Video Coding using Deep Learning

Focus: Deep Contextual Video Compression

Gerald Schuller

TU Ilmenau

July 17, 2024

Outline

Deep Learning for Video Coding
- Standardization Exploration
- Standardization Approaches

Deep Contextual Video Compression (DCVC)

Components of Deep Contextual Video Compression
- Autoencoders
- Context Model

Training Deep Video Compression Models

Evaluation Metrics

Future Directions

Conclusion

Deep Learning for Video Coding

- Overview of deep learning
- Application of deep learning in video compression
- Advantages over traditional methods

Standardization Exploration

- JVET is the "Joint Video Experts Team" between ITU and MPEG,
- it standardized VVC and also conducts "exploration studies of candidate technology for potential future video coding standardization action." Neural-Network-Based Video Coding
- `https://www.itu.int/en/ITU-T/studygroups/2017-2020/16/Pages/video/jvet.aspx`

Explorations

- Sharp, 2022: "Overview of technologies considered in JVET's neural network-based video coding exploration" https://jvet-experts.org/doc_end_user/current_document.php?id=11448
- Huawei, 2022: "Methodologies for evaluation and complexity assessment of neural network based video coding technology" https://jvet-experts.org/doc_end_user/current_document.php?id=11450

Description of Approaches

- Neural Network based Video Coding in JVET: https://www.kibme.org/resources/journal/20230118120317089.pdf
- Designs and Implementations in Neural Network-based Video Coding: https://arxiv.org/pdf/2309.05846
- MPEG protococol of explorations: https://www.mpeg.org/standards/Explorations/36/
- Description of algorithms and software in neural network-based video coding (NNVC) version 7: https://www.mpeg.org/wp-content/uploads/mpeg_meetings/146_Rennes/w23772.zip

Two basic Approaches

- Replace Tools in traditional video coders by deep learning approaches
- Build an end-to-end deep learning based video coder
- In the following, the latter will be described

Deep Learning for Video Coding: New Tool Set

- Autoencoder
- Abstract motion compensation in latent space
- Learned entropy coder

Deep Contextual Video Compression (DCVC)

- DCVC is developed by Microsoft: https://proceedings.neurips.cc/paper/2021/file/96b250a90d3cf0868c83f8c965142d2a-Paper.pdf

DCVC Github

- DCVC can be installed from the follwoing repository,
- It includes several versions and links to training checkpoints for psnr and msssim
- `https://github.com/microsoft/DCVC`
- Our Colab notebook implementation example: `https://github.com/TUIlmenauAMS/Videocoding/blob/main/DCVC_FM_example.ipynb`
- Let it run (it takes ca. 14 Minutes)
- Our Colab example for using webcam video recording: `https://github.com/TUIlmenauAMS/Videocoding/blob/main/DCVC_FM_Record_Video.ipynb`

Resulting Original vs. Reconstructed Frame, 1920x1080 Pixel

(a) Original Image

(b) Reconstructed Image

Figure: Comparison of Original and Reconstructed Frames, at about 0.016 bits per pixel, or 1920*1080*25*0.016= 829 kb/s. At lowest quality q0 it goes down to 0.00087 bits per pixel or 45 kb/s.

DCVC Comparison with traditional methods

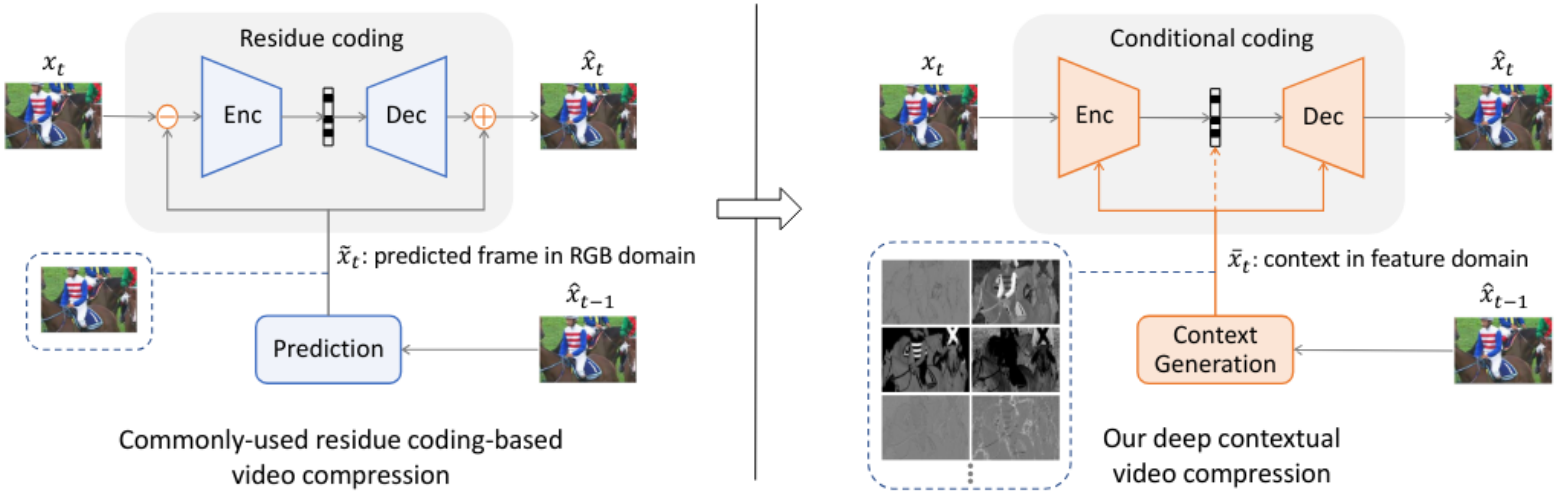


Figure 1: Paradigm shift from residue coding-based framework to conditional coding-based framework. x_t is the current frame. $\hat{x}_t$ and $\hat{x}_{t-1}$ are the current and previous decoded frames. The orange dashed line means that the context is also used for entropy modeling.

Figure: From: Jiahao Li, Bin Li, Yan Lu: "Deep Contextual Video Compression"

DCVC Detailed Structure

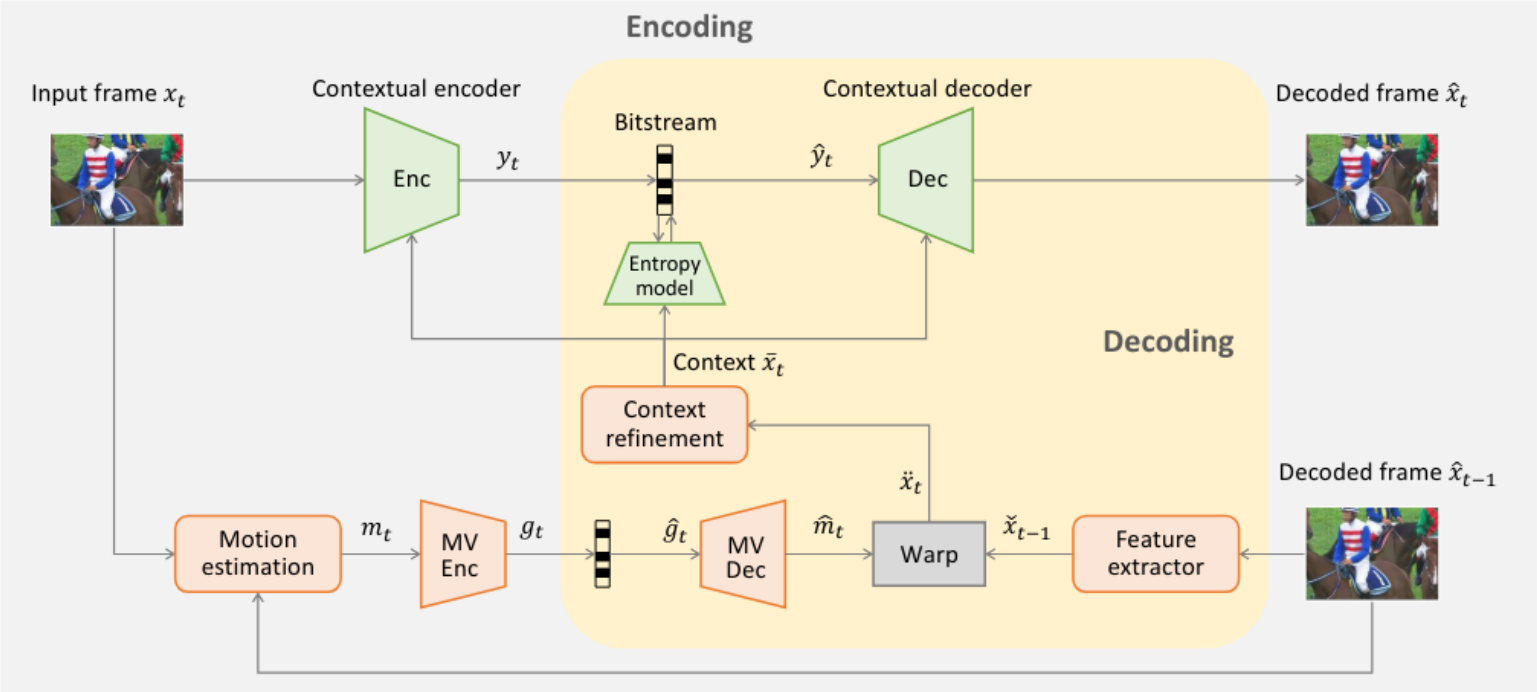


Figure: From: Jiahao Li, Bin Li, Yan Lu: "Deep Contextual Video Compression"

DCVC Bit Per Pixel vs Quality in MS-SSIM

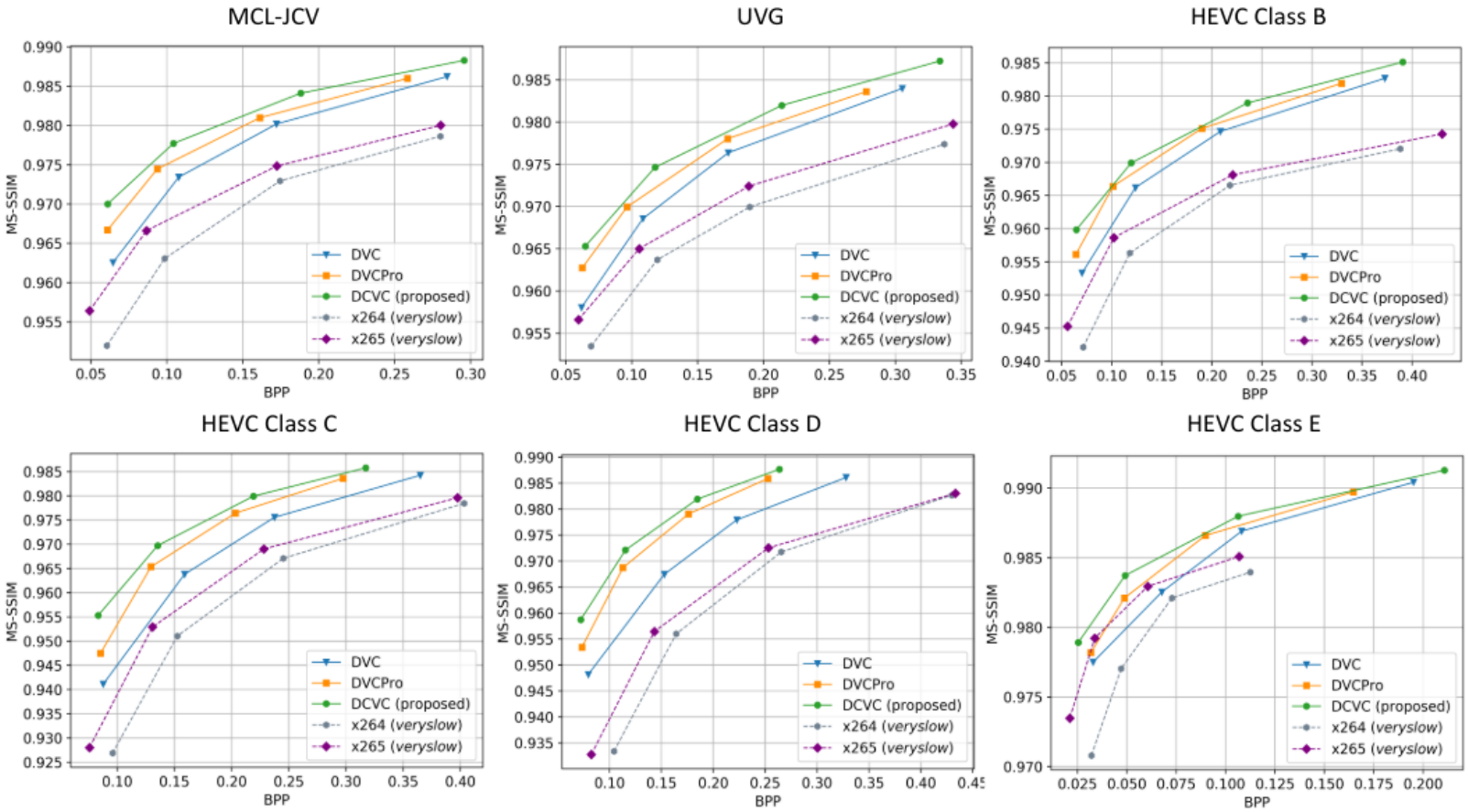


Figure 6: MS-SSIM and bitrate comparison. The DL-based codecs are fine-tuned for MS-SSIM.

Figure: DCVC gets comparable quality at roughly half the bitrate of X264 and x265. From: Jiahao Li, Bin Li, Yan Lu: "Deep Contextual Video Compression"

Autoencoders as Core Component of DCVC

- Definition and function of autoencoders
- Types of autoencoders (e.g., Variational Autoencoders, Convolutional Autoencoders)
- Role of autoencoders in video compression

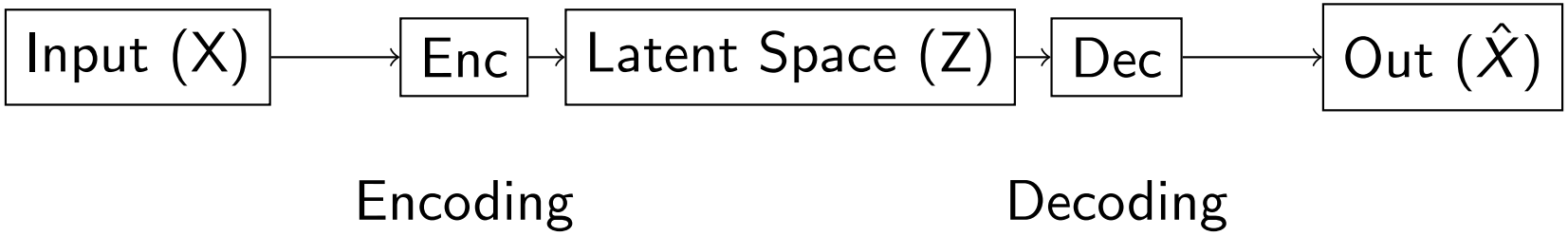
Autoencoders: Definition

- **Definition**
  - An autoencoder is a type of artificial neural network used to learn efficient codings of input data.
  - It consists of two main parts: an encoder that compresses the input into a latent-space representation, and a decoder that reconstructs the input from this representation.

Autoencoders: Function

- **Function**
  - **Dimensionality Reduction**
    - Autoencoders reduce the dimensionality of data, making it easier to process and analyze.
  - **Feature Learning**
    - They automatically learn the most relevant features of the data without human intervention.
  - **Data Reconstruction**
    - By reconstructing the input data from the encoded representation, autoencoders can be used for data denoising and anomaly detection.
  - **Compression in Video Coding**
    - In video compression, autoencoders help in reducing the size of video frames by learning efficient representations, which are then used for reconstructing high-quality frames.

Structure of an Autoencoder



- **Input (X):** Original data.
- **Encoder:** Compresses the input data into a latent-space representation (Z).
- **Latent Space (Z):** Compressed representation of the data.
- **Decoder:** Reconstructs the data from the latent-space representation.
- **Output ($\hat{X}$):** Reconstructed data.

Convolutional Autoencoder

- A convolutional autoencoder uses a convolutional layer as front and back end
- This is particularly useful for large frames, because it reduces the number of weights in comparison to a linear or fully connected layer
- A convolutional layer can be seen as a filter bank
- Analysis filter bank in the encoder, synthesis filter bank in the decoder
- The latent space then can be seen as subband representation
- The reduced dimension of the latent space then leads to "over critical" sampled filter banks, with sampling rate larger than the number of subbands

Convolutional Autoencoder

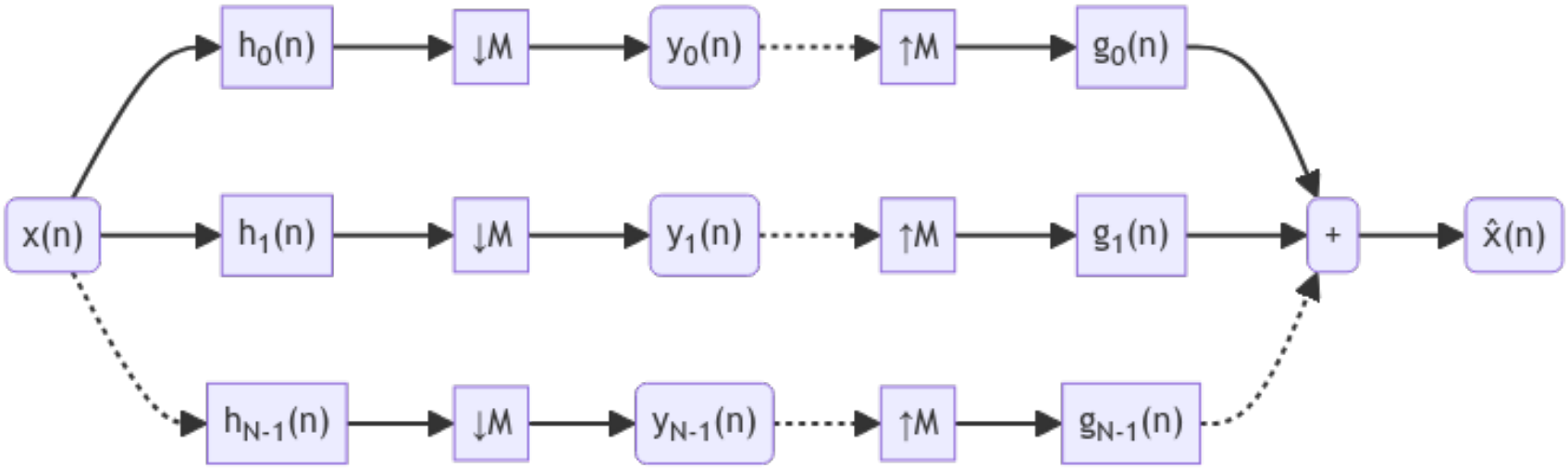


Figure: Over-critically samples filter bank, with sampling rate M larger than the number of subbands N, as model of a convolutional autoencoder. $x(n)$ can be our input frame, e.g., $\hat{x}(n)$ our reconstructed frame.

Autoencoder Audio Example Implementation

- A Colab notebook example for an autoencoder for audio:
- `https://github.com/TUIlmenauAMS/AES_Tutorial_2021`
- Chapter: "The Convolutional Autoencoder Network"

Autoencoder Image Example Implementation

- A Colab notebook example for a "Variational" Autoencoder for images
- "variational" means that it adds noise to the latent space signal. This can be seen as simulating quantization noise.
- Based on the "MNIST" handwritten image data set (only 28x28 pixels each)
- made with the help of ChatGPT,
- `https://github.com/TUIlmenauAMS/Videocoding/blob/main/imageVAE.ipynb`

Autoencoder Image Example Implementation

- Observe how the trainig- and test-losses decrease during the training eopochs. If the latter goes up consistently, training should be stopped.
- Also observe the difference between original and reconstructed images.

Context Model

- Definition of context in video compression (e.g. motion)
- Building a context model
- Using context to enhance compression

Training Deep Video Compression Models

- Data preparation
- Training process
- Loss functions and optimization

Evaluation Metrics

- Common metrics (e.g., PSNR, SSIM)
- Subjective evaluation
- Benchmark datasets and comparisons

Future Directions

- Emerging trends in video compression
- Potential improvements with deep learning
- Research challenges and opportunities

Conclusion

- Summary of key points
- Impact of deep learning on video compression
- Final thoughts

Questions

Questions?