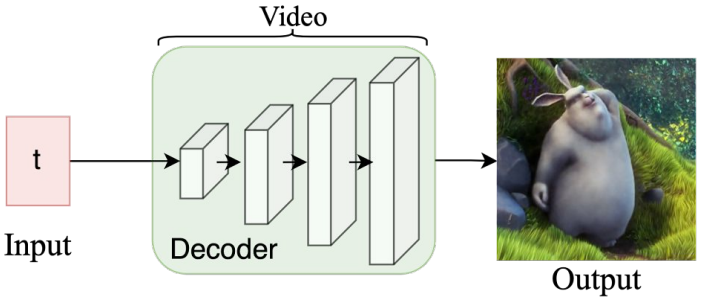
HNeRV: A Hybrid Neural Representation for Videos

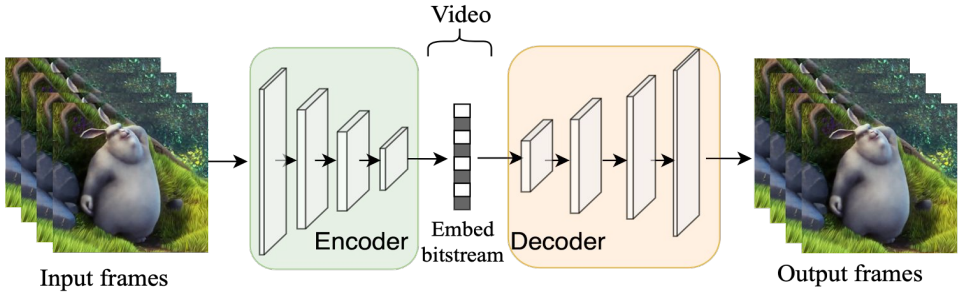
Hao Chen, Matthew Gwilliam, Ser-Nam Lim, Abhinav Shrivastava

WED-AM-195

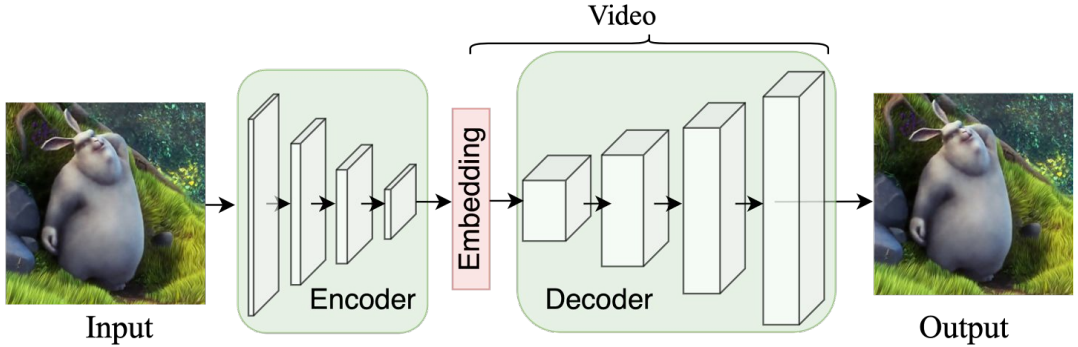
HNeRV: Architecture Preview



Implicit neural representation

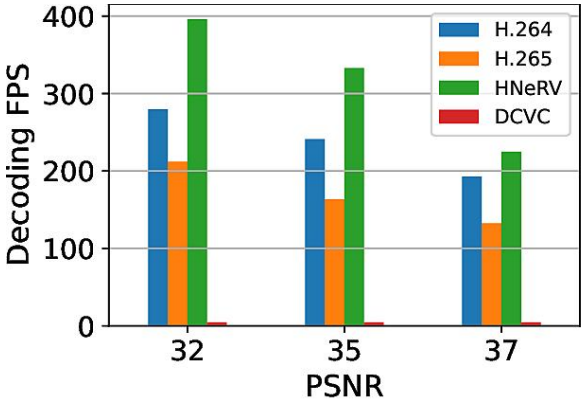


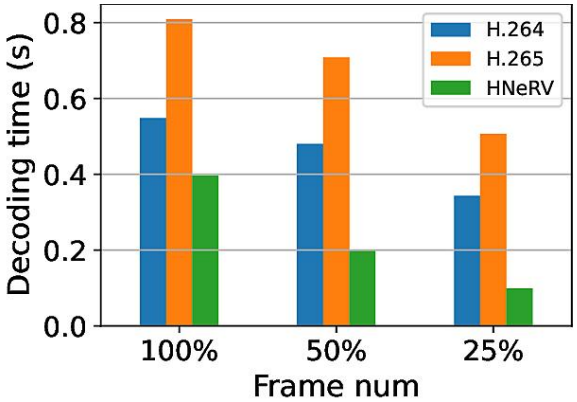
Explicit video representation

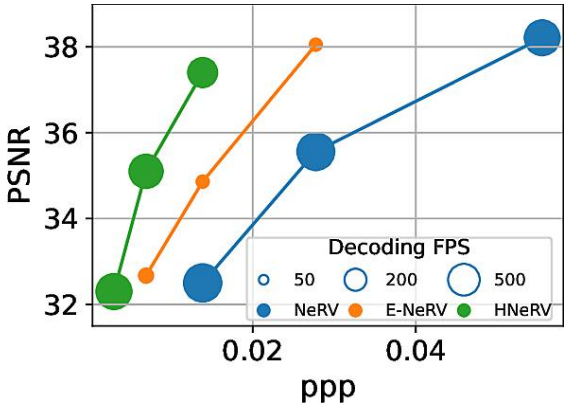


Hybrid neural representation

HNeRV: Results Preview

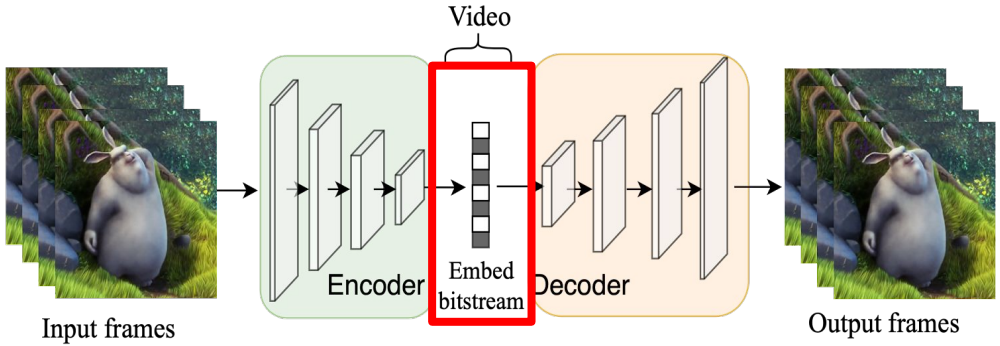


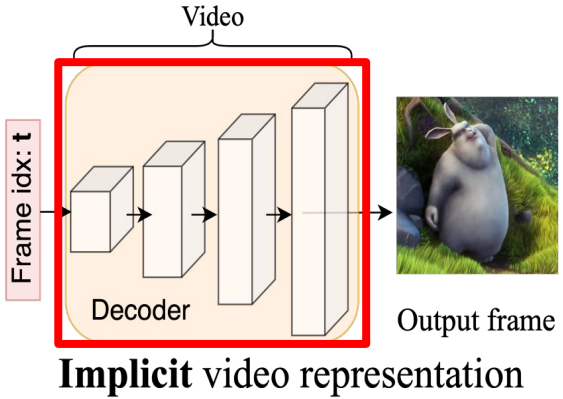




Background: Explicit vs Implicit Video Representation

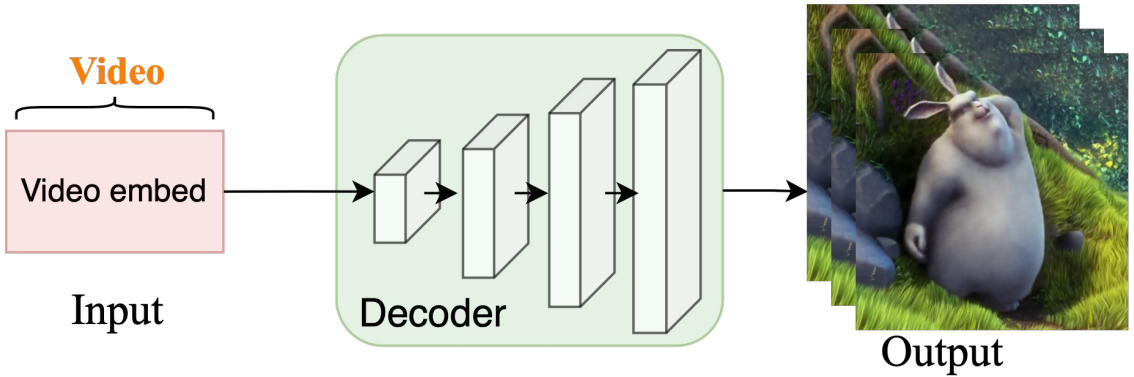
- Video = embed + decoder
- **Explicit** representation: video content = **embed**
- **Implicit** representation: video content = **decoder**





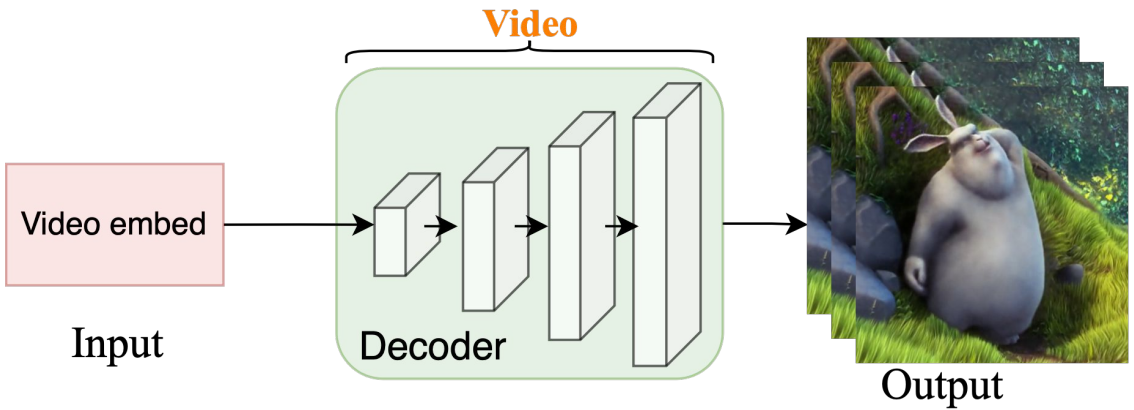
Background: **Explicit** Video Representation

- Video = **embed** + decoder
- Video content = embed
  - Video content stored *explicitly* in video embed
  - Decoder *shared* by all videos (video-agnostic)



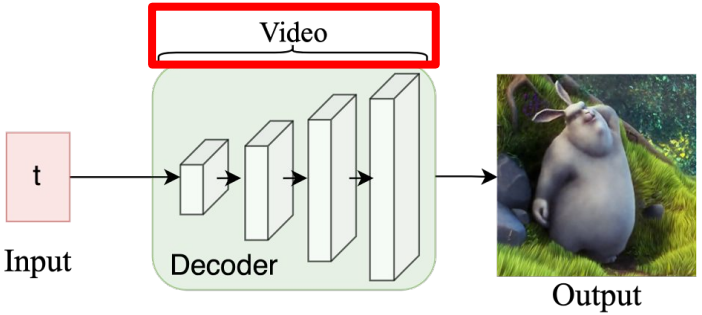
Background: **Implicit** Video Representation

- Video = embed + **decoder**
- Video content = decoder
  - Video content stored *implicitly* in the decoder
  - Input: frame index t
  - Video-specific decoder
  - One video = one decoder



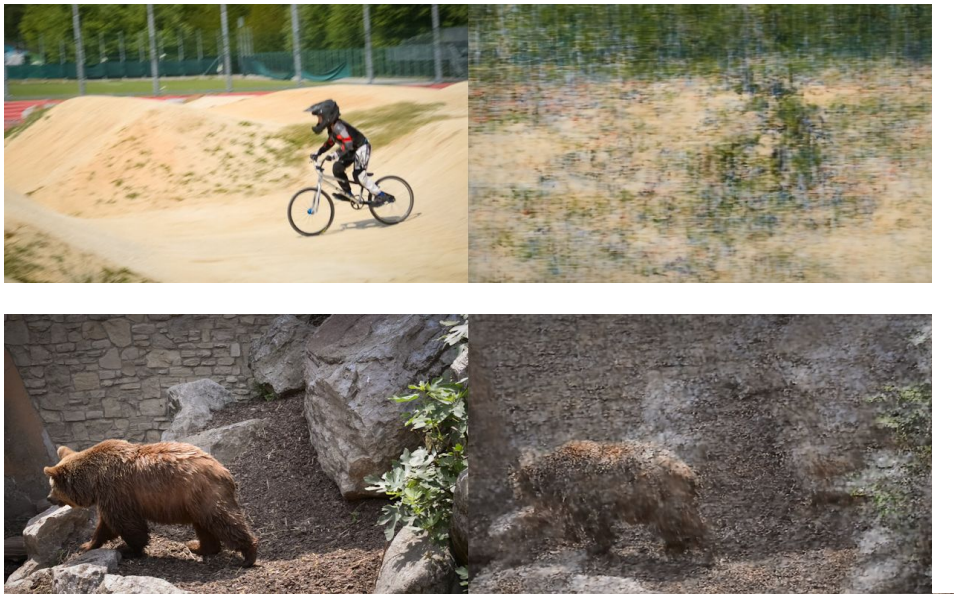
NeRV Limitation: Efficiency

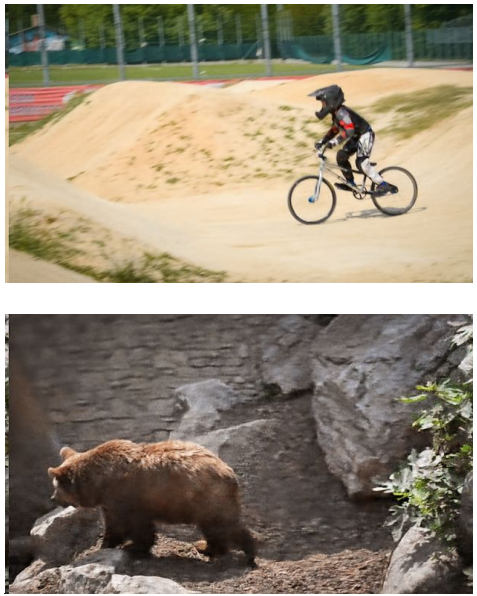
- **Implicit** neural representation for videos (NeRV)
  - *Content-agnostic* input: frame index t
  - A video = the decoder neural network
- How to make a more efficient representation?
  - Better reconstruction
  - Faster convergence



Implicit neural representation

NeRV Limitation: Video Interpolation





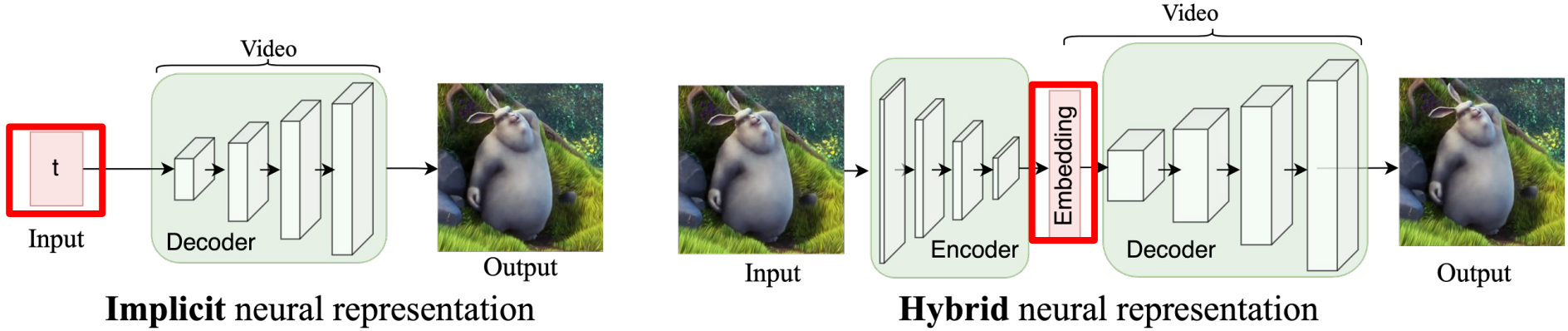
Frame T

Frame T+0.5
(NeRV)

Frame T+1

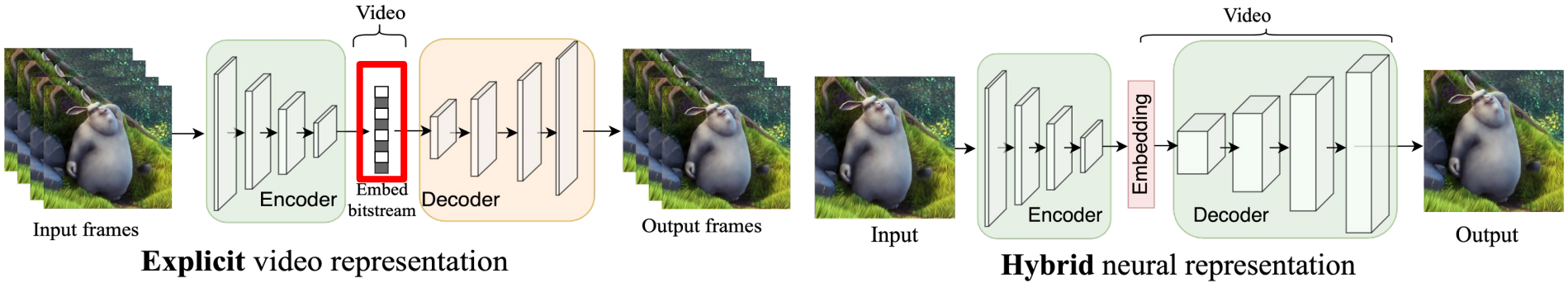
HNeRV: Hybrid Neural Representation

- **Hybrid** neural representation for videos (HNeRV)
  - Learnable, content-adaptive embedding
  - A video = *tiny* frame embed + a decoder

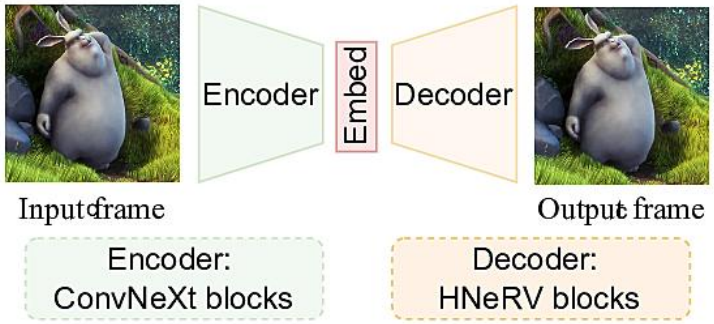


Explicit Representations vs. HNeRV

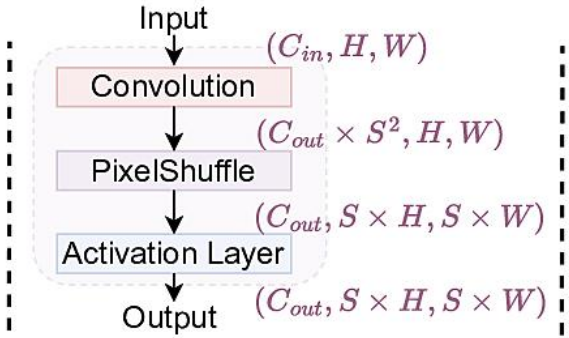
- Auto-encoder
  - universal vs **video-specific**
- Frame embed
  - huge vs **tiny**



HNeRV Architecture



a) HNeRV architecture

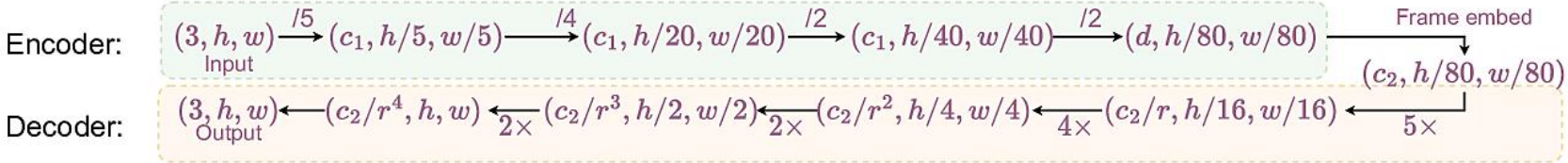


b) HNeRV block

Block parameters =
$K^2 \times C_{\text{in}} \times C_{\text{out}} \times S^2$

K: Conv kernel size
C_{in}: Input channel
C_{out}: Output channel
S: Upscale size
H, W: Spatial size

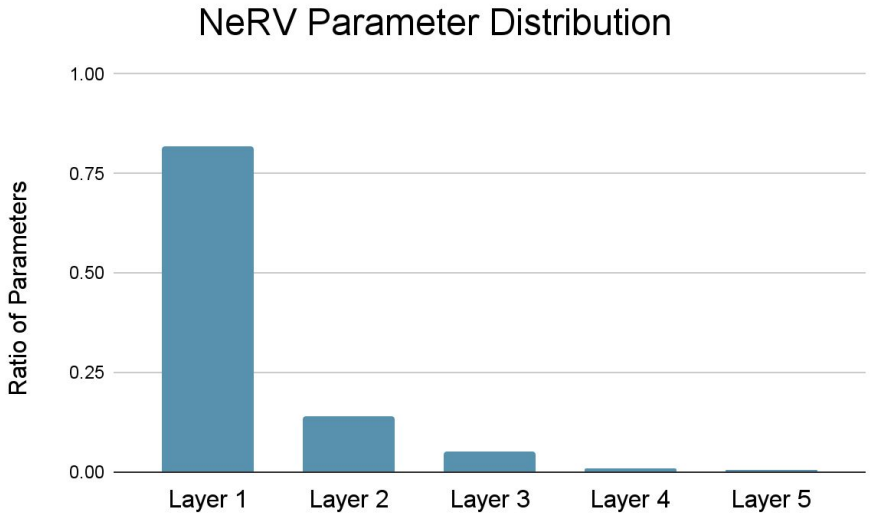
c) block parameters



d) output size of HNeRV stages

HNeRV: Hybrid Neural Representation

- NeRV: Uneven-distributed model parameters
  - Parameters decrease dramatically for higher layers



NeRV blocks	$k = 3, \quad C_{\text{out}} = C_{\text{in}}/2$

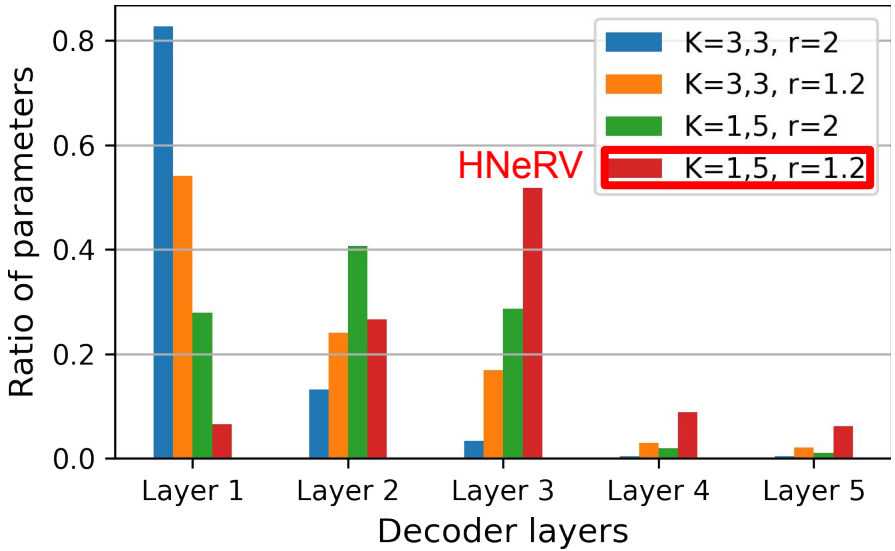
HNeRV: Hybrid Neural Representation

- NeRV: Uneven-distributed model parameters
  - Parameters decrease dramatically for higher layers
- HNeRV: **Even-distributed** model parameters

NeRV blocks	$k = 3, \quad C_{\text{out}} = C_{\text{in}}/2$
HNeRV blocks	$k = 1, 3, ..., K_{\text{max}}, \quad C_{\text{out}} = C_{\text{in}}/r$

HNeRV: Hybrid Neural Representation

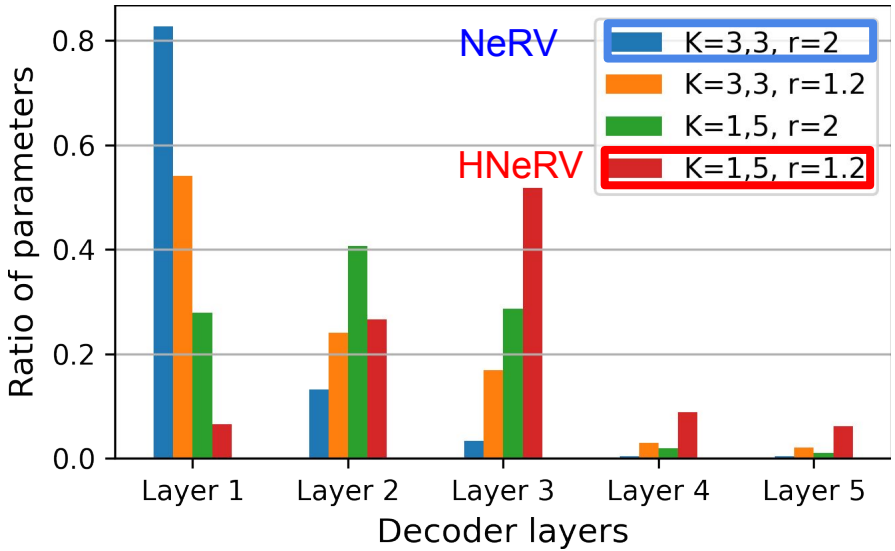
- NeRV: Uneven-distributed model parameters
  - Parameters decrease dramatically for higher layers
- HNeRV: **Even-distributed** model parameters



NeRV blocks	$k = 3, \quad C_{\text{out}} = C_{\text{in}}/2$
HNeRV blocks	$k = 1, 3, ..., K_{\text{max}}, \quad C_{\text{out}} = C_{\text{in}}/r$

HNeRV: Hybrid Neural Representation

- NeRV: Uneven-distributed model parameters
  - Parameters decrease dramatically for higher layers
- HNeRV: **Even-distributed** model parameters

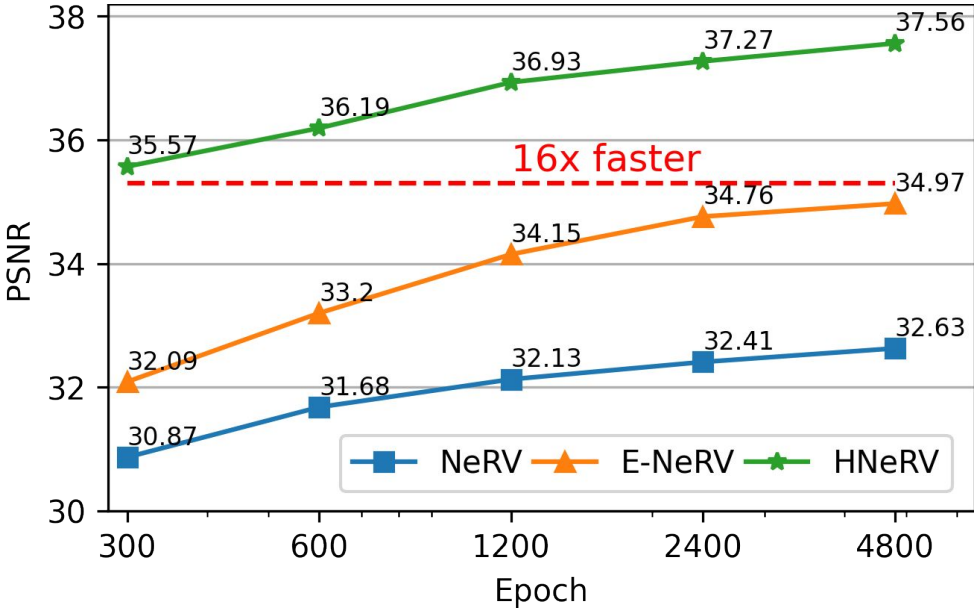


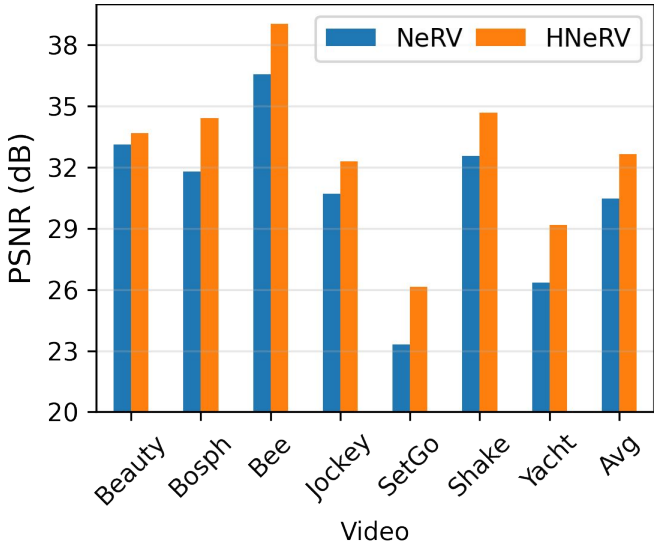
NeRV blocks	$k = 3, \quad C_{\text{out}} = C_{\text{in}}/2$
HNeRV blocks	$k = 1, 3, ..., K_{\text{max}}, \quad C_{\text{out}} = C_{\text{in}}/r$

		NeRV		HNeRV	
K	r	PSNR	MS-SSIM	PSNR	MS-SSIM
3,3	2	30.87	0.9341	29.91	0.9203
3,3	1.2	32.27	0.9496	33.09	0.9587
1,5	2	31.34	0.9399	34.32	0.9715
1,5	1.2	**33.03**	**0.9573**	**35.57**	**0.9773**

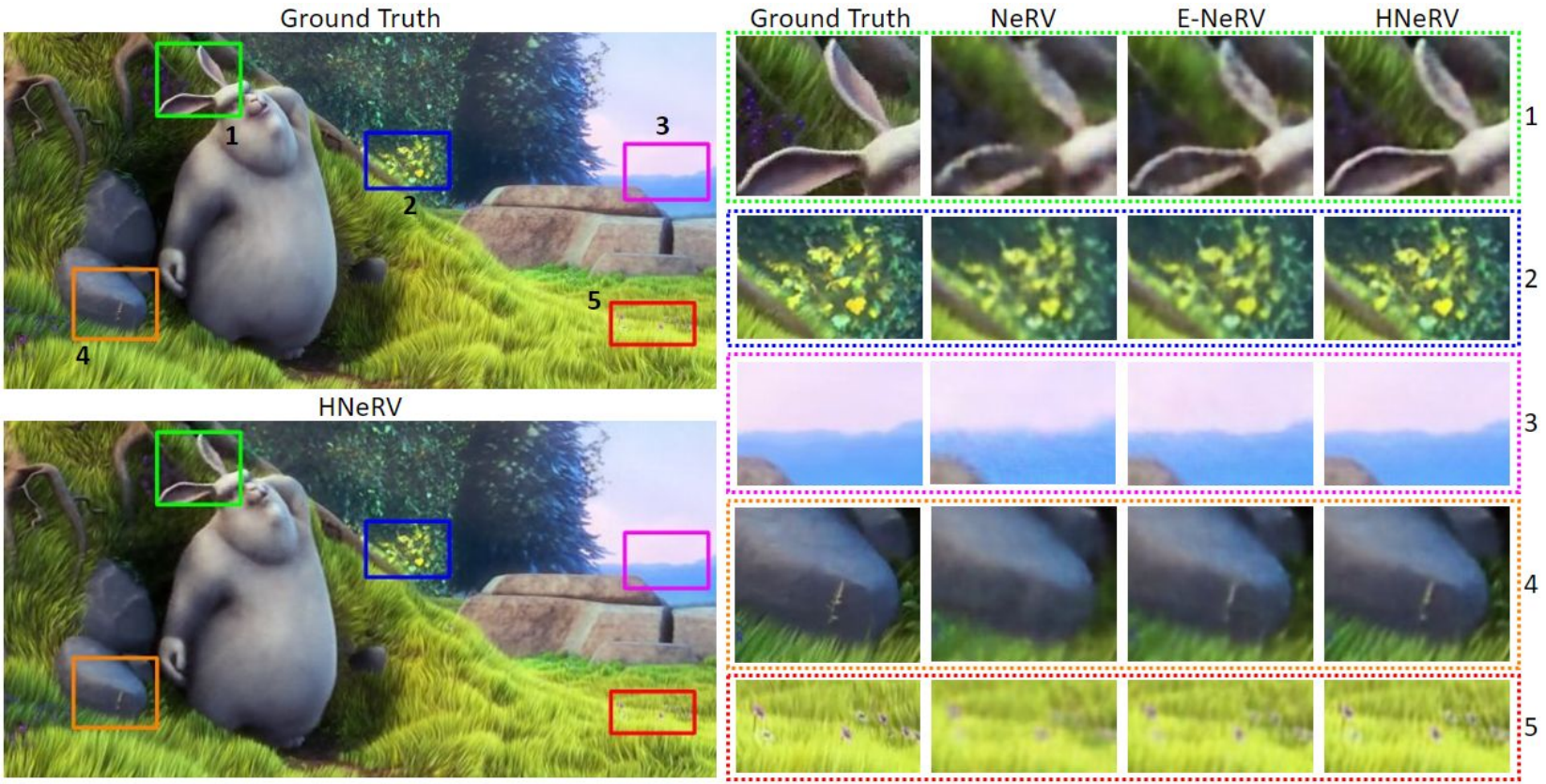
HNeRV: Hybrid Neural Representation

- Better reconstruction
- Faster convergence

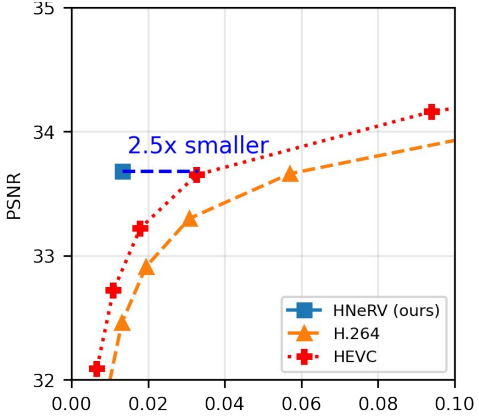


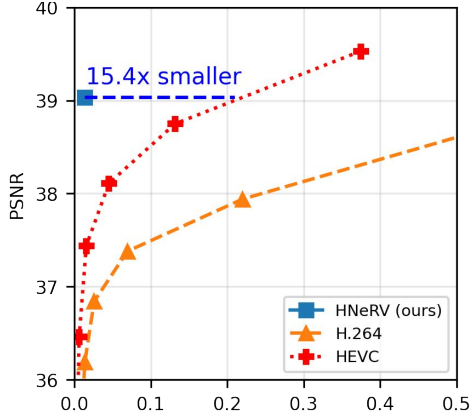


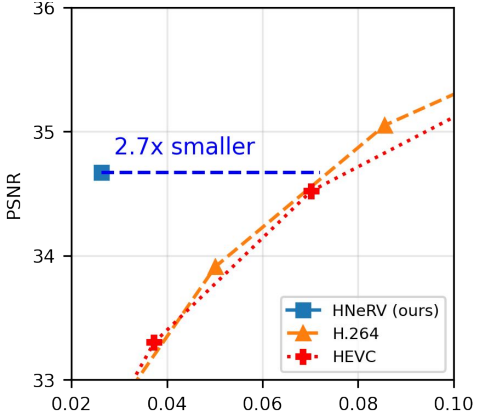
HNeRV: Hybrid Neural Representation

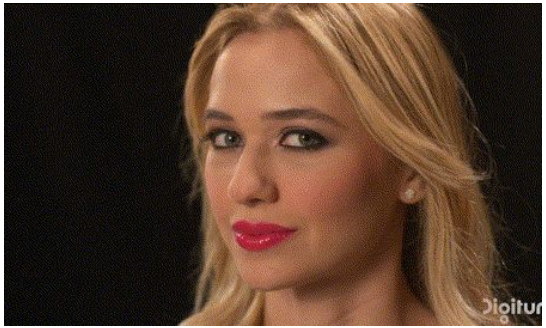


HNeRV: Videos with **Still** Background

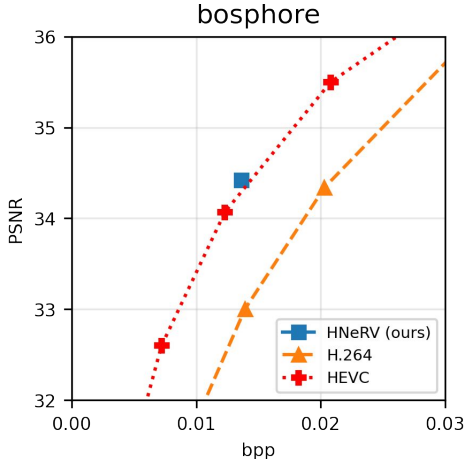


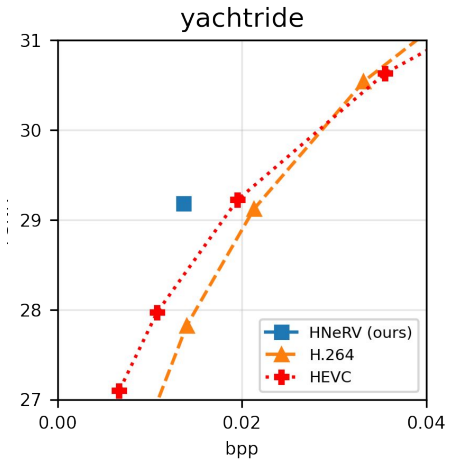




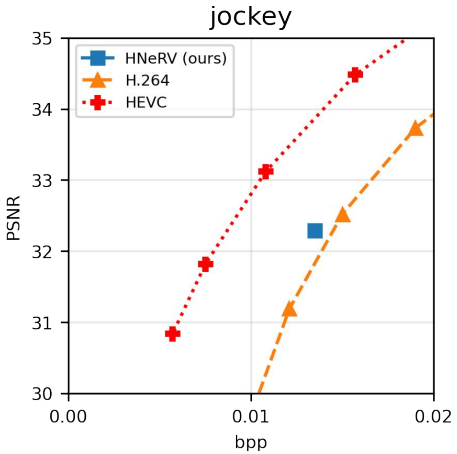


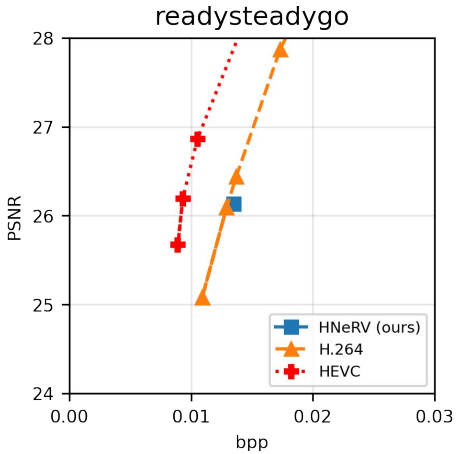
HNeRV: Videos of **Slow-moving** Background





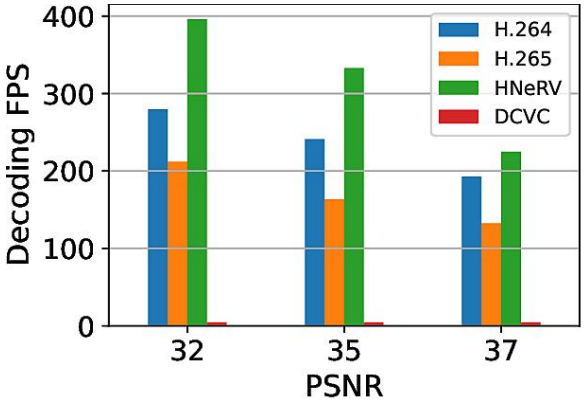
HNeRV: Videos of **Fast-moving** Backgrounds

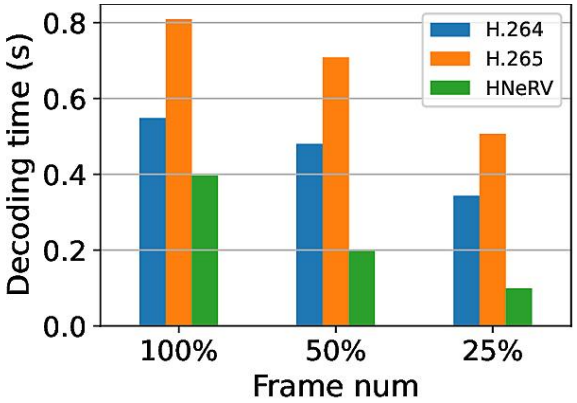


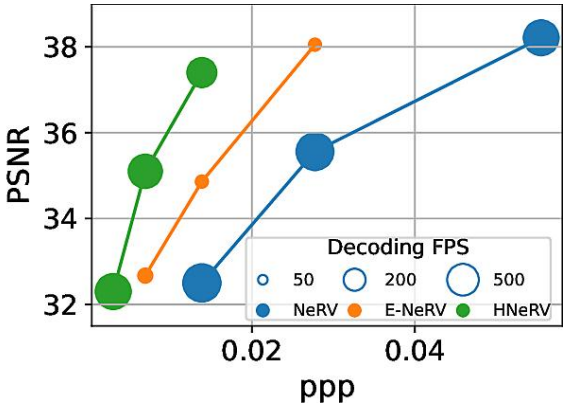


HNeRV: Efficient Decoding

- Smaller decoder since video-specific
- Faster than learning-based methods
- Easily deployed

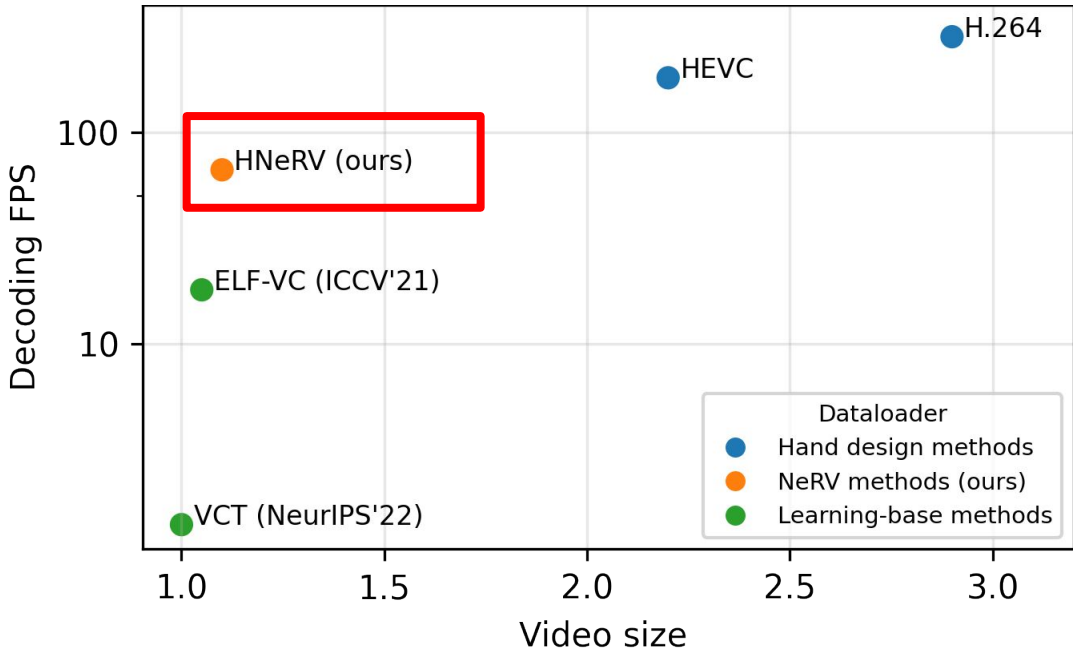






HNeRV: Efficient Video Representation

- Compression 😄
- Decoding speed 😄



HNeRV: Video **Interpolation** by Embedding Interpolation

Frame T

Frame T+0.5
(NeRV)

Frame T+0.5
(HNeRV)

Frame T+1

HNeRV: Video Inpainting

HNeRV input

HNeRV output

HNeRV: Video Restoration

HNeRV input

HNeRV output

HNeRV: Video Restoration

HNeRV input

HNeRV output