- Our **single-coder** system is **on par** with HEVC for the 3 coding config.

Thanks for your attention