EECE5698
Networked XR Systems

Lecture Outline for Today

- Limitations of traditional Compression
- Machine Learning based Compression

Codec Chronicles: Decoding the Shift from Classical to Neural Video Codecs

Mallesh Dasari

11 min read · Apr 11, 2022

https://mdasari.medium.com/neural-video-codecs-a-paradigm-shift-in-the-internet-video-transmission-d4f97192fd29

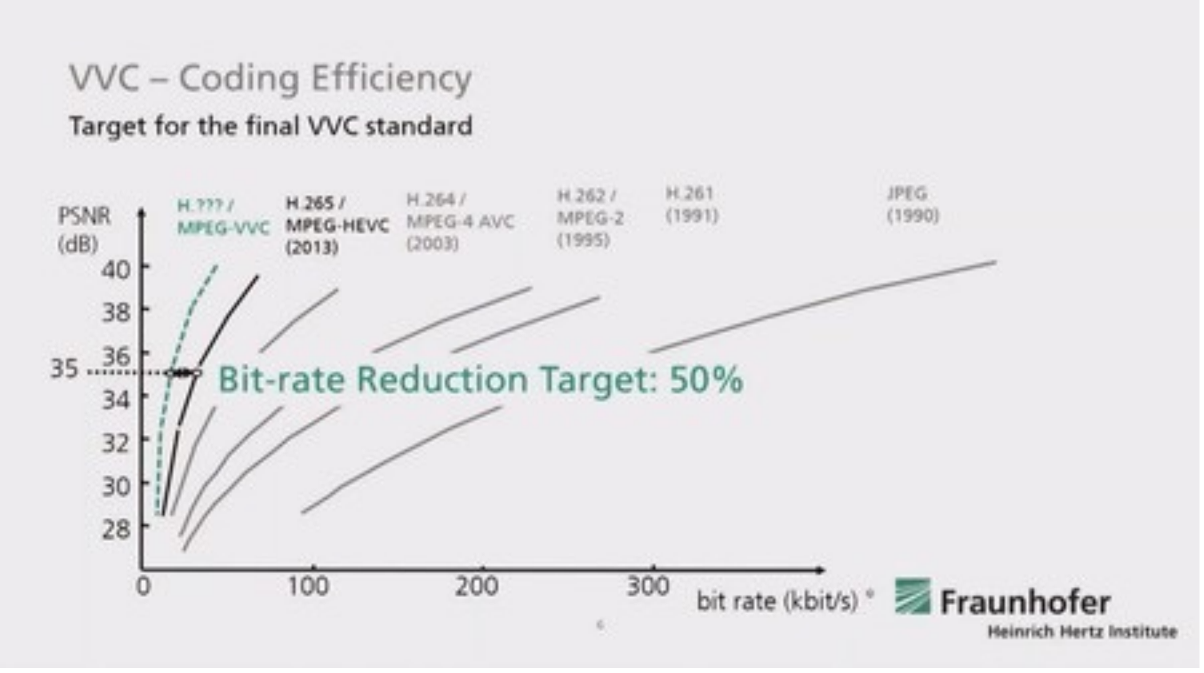
A medium blogpost I wrote a few years ago

Traditional Compression Algorithms

- Video Compression
  - H.26x series
  - VP series
- Point cloud compression
  - MPEG GPCC, VPCC
- Mesh compression
  - Vertex and connectivity compression methods (e.g., Edgebreaker or TFAN)

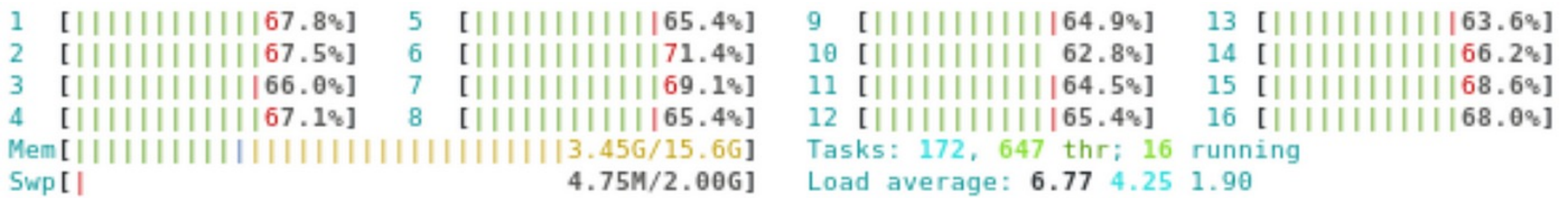
Limitations of Traditional Compression Algorithms

- Reaching a saturation point in compression ratio
  - E.g., 2D video codecs have been engineered for decades



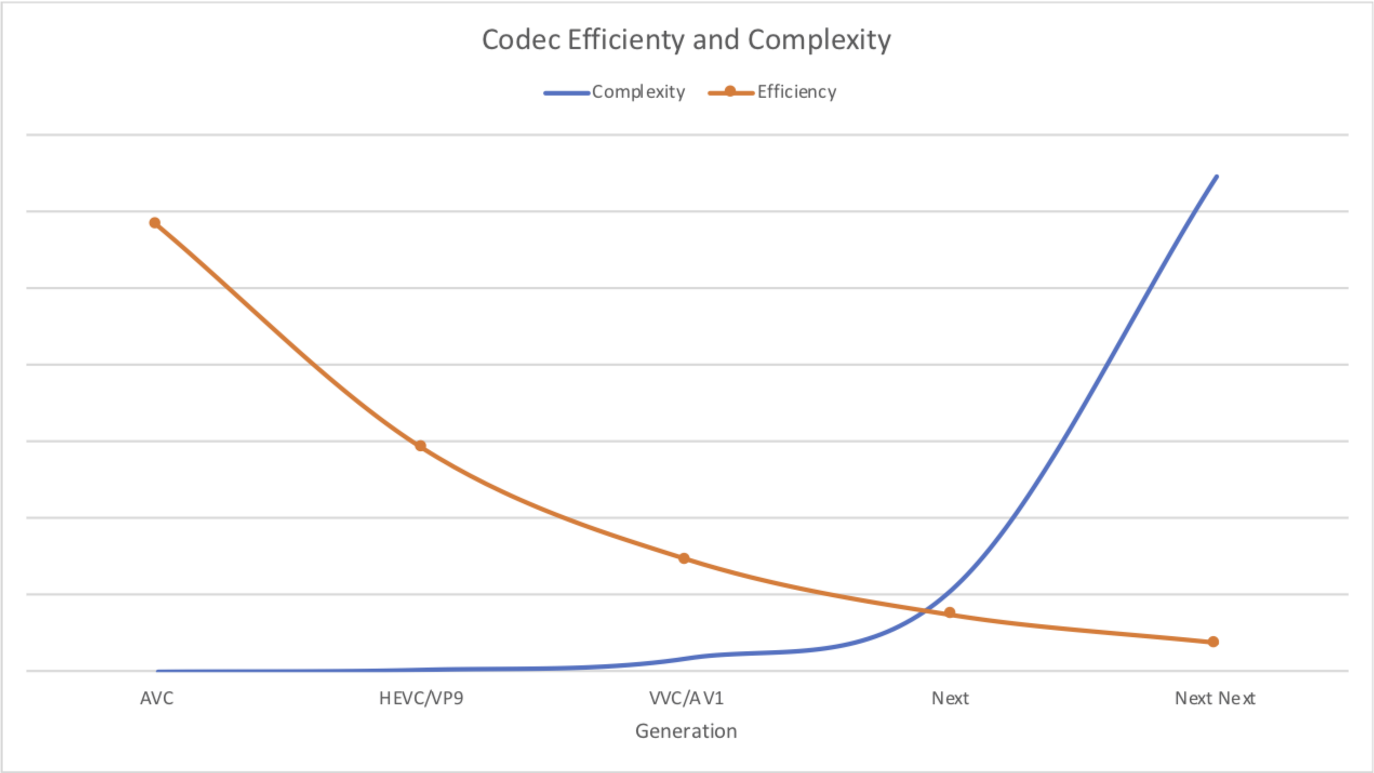
Limitations of Traditional Compression Algorithms

- Computational complexity



Computational complexity of H.264 decoding a 8K video in a Chrome browser on an Intel i9–9900K CPU with 3.60GHz and 16 cores. Even with 800% CPU usage, Chrome was not able to render the video.

Limitations of Traditional Compression Algorithms



Credits: David Ronca - Netflix

Limitations of Traditional Compression Algorithms

- Hitting the power wall too
  - Not practical to run software codecs on mobile devices or XR headsets
  - Need to be in Hardware

Limitations of Traditional Compression Algorithms

- Problems with hardware codecs
  - Slower deployment (e.g., H.264 standard was released in 2003, and it is still the most popular codec for many applications)
  - Cross-platform compatibility
  - No control for users

Limitations of Traditional Compression Algorithms

- Handcrafted design of the algorithms – difficult & takes time
  - Content unaware or difficult to make the codecs content aware

- Same codec is used across diverse settings
  - E.g., treats a low complexity same as high complex video
  - E.g., no distinction between a low res and a high res video

Limitations of Traditional Compression Algorithms

- Among others
  - Limited coordination with transport protocols
    - Synchronization issues
  - Coarse-grained compression for adaptive streaming scenarios – will be discussed in-depth in streaming lecture

ML Based Compression

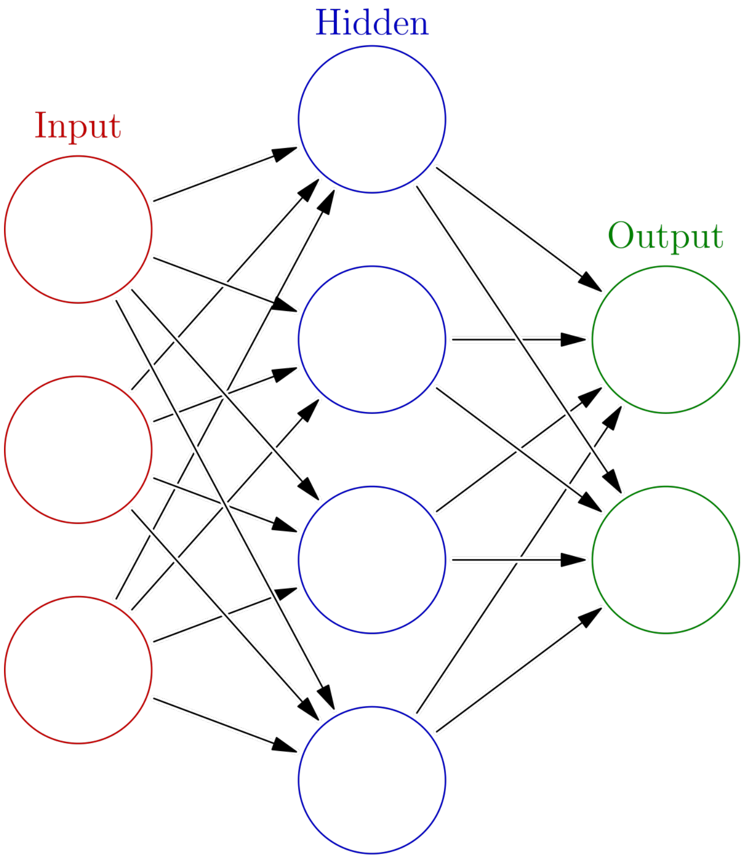
- Fundamental principles
  - Data-driven
  - Neural networks
  - Learn the weights (training a neural network model by passing a lot lot of example data samples)
  - Need large data sets for training and testing
  - Need data parallel accelerators (e.g., GPUs) for practical speeds

ML Based Compression

- Benefits
  - Can be software-driven
  - Flexible across different types of content

ML Based Compression

- Neural Networks
  - Input
  - Weights
  - Neurons
  - Activation Function
  - Output
  - Loss function
  - Change weights based on loss
  - Update weights



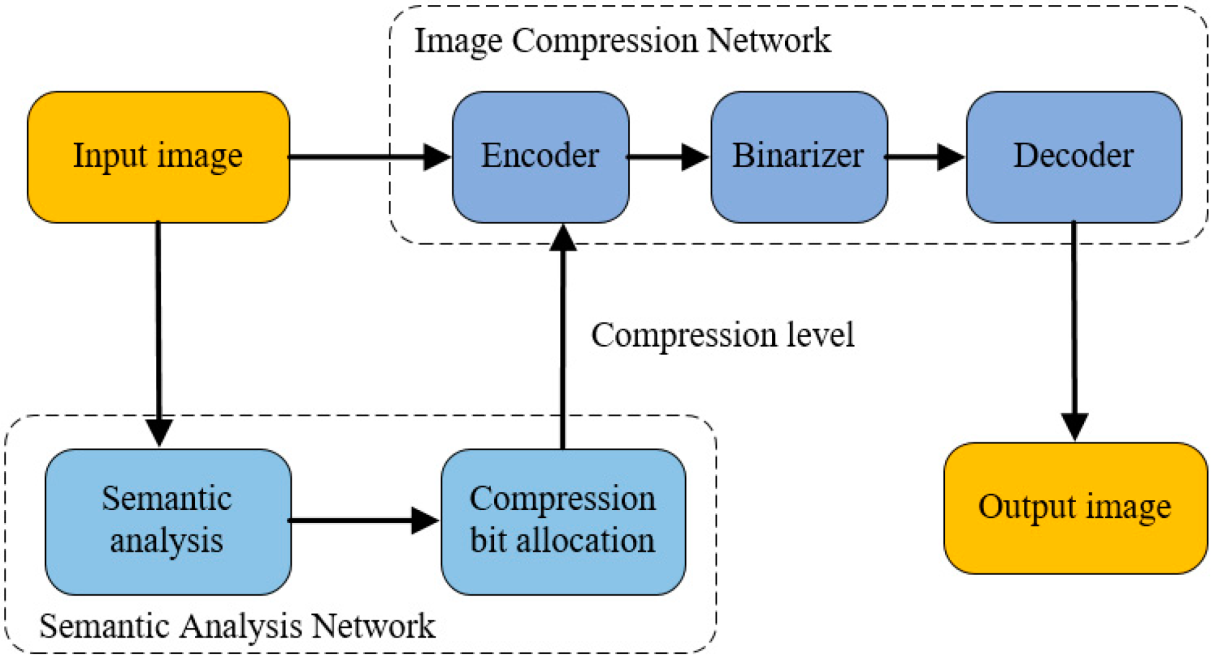
ML Based Compression

- The concept has been around for decades, but practical methods have become mainstream since 2018

- Popular models used for ML based compression
  - AutoEncoders
  - GANs
  - Diffusion Models
  - Transformers

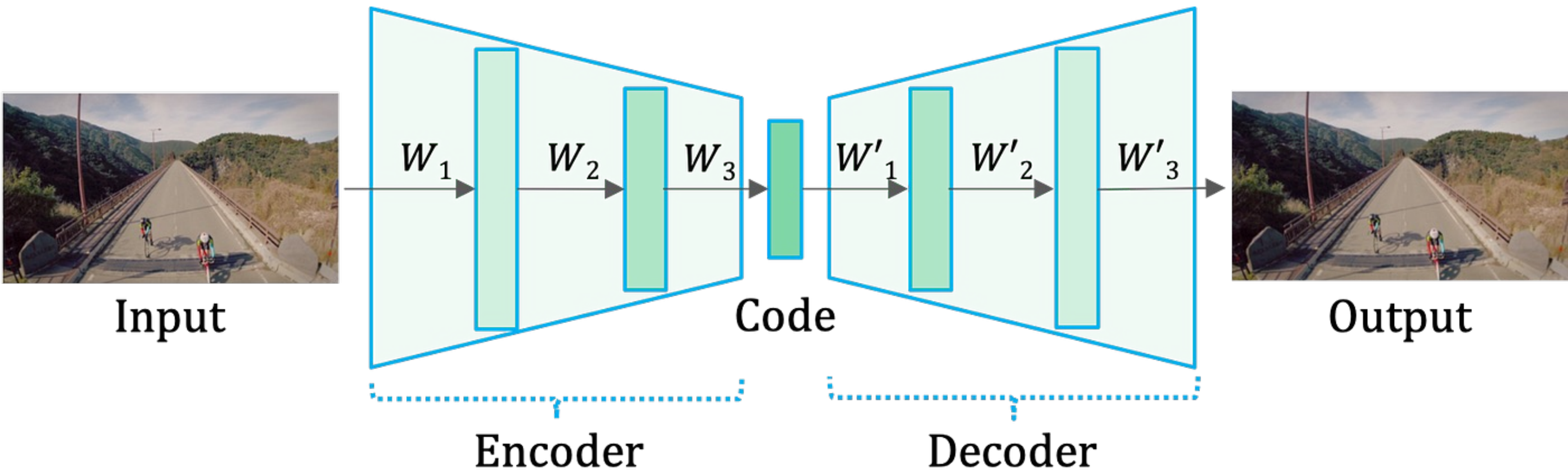
ML Based Compression

- Utilize layers of artificial neurons to process data in complex patterns, ideal for capturing nonlinear dependencies in data.



ML Based Compression

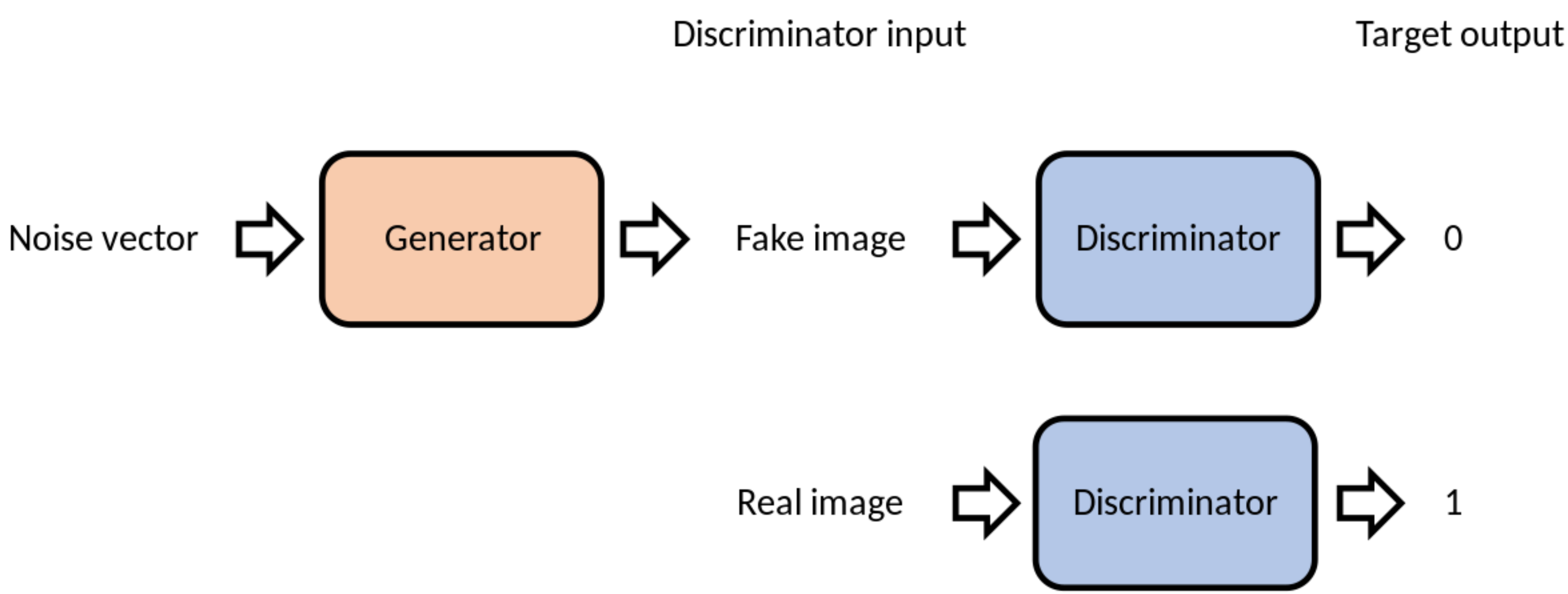
- Auto Encoder
  - Compresses input into a lower-dimensional code and then reconstructs the output from this code



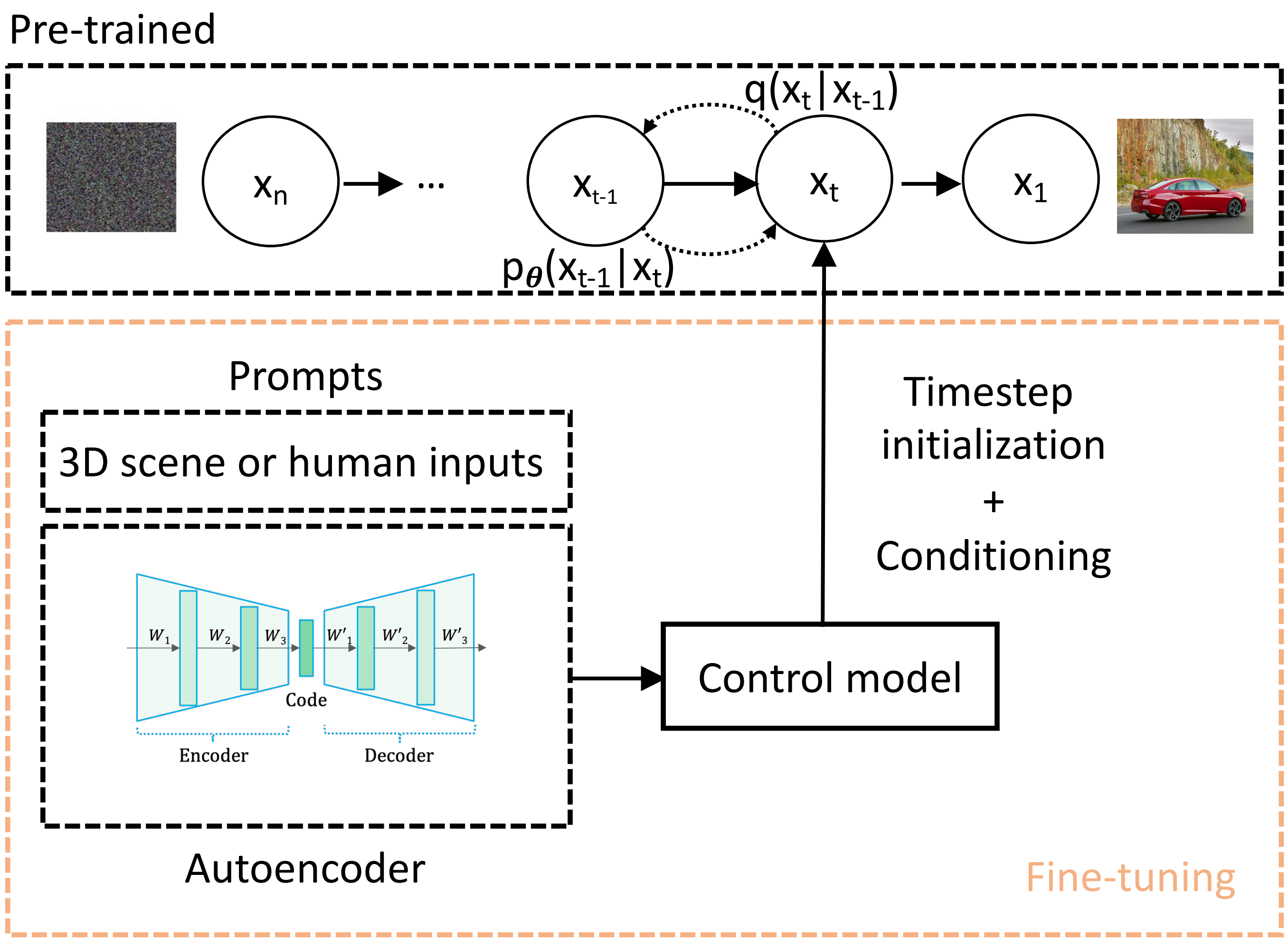
Weights & Latent code vector – the internal logic can be much more complex

ML Based Compression

- GANs (Generative adversarial networks)
  - Consist of two neural networks, the generator and the discriminator, competing against each other to generate data very similar to the original data, useful for high-fidelity compression.



Diffusion Model Based Compression



Transformer Based Compression

- Computational Attention based
  - Computes ‘soft’ weights that change during run time
  - Attends more towards certain weights i.e., gives more importance to certain regions

Visual Attention

- Semantic or salient features

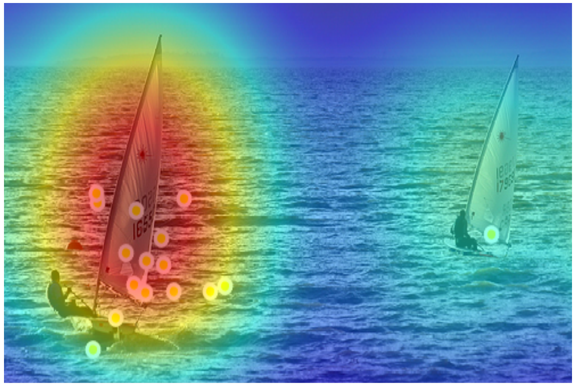
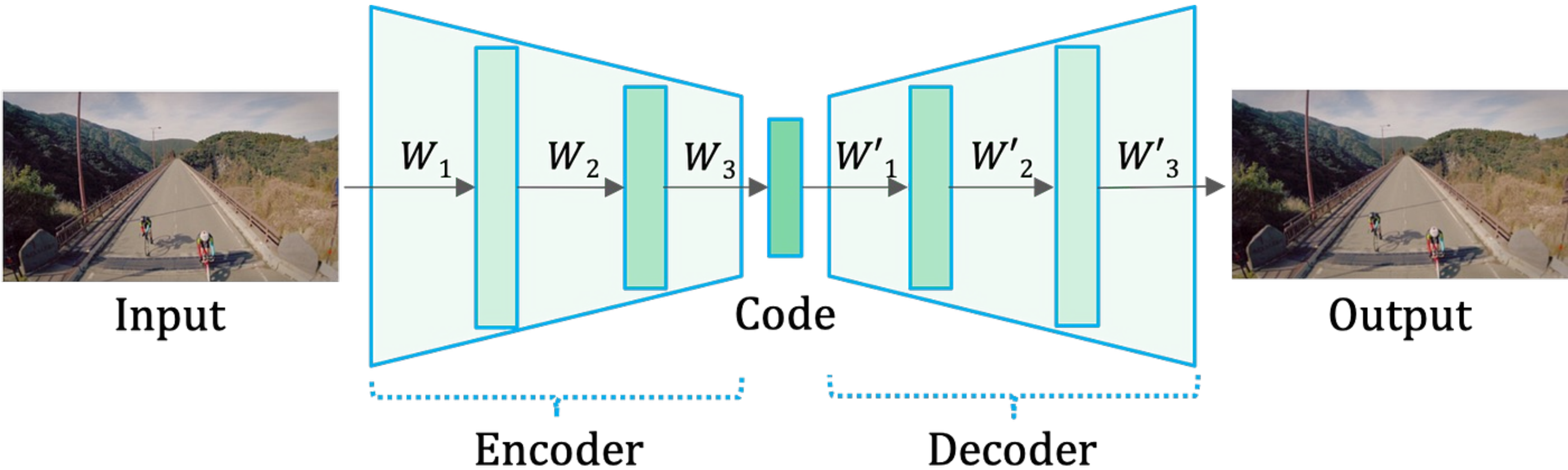
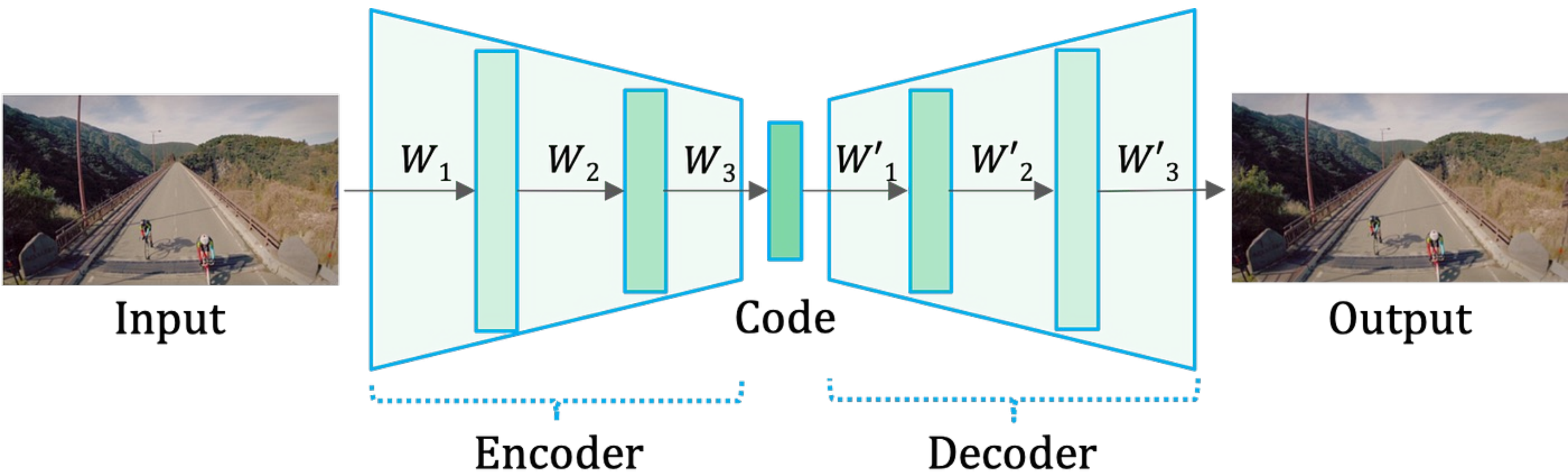


Image Compression



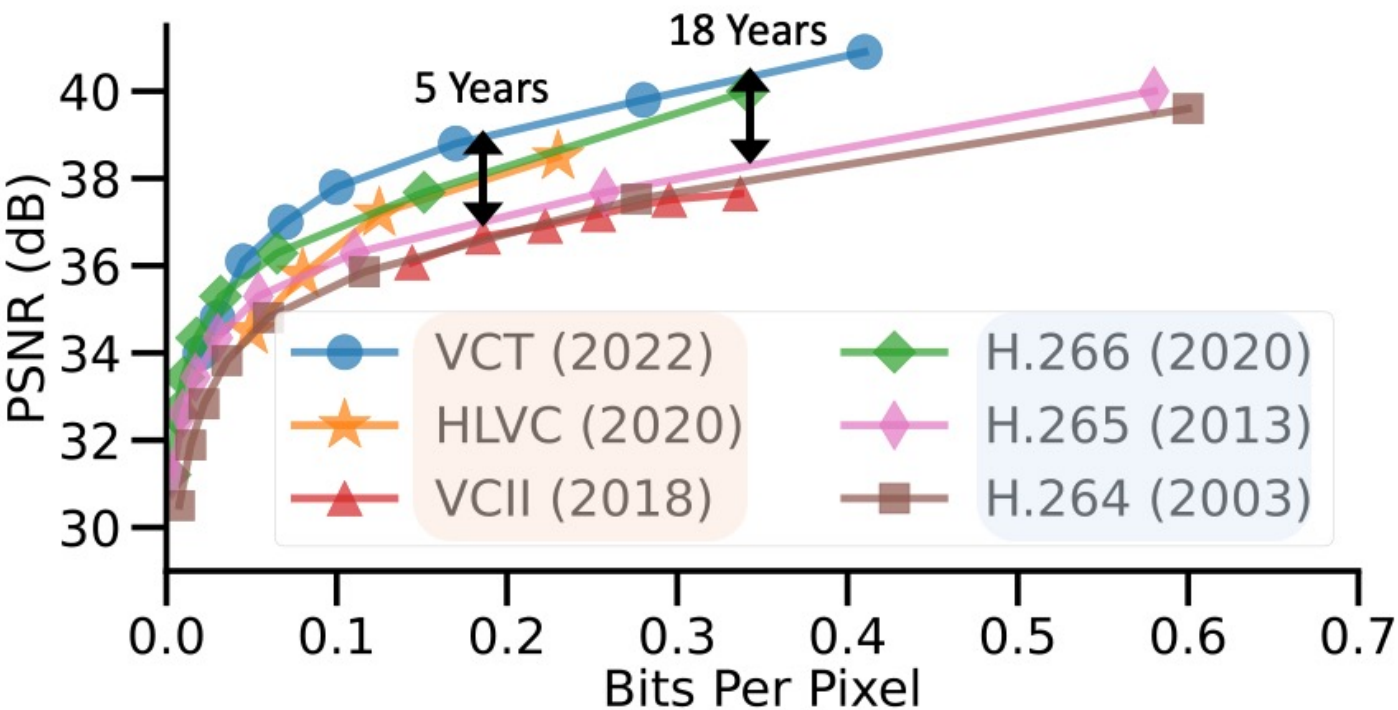
Spatial redundancy – Convolutional neural networks (CNNs)

Video Compression



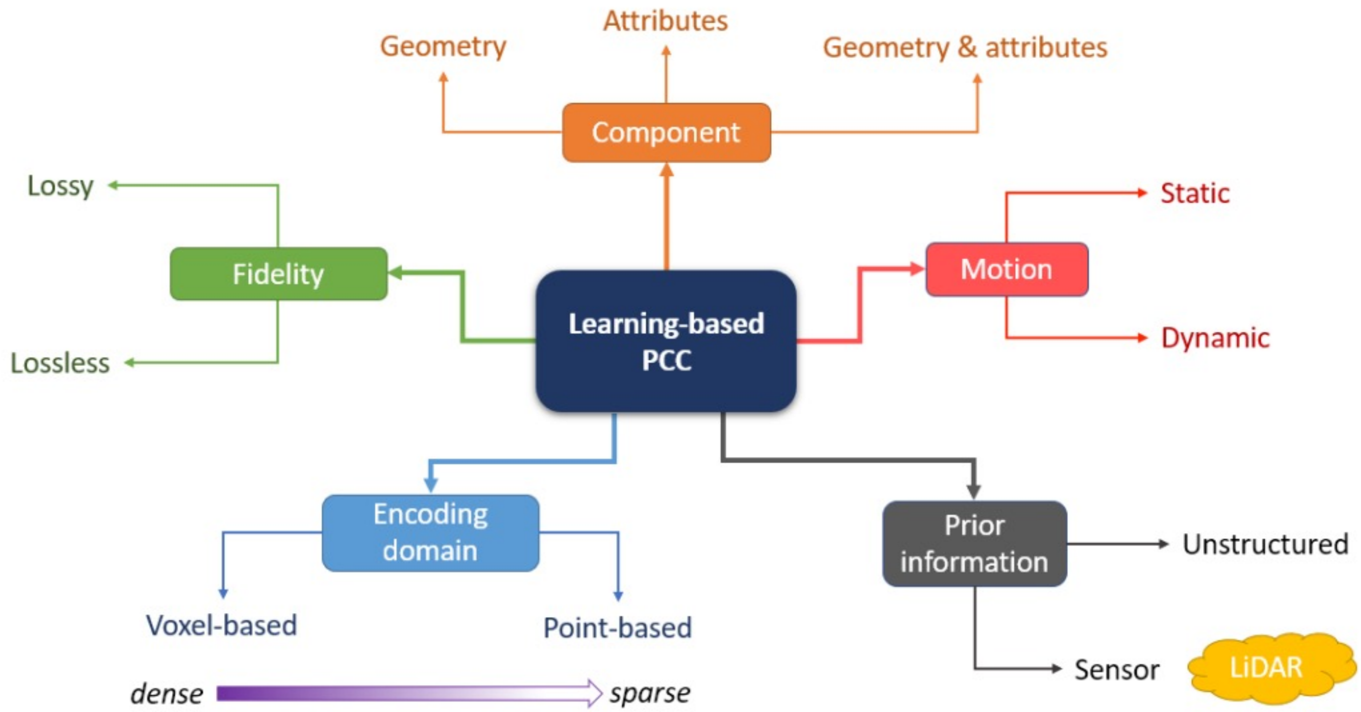
Spatial & Temporal redundancy – 3D CNNs or LSTMs, need to estimate residuals

Evolution of Video Codecs

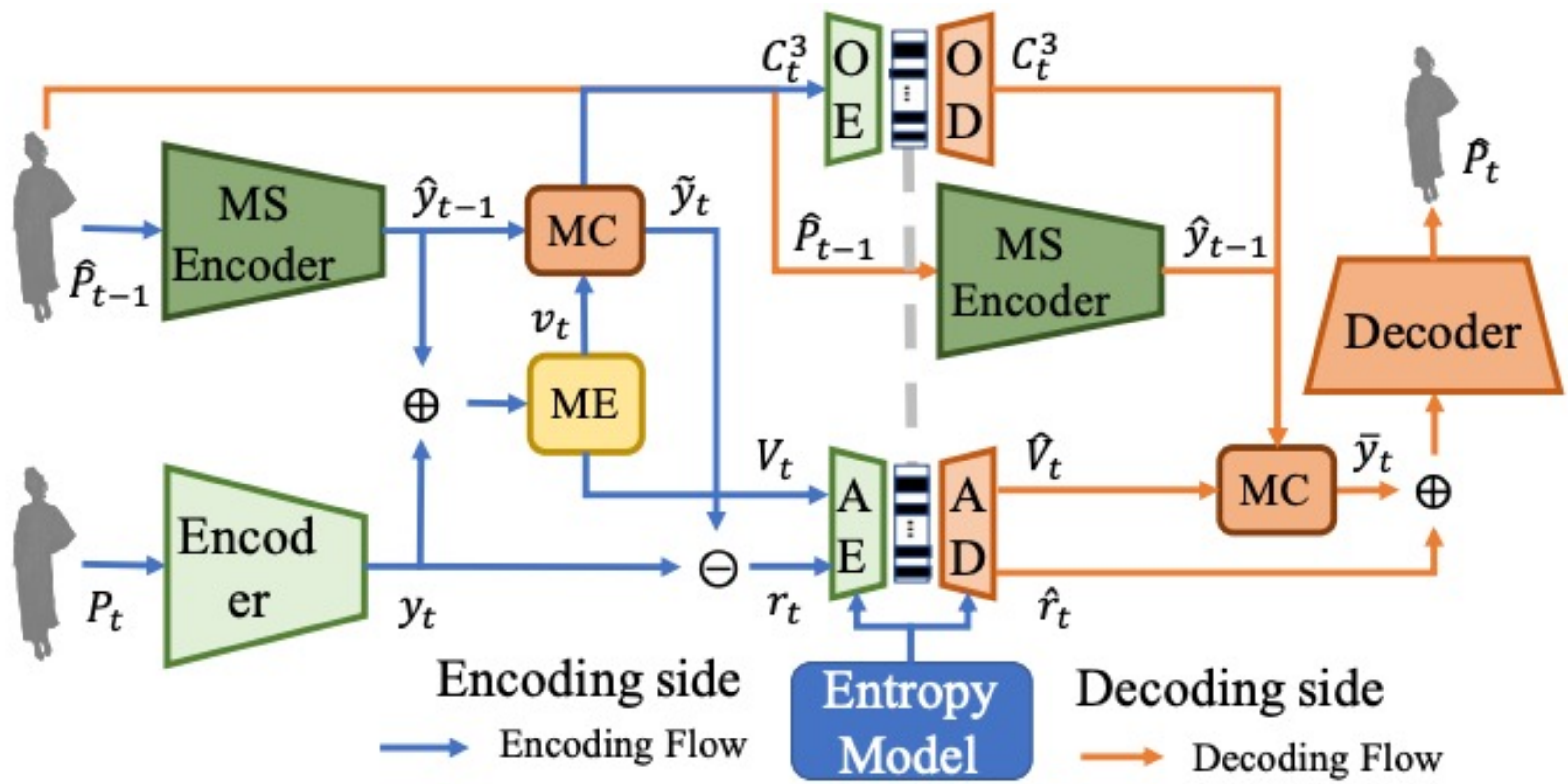


neural and classical video codecs showing compression efficiency across generations.

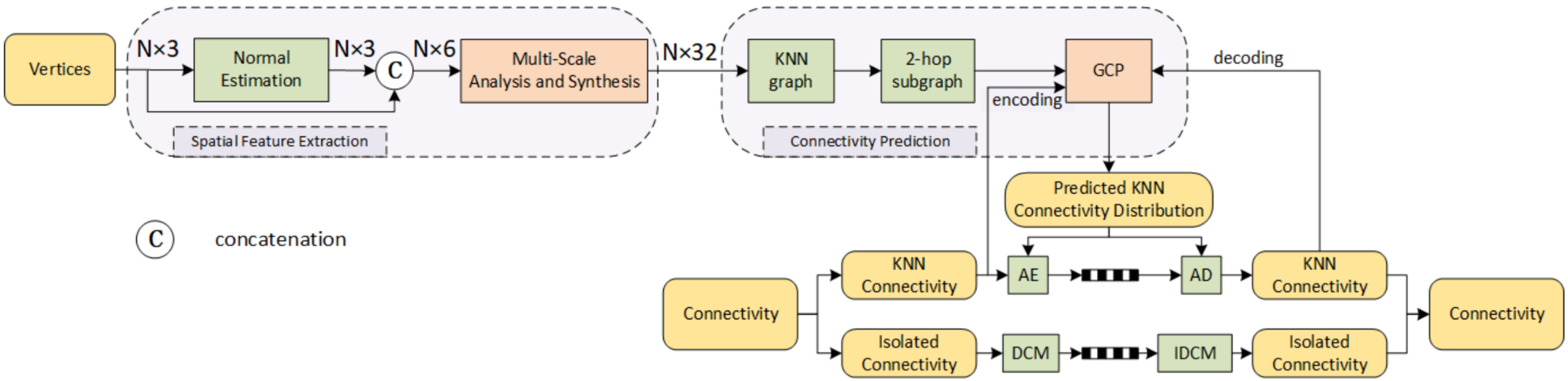
Point Cloud Compression



Point Cloud Compression

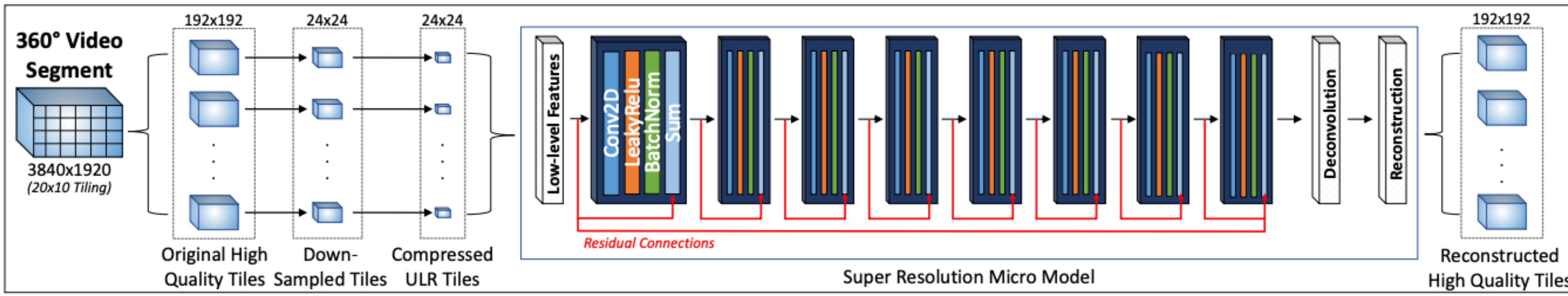


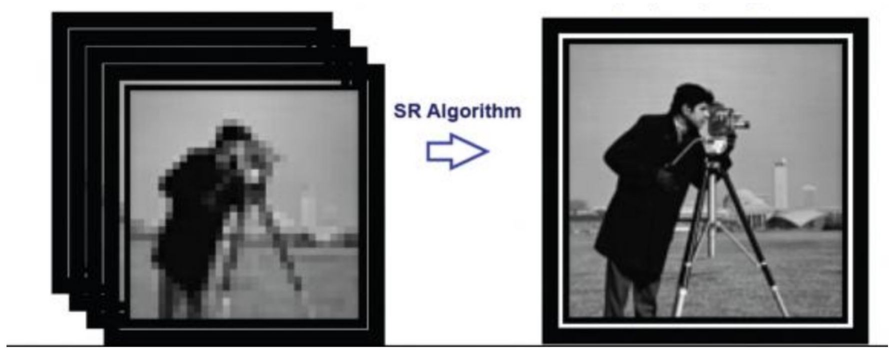
Mesh compression - Connectivity



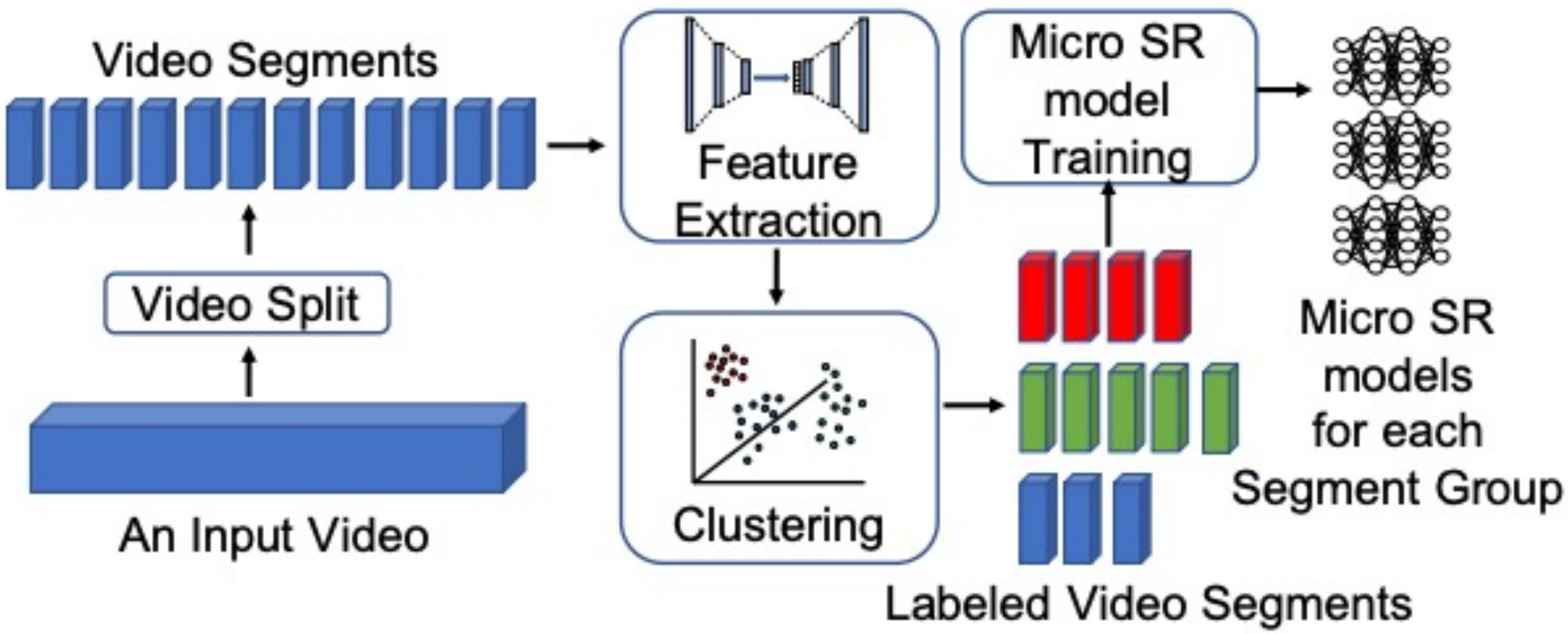
Vertex prediction & Connectivity prediction

Super Resolution of Low Res content to High Res





Super Resolution of Low Res content to High Res



Super Resolution of Low Res content to High Res

- Can be applied on traditional compression settings as well
  - E.g., Compress excessively using traditional codec, and use super resolution to enhance the quality after decoding

Performance Metrics

- Quality
  - PSNR
  - SSIM
  - VMAF - Netflix
- Compression ratio
- Latency
- Power consumption

Type of Codecs

- Generalized model
  - Train on a large-scale dataset – as much data as possible
  - Complex model
- Category-specific model
  - Train on a particular class of dataset e.g., sports or Netflix database
- Video-specific model
  - Model specific to video – memorize the conent

Limitations

- Difficult to generalize
  - There is never enough data to train a model
  - We can circumvent this problem in certain scenarios (e.g., when streaming on-demand stored content like Netflix or YouTube)
- Not many devices have GPUs in practice
- High Power consumption

Summary of the Lecture

- Limitations of traditional algorithms
- Advances in ML based compression
- Auto encoders, GANs, Transformers, Attention, Diffusion Models
- Super Resolution
- Performance metrics

Next up: multi-view compression