Deep Probabilistic Video Compression

Jun Han
with Salvator Lombardo, Christopher Schroers, Stephan Mandt
Ph.D. at Dartmouth College & UT Austin

November 15, 2018
Background

Traditional codecs
- H.264/H265; VP9

Main elements:
- Process I-frame using image encoder and decoder.
- Store block motion and residual information for P-frames.

Limitation: Blocky artifacts in low-bit rate regime.
Background

Traditional codecs

- H.264/H265; VP9

Main elements:

- Process I-frame using image encoder and decoder.
- Store block motion and residual information for P-frames.

Limitation: Blocky artifacts in low-bit rate regime.

H.265 (0.86 bpp)  
VP9 (0.57 bpp)  
Ours (0.06 bpp)

bpp: bit per pixel
Outline

- Variational Autoencoder
- Deep Image Compression
- Proposed Baseline Model
- Improved Video Compression Model
- Experiments
Variational Autoencoder

Let $q_\phi$: inference model; $p_\theta$: generative model; $p(z)$: prior,

$$
\mathcal{L}(\phi; \theta) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)] - D_{KL}(q_\phi(z \mid x) \parallel p(z)),
$$
Variational Autoencoder

Let $q_\phi$: inference model; $p_\theta$: generative model; $p(z)$: prior,

$$
\mathcal{L}(\phi; \theta) = \mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)] - D_{KL}(q_\phi(z \mid x) \mid\mid p(z)),
$$

**Problem for coding**

- Inference model $q_\phi$ is Gaussian and $z$ is continuous.
- $z$ should be discrete for coding (arithmetic coding; Huffman coding).
Figure: $\tilde{z}$: adding noise (training) or rounding (after training).
Deep Image Compression (Balle, ICLR 2017)

Figure: \( \tilde{z} \): adding noise (training) or rounding (after training). Prior \( p_\theta(\tilde{z}) \): parametric form to fit data distribution for entropy coding.
Deep Image Compression (Balle, ICLR 2017)

Figure: \(\tilde{z}\): adding noise (training) or rounding (after training). Prior \(p_\theta(\tilde{z})\): parametric form to fit data distribution for entropy coding.

- Inference model: \(\tilde{z} \sim q_\phi(z \mid x) = \mathcal{U}(z - \frac{1}{2}, z + \frac{1}{2})\)

- Loss: \(\mathcal{L}(\phi; \theta) = \mathbb{E}_{\tilde{z} \sim q}[\log p_\theta(x \mid \tilde{z})] - \beta \left( 0 - \mathbb{E}_{\tilde{z} \sim q}[\log p_\theta(\tilde{z})] \right)\)

where \(\beta\) adjusts rate-distortion ratio.
## Entropy Coding

### Arithmetic Coding

<table>
<thead>
<tr>
<th>symbols</th>
<th>probs</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.12</td>
</tr>
<tr>
<td>E</td>
<td>0.42</td>
</tr>
<tr>
<td>I</td>
<td>0.09</td>
</tr>
<tr>
<td>O</td>
<td>0.3</td>
</tr>
<tr>
<td>U</td>
<td>0.07</td>
</tr>
</tbody>
</table>

**IOU**: $0.37630 \leq C < 0.37819$.

**Binary**: 0.011000001 (9 bits).
Entropy Coding

Arithmetic Coding

<table>
<thead>
<tr>
<th>symbols</th>
<th>probs</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.12</td>
</tr>
<tr>
<td>E</td>
<td>0.42</td>
</tr>
<tr>
<td>I</td>
<td>0.09</td>
</tr>
<tr>
<td>O</td>
<td>0.3</td>
</tr>
<tr>
<td>U</td>
<td>0.07</td>
</tr>
</tbody>
</table>

Figure: Symbol IOU

**IOU:** \( 0.37630 \leq C < 0.37819 \).

**Binary:** 0.011000001 (9 bits).

- known probability models and length of symbols to decode
- steam based coding, not need to store coding table like Huffman.
Proposed Baseline Video Compression Model

Figure: Two choices of predictive models for entropy coding: $p_{\theta}(\tilde{z}_t \mid \tilde{z}_{t-1})$ Kalman Filter; $p_{\theta}(\tilde{z}_t \mid \tilde{z}_{<t})$ LSTM.
Figure: Global state $f$: per-segment, inferred from a segment $T$ of video by LSTM after encoder $\mu_\phi$; local state $z_t$: per-frame, inferred from $x_t$ after encoder $\mu_\phi$. 
Datasets:
- **Sprites**: characters from video game (64 × 64).
- **BAIR**: a robot pushing objects on a table (64 × 64).
- **Kinetics**: diverse set of human actions (downsampled and cropped to 64 × 64).
Experiments

Datasets:

- **Sprites**: characters from video game (64 × 64).
- **BAIR**: a robot pushing objects on a table (64 × 64).
- **Kinetics**: diverse set of human actions (downsampled and cropped to 64 × 64).

Results

- Verification: latent distribution and entropy visualizations
- Comparison with codecs
Latent Variable Distribution Visualization

Figure: Empirical distributions of the latent states from inference posteriors and ground truth prior model in one BAIR example.
Figure: Average bits of information stored in $f$ and $z_{1:T}$ for **LSTMP-L** (LSTM prior with only local states), **KFP-LG** (Kalman Filter prior with local and global states), and **LSTMP-LG** (LSTM prior with local and global states).
Quantitative Comparison

(a) Sprites  (b) BAIR  (c) Kinetics

Figure: Rate-distortion curves on three datasets measured in PSNR (higher corresponds to lower distortion). Solid lines correspond to our models.
Qualitative Comparison (BAIR)

Ours (0.29 bpp, PSNR=38.1)

VP9 (0.44 bpp, PSNR=25.7)

Figure: Compressed videos by our LSTMP-LG and VP9 in low-bit rate regime.
Qualitative Comparison (Kinetics)

Figure: Compressed videos by our LSTMP-LG and VP9 in low-bit rate regime.
Summary

- first end-to-end trainable video codec in VAE framework.
- global and local states with predictive model for efficient entropy coding.
- small rate in specialized content and competitive rate in generic content.
Summary

- first end-to-end trainable video codec in VAE framework.
- global and local states with predictive model for efficient entropy coding.
- small rate in specialized content and competitive rate in generic content.

Future Work

- Extend to full-resolution video (Conv LSTM for predictive models)
- Incorporate optical flow or motion information as side information

Thank You