ModeNet: Mode Selection Network for Learned Video Coding

Paper ID: 25

Théo LADUNE[1,2], Pierrick PHILIPPE[1], Wassim HAMIDOUCHE[2], Lu ZHANG[2], Olivier DÉFORGES[2]

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

IEEE International Workshop on Machine Learning for Signal Processing (MLSP), Sept. 2020

Introduction

- Video signals exhibit many **temporal redundancies**

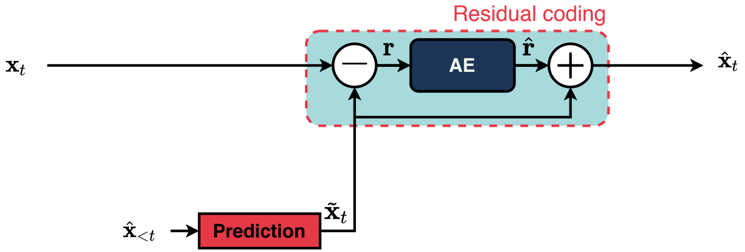
$\mathbf{x}_{t-2}$ $\mathbf{x}_{t-1}$ $\mathbf{x}_{t}$

Three consecutive frames of a video

- Video codecs leverage them with **inter-frame coding** *i.e.* using information from already received frames $\hat{\mathbf{x}}_{<t} = \{\hat{\mathbf{x}}_{t-1},\ \hat{\mathbf{x}}_{t-2} \ldots\}$ to **lower the amount of data** needed to transmit $\mathbf{x}_t$

Introduction – Problem statement

- Recent **learning-based** codecs[1,2,3] implement inter-frame coding by
    1. Computing $\tilde{\mathbf{x}}_t$ a prediction of $\mathbf{x}_t$
    2. Residual (*i.e.* prediction error $\mathbf{r} = \tilde{\mathbf{x}}_t - \mathbf{x}_t$) coding with an Auto-Encoder



- Improve 2: *Given a prediction $\tilde{\mathbf{x}}_t$, what is the best way of sending $\mathbf{x}_t$?*

[1] Lu *et al.*, *DVC: an end-to-end deep video compression framework*, **CVPR 19**
[2] Djelouah *et al.*, *Neural inter-frame compression for video coding*, **ICCV 19**
[3] Liu *et al.*, *Learned video compression via joint spatial-temporal correlation exploration*

Introduction – Contributions

- We argue that **residual coding** of the **entire frame** is not ideal
- **ModeNet** (coding mode selection network) is proposed
  - Learn and convey a pixel-wise partitioning of $\mathbf{x}_t$
  - Allow pixel-wise **coding mode competition**
- **Conditional Coding** is introduced
  - Novel Auto-Encoder architecture
  - Perform a **more complex mixture** of $\mathbf{x}_t$ and $\tilde{\mathbf{x}}_t$ than residual coding
- **ModeNet** & **Conditional Coding** achieve a 40% rate reduction compared to residual coding on a P-frame coding task

Proposed system – Frame partitioning

- Let us define two local RD costs for a pixel i

$$J_{copy,\lambda}(i) = d(\tilde{\mathbf{x}}_t, \mathbf{x}_t; i) + 0$$

Proposed system – Frame partitioning

- Let us define two local RD costs for a pixel i

$$J_{copy,\lambda}(i) = d(\tilde{\mathbf{x}}_t, \mathbf{x}_t; i) + 0 \; ; \; J_{AE,\lambda}(i) = d(\hat{\mathbf{x}}_t, \mathbf{x}_t; i) + \lambda r(\mathbf{x}_t, \tilde{\mathbf{x}}_t; i)$$

Proposed system – Frame partitioning

- Let us define two local RD costs for a pixel i

$$J_{copy,\lambda}(i) = d(\tilde{\mathbf{x}}_t, \mathbf{x}_t; i) + 0 \ ; \ J_{AE,\lambda}(i) = d(\hat{\mathbf{x}}_t, \mathbf{x}_t; i) + \lambda r(\mathbf{x}_t, \tilde{\mathbf{x}}_t; i)$$

- Let us define $\mathcal{S}$, the set pixels of $\mathbf{x}_t$ verifying

$$\mathcal{S} = \{x_{t,i} \mid x_{t,i} \in \mathbf{x}_t,\ J_{copy,\lambda}(i) < J_{AE,\lambda}(i)\}$$

Proposed system – Frame partitioning

- Let us define two local RD costs for a pixel i

$$J_{copy,\lambda}(i) = d(\tilde{\mathbf{x}}_t, \mathbf{x}_t; i) + 0 \ ; \ J_{AE,\lambda}(i) = d(\hat{\mathbf{x}}_t, \mathbf{x}_t; i) + \lambda r(\mathbf{x}_t, \tilde{\mathbf{x}}_t; i)$$

- Let us define $\mathcal{S}$, the set pixels of $\mathbf{x}_t$ verifying

$$\mathcal{S} = \{x_{t,i} \mid x_{t,i} \in \mathbf{x}_t,\ J_{copy,\lambda}(i) < J_{AE,\lambda}(i)\}$$

- $\mathcal{S}$ allows to partition $\mathbf{x}_t$ into two coding modes
  - **Skip** (prediction copy) for pixels in $\mathcal{S}$
  - **Transmission** with an Auto-Encoder for pixels in $\bar{\mathcal{S}}$

Proposed system – Frame partitioning

- Let us define two local RD costs for a pixel i

$$J_{copy,\lambda}(i) = d(\tilde{\mathbf{x}}_t, \mathbf{x}_t; i) + 0 \ ; \ J_{AE,\lambda}(i) = d(\hat{\mathbf{x}}_t, \mathbf{x}_t; i) + \lambda r(\mathbf{x}_t, \tilde{\mathbf{x}}_t; i)$$

- Let us define $\mathcal{S}$, the set pixels of $\mathbf{x}_t$ verifying

$$\mathcal{S} = \{x_{t,i} \mid x_{t,i} \in \mathbf{x}_t,\ J_{copy,\lambda}(i) < J_{AE,\lambda}(i)\}$$

- $\mathcal{S}$ allows to partition $\mathbf{x}_t$ into two coding modes
  - **Skip** (prediction copy) for pixels in $\mathcal{S}$
  - **Transmission** with an Auto-Encoder for pixels in $\bar{\mathcal{S}}$
- Handcrafting $\mathcal{S}$ is **not trivial** as it depends on past & future pixels

Proposed system – Frame partitioning

- Let us define two local RD costs for a pixel i

$$J_{copy,\lambda}(i) = d(\tilde{\mathbf{x}}_t, \mathbf{x}_t; i) + 0 \; ; \; J_{AE,\lambda}(i) = d(\hat{\mathbf{x}}_t, \mathbf{x}_t; i) + \lambda r(\mathbf{x}_t, \tilde{\mathbf{x}}_t; i)$$

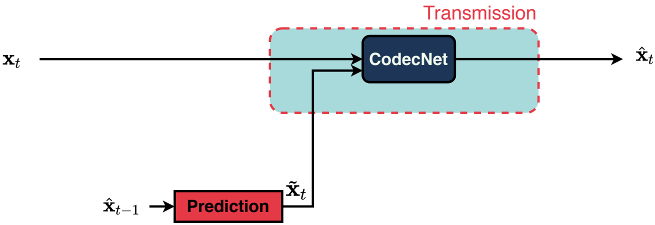
- Let us define $\mathcal{S}$, the set pixels of $\mathbf{x}_t$ verifying

$$\mathcal{S} = \{x_{t,i} \mid x_{t,i} \in \mathbf{x}_t,\; J_{copy,\lambda}(i) < J_{AE,\lambda}(i)\}$$

- $\mathcal{S}$ allows to partition $\mathbf{x}_t$ into two coding modes
  - **Skip** (prediction copy) for pixels in $\mathcal{S}$
  - **Transmission** with an Auto-Encoder for pixels in $\bar{\mathcal{S}}$
- Handcrafting $\mathcal{S}$ is **not trivial** as it depends on past & future pixels
- This work introduces a mode selection network **ModeNet**
  - Learn the partitioning of $\mathbf{x}_t$ into $\mathcal{S}$ and $\bar{\mathcal{S}}$
  - Convey it to the decoder

Proposed system

- CLIC20 P-frame test conditions[4]
  - **One lossless reference** frame: $\hat{\mathbf{x}}_{<t} = \hat{\mathbf{x}}_{t-1} = \mathbf{x}_{t-1}$
- CodecNet is a coding system (residual or more complex)

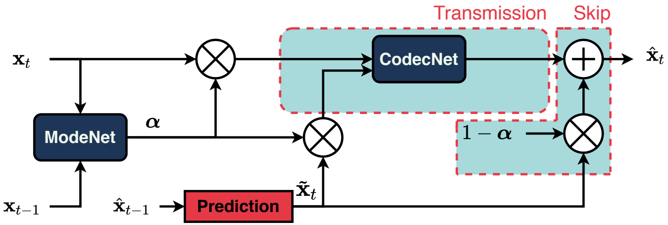


$$\hat{\mathbf{x}}_t = c(\mathbf{x}_t, \tilde{\mathbf{x}}_t)$$

[4] *Challenge on Learned Image Compression*, `www.compression.cc`, **CVPR 20**

Proposed system

- CLIC20 P-frame test conditions[4]
  - **One lossless reference** frame: $\hat{\mathbf{x}}_{<t} = \hat{\mathbf{x}}_{t-1} = \mathbf{x}_{t-1}$
- CodecNet is a coding system (residual or more complex)
- ModeNet is added to a Transmission-only system
  - $\boldsymbol{\alpha} \in [0,1]^{H \times W}$ is a **continuous pixel-wise** weighting

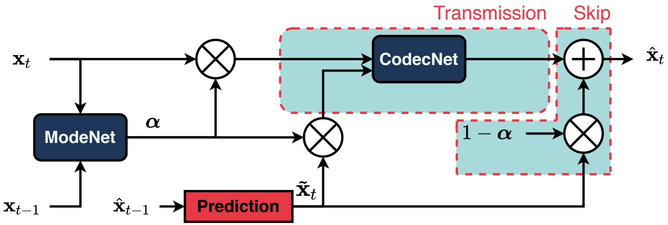


$$\hat{\mathbf{x}}_t = c(\boldsymbol{\alpha} \odot \mathbf{x}_t, \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t) + (1-\boldsymbol{\alpha}) \odot \tilde{\mathbf{x}}_t$$

[4] *Challenge on Learned Image Compression*, `www.compression.cc`, **CVPR 20**

Proposed system

- CLIC20 P-frame test conditions[4]
  - **One lossless reference** frame: $\hat{\mathbf{x}}_{<t} = \hat{\mathbf{x}}_{t-1} = \mathbf{x}_{t-1}$
- CodecNet is a coding system (residual or more complex)
- ModeNet is added to a Transmission-only system
  - $\boldsymbol{\alpha} \in [0,1]^{H \times W}$ is a **continuous pixel-wise** weighting
- Naive prediction: $\tilde{\mathbf{x}}_t = \hat{\mathbf{x}}_{t-1}$

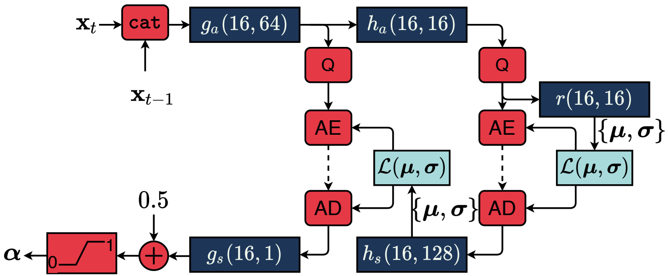


$$\hat{\mathbf{x}}_t = c(\boldsymbol{\alpha} \odot \mathbf{x}_t, \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t) + (1-\boldsymbol{\alpha}) \odot \tilde{\mathbf{x}}_t$$

[4] *Challenge on Learned Image Compression*, `www.compression.cc`, **CVPR 20**

Implementation – ModeNet architecture

- ModeNet architecture: **standard Auto-Encoder** with hyperprior (AE-HP)[5]



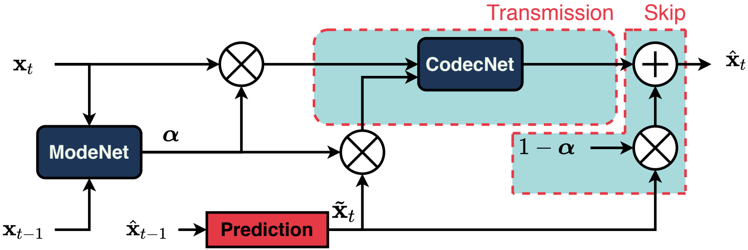
Transform syntax is f (internal features, output features)

- Lightweight: 200 000 parameters → **10%** of CodecNet parameters

[5] Minnen *et al.*, *Joint Autoregressive and Hierarchical Priors for Learned Image Compression*, **NIPS 18**

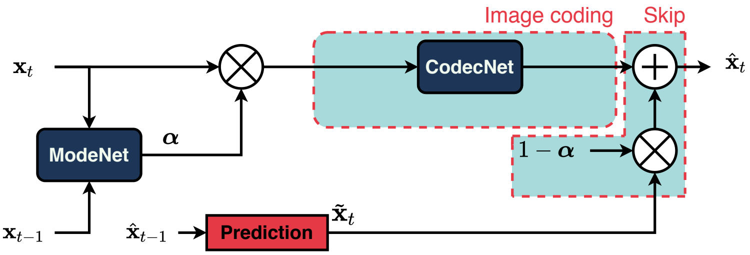
Implementation – CodecNet

- 3 configurations of CodecNet are investigated



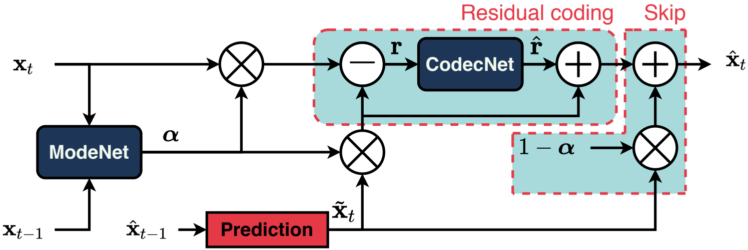
Implementation – CodecNet

- 3 configurations of CodecNet are investigated
  - Image coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t)$



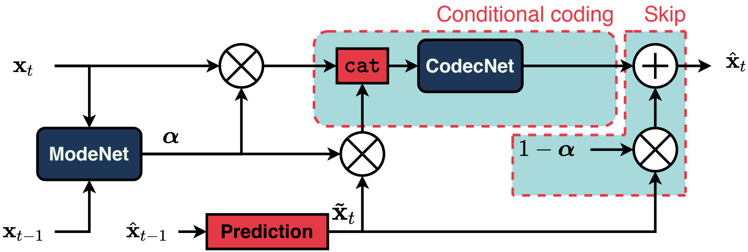
Implementation – CodecNet

- 3 configurations of CodecNet are investigated
  - Image coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t)$
  - Residual coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t - \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$



Implementation – CodecNet

- 3 configurations of CodecNet are investigated
  - Image coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t)$
  - Residual coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t - \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$
  - Conditional coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t \mid \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$

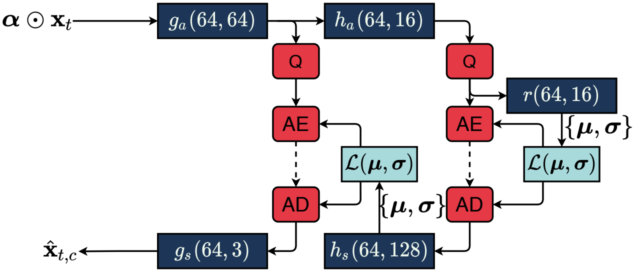


Implementation – Codec architecture

- CodecNet architecture is a **standard AE-HP**

Implementation – Codec architecture

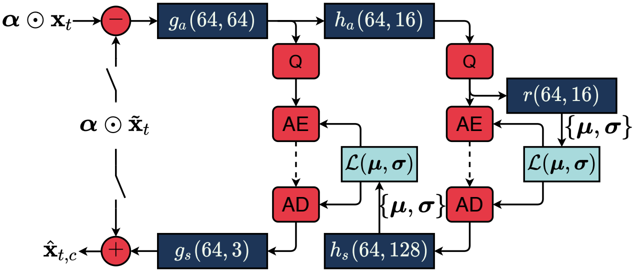
- CodecNet architecture is a **standard AE-HP**
  - Image coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t)$



Transform syntax is f(internal features, output features)

Implementation – Codec architecture

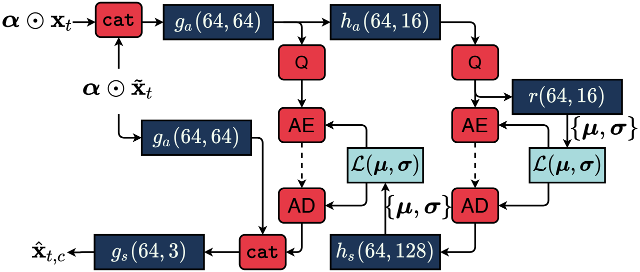
- CodecNet architecture is a **standard AE-HP**
  - Image coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t)$
  - Residual coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t - \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$



Transform syntax is $f\,(\textit{internal features}, \textit{output features})$

Implementation – Codec architecture

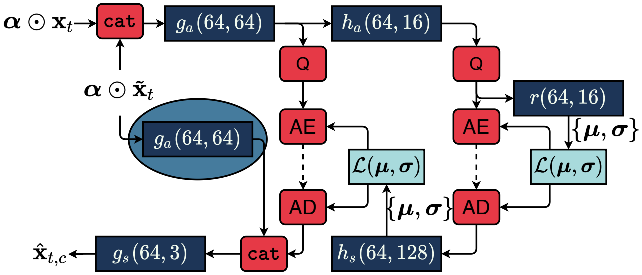
- CodecNet architecture is a **standard AE-HP**
  - Image coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t)$
  - Residual coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t - \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$
  - Conditional coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t \mid \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$



Transform syntax is f(internal features, output features)

Implementation – Codec architecture

- CodecNet architecture is a **standard AE-HP**
  - Image coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t)$
  - Residual coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t - \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$
  - Conditional coding: $c(\boldsymbol{\alpha} \odot \mathbf{x}_t \mid \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t)$



Transform syntax is f(internal features, output features)

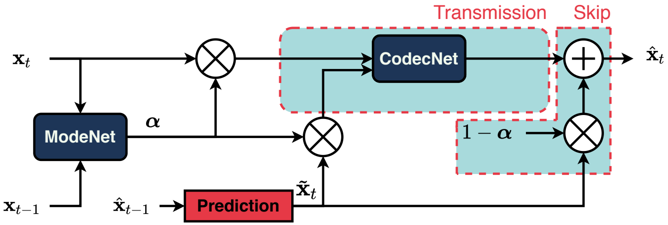
Implementation – Training

- **End-to-end** training with rate distortion cost: **no dedicated α loss**

$$\mathcal{L}_\lambda = \mathrm{D}(\mathbf{x}_t, \hat{\mathbf{x}}_t) + \lambda\,(R_m + R_c)$$

- CLIC20 P-frame test condition

$$\mathrm{D}(\mathbf{x}_t, \hat{\mathbf{x}}_t) = 1 - \text{MS-SSIM}(\mathbf{x}_t, \hat{\mathbf{x}}_t)$$



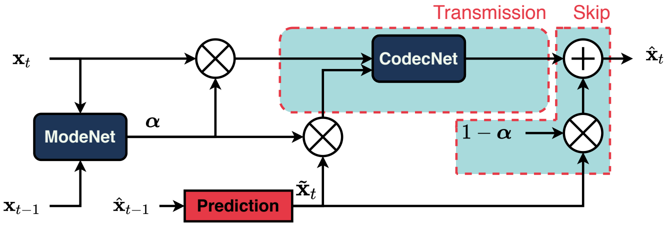
- **End-to-end** training with rate distortion cost: **no dedicated $\boldsymbol{\alpha}$** loss

$$\mathcal{L}_\lambda = \mathrm{D}(\mathbf{x}_t, \hat{\mathbf{x}}_t) + \lambda\,(R_m + R_c)$$

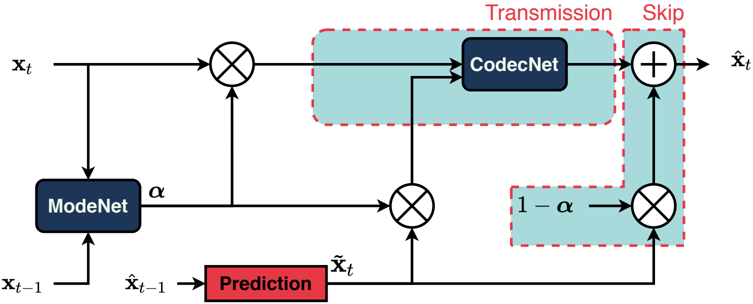
- CLIC20 P-frame test condition

$$\mathrm{D}(\mathbf{x}_t, \hat{\mathbf{x}}_t) = 1 - \text{MS-SSIM}(\mathbf{x}_t, \hat{\mathbf{x}}_t)$$

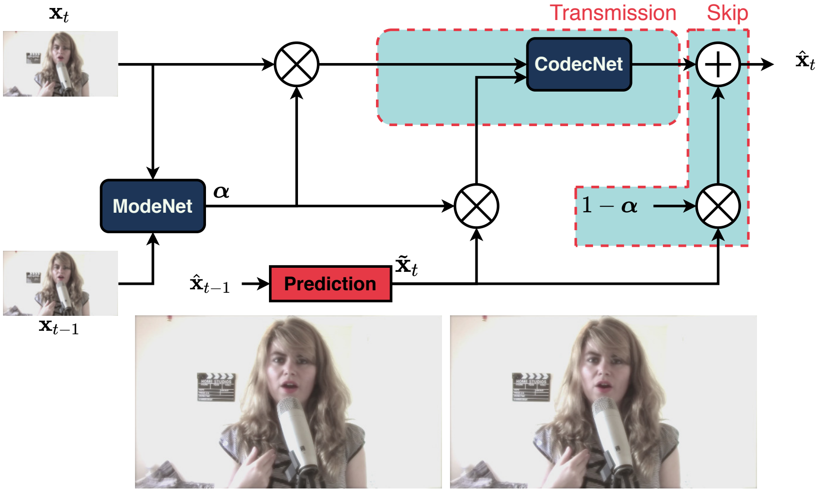
- 2 training stages
  1. Warm-up: CodecNet only, $\boldsymbol{\alpha} = 1$ for one half of $\mathbf{x}_t$, 0 for the other
  2. Alternate: Train ModeNet and CodecNet alternatively



Experimental results – Visualisation

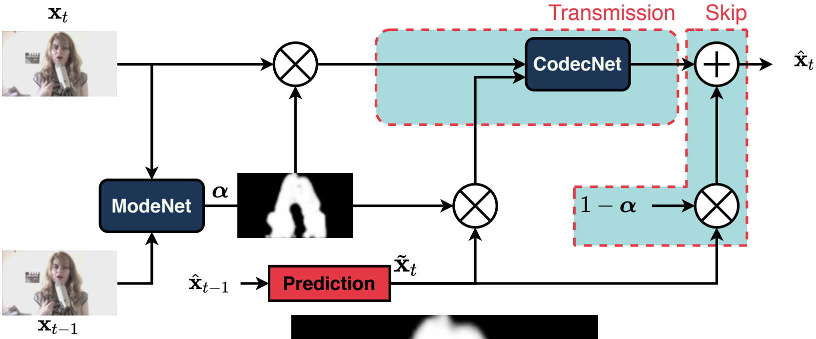


Experimental results – Visualisation



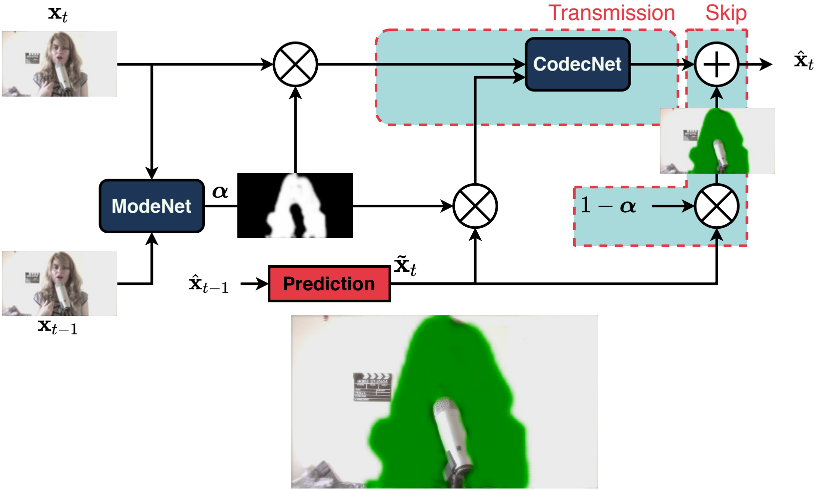
Input frames $(\mathbf{x}_{t-1}, \mathbf{x}_t)$

Experimental results – Visualisation

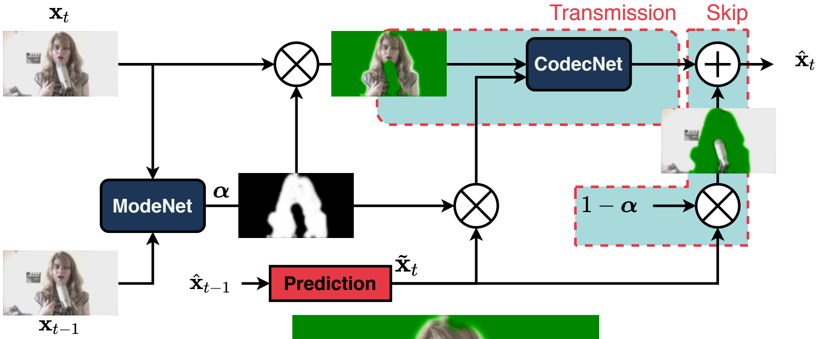


Coding mode selection $\boldsymbol{\alpha}$

Experimental results – Visualisation

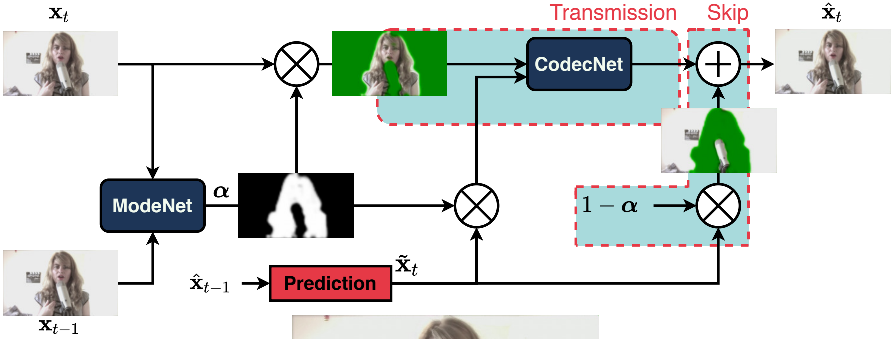


Skip part $(1 - \boldsymbol{\alpha}) \odot \tilde{\mathbf{x}}_t$



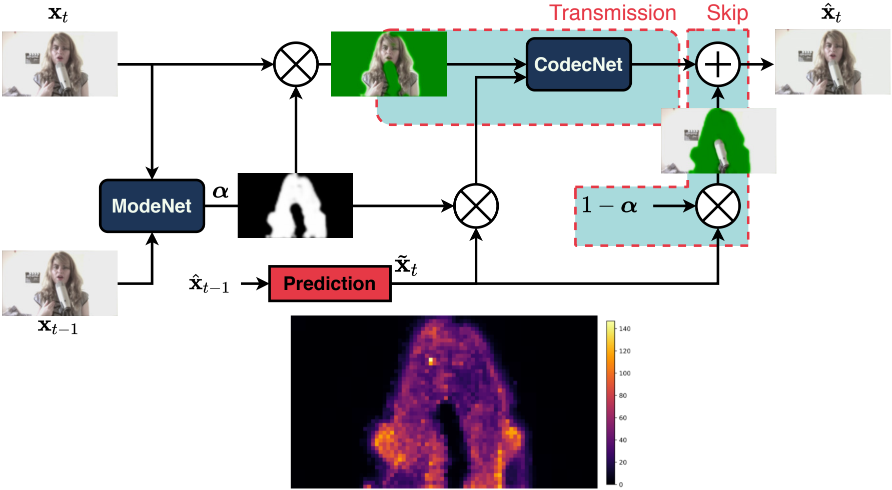
Transmission part $\boldsymbol{\alpha} \odot \mathbf{x}_t$

Experimental results – Visualisation



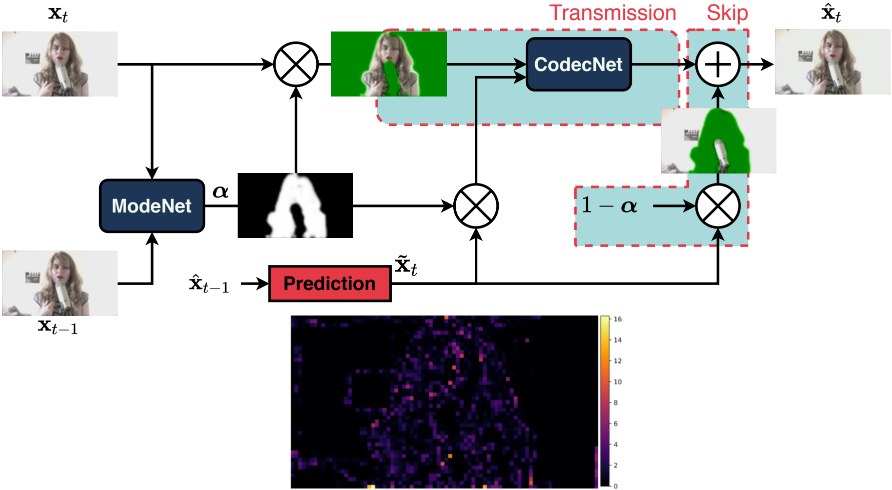
$$\hat{\mathbf{x}}_t = c(\boldsymbol{\alpha} \odot \mathbf{x}_t, \boldsymbol{\alpha} \odot \tilde{\mathbf{x}}_t) + (1 - \boldsymbol{\alpha}) \odot \tilde{\mathbf{x}}_t$$

Experimental results – Visualisation



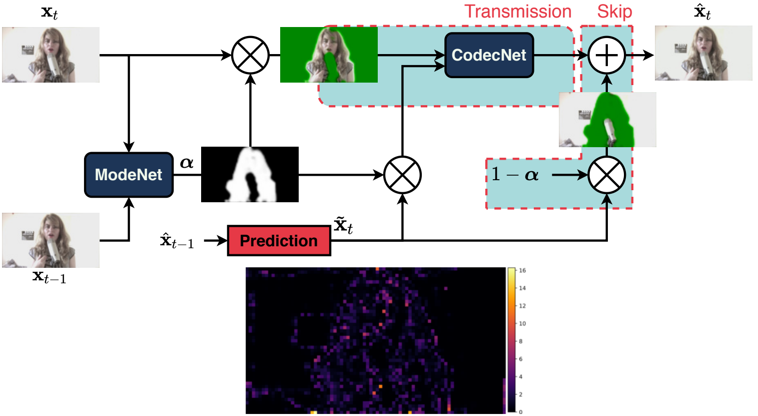
Rate CodecNet $R_c = 0.071$ bpp

Experimental results – Visualisation



Rate ModeNet $R_m = 0.005$ bpp

Experimental results – Visualisation



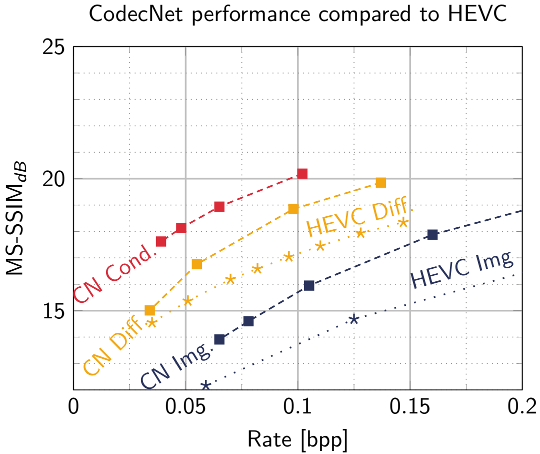
Rate ModeNet $R_m = 0.005$ bpp

- The proposed ModeNet
  - Learns a **complex partitioning**, trained only with a rate distortion loss
  - Conveys the partitioning at **very low rate**
  - Has only 200 000 parameters $\rightarrow$ 10% of CodecNet

Performance

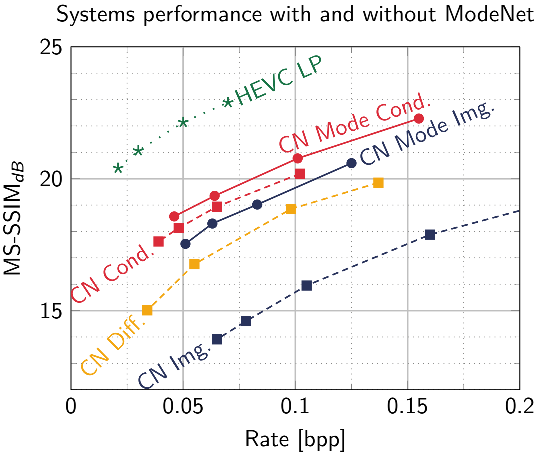
- This works follow CLIC20 P-frame coding test conditions
  - Quality metric is **MS-SSIM**
  - Rate target is **0.075 bpp**
  - CLIC20 P-frame validation set
- Two experiments carried out
  1. Training and test of **CodecNet alone** vs. HEVC
  2. Training and test of the **complete system**: CodecNet + ModeNet

Experimental results – CodecNet



- CodecNet **outperforms HEVC**
- Conditional coding **outperforms residual** (Diff.) coding

Experimental results – Complete system



- ModeNet improves image and conditional coding
- ModeNet + Conditional coding: **40% rate reduction** / residual coding
- HEVC LP outperforms all systems thanks to a **motion-compensated** $\tilde{\mathbf{x}}_t$

Conclusion

- This paper proposes ModeNet, a coding mode selection network
  1. Learn **complex partitioning** through **end-to-end** training
  2. Convey the partitioning at **very low rate**
  3. **Lightweight** AE with hyperprior
  4. Can be **integrated seamlessly** into existing learning-based coding scheme to allow **coding modes competition**

Conclusion

- This paper proposes ModeNet, a coding mode selection network
  1. Learn **complex partitioning** through **end-to-end** training
  2. Convey the partitioning at **very low rate**
  3. **Lightweight** AE with hyperprior
  4. Can be **integrated seamlessly** into existing learning-based coding scheme to allow **coding modes competition**

- Tested on a P-frame coding task
  - Using ModeNet to **select** the best coding mode increases performance
  - ModeNet arbitrating between **skip and conditional coding** achieves a **40% rate reduction** compared to residual coding

Conclusion

- This paper proposes ModeNet, a coding mode selection network
  1. Learn **complex partitioning** through **end-to-end** training
  2. Convey the partitioning at **very low rate**
  3. **Lightweight** AE with hyperprior
  4. Can be **integrated seamlessly** into existing learning-based coding scheme to allow **coding modes competition**

- Tested on a P-frame coding task
  - Using ModeNet to **select** the best coding mode increases performance
  - ModeNet arbitrating between **skip and conditional coding** achieves a **40% rate reduction** compared to residual coding

- This work has already been extended: ModeNet also transmits **motion information** to improve the prediction process[6]

[6] Ladune *et al.*, *Optical Flow and Mode Selection for Learning-based Video Coding*, **MMSP 20**