Autoencoder based image compression: can the learning be quantization independent?

Thierry Dumas, Aline Roumy, Christine Guillemot

INRIA Rennes, France
Transform coding

- **transform**
  - KLT
    - if image pixels ~ Gaussian, optimal
    - not image independent
      - need to transmit KLT basis
  - DCT
    - if images ~ highly correlated GM process, almost optimal
    - image independent
      - no need to transmit DCT basis
Learning alternative transforms

- Discovering alternative transforms
- Suitable for image compression

Learning with rate-distortion optimization.

\[ \mathcal{D} + \gamma \mathcal{R}, \quad \gamma \in \mathbb{R}_+^* \]

Training encoding-decoding graph
Usually,

\[ \gamma_0 \rightarrow \{ \theta_0^*, \phi_0^* \} \]
\[ \gamma_1 \rightarrow \{ \theta_1^*, \phi_1^* \} \]
\[ \vdots \]
\[ \gamma_{n-1} \rightarrow \{ \theta_{n-1}^*, \phi_{n-1}^* \} \]

\[ \{ \theta_0^*, \phi_0^* \} \]
\[ \{ \theta_1^*, \phi_1^* \} \]
\[ \vdots \]
\[ \{ \theta_{n-1}^*, \phi_{n-1}^* \} \]

Training \[
\rightarrow n \text{ rates for compression}
\]

Test

- Varying the quantization at test time? \[\rightarrow\] Learning and storing one transform instead of \( n \).
- Learning jointly the quantization and the transform? \[\rightarrow\] Towards optimality?
I – Autoencoder for image compression

\[ \min_{\theta, \phi} \mathbb{E} \left[ \|X - g_d(Q(g_e(X; \theta)); \phi)\|^2_2 \right] + \gamma \mathbb{E} \left[ \sum_{i=1}^{m} \frac{1}{h \times w} \sum_{j=1}^{h \times w} \log_2 \left( \hat{p}_i(\hat{y}_{ij}) \right) \right] \]

where

- \( g_e(\cdot; \theta) \) is the encoder function,
- \( g_d(\cdot; \phi) \) is the decoder function,
- \( Q \) is the quantization function,
- \( X \) is the input image,
- \( \hat{X} \) is the reconstructed image,
- \( m \) is the number of channels,
- \( h \) and \( w \) are the height and width of the image,
- \( \mathbb{E} \) denotes the expected value,
- \( \mathbb{E} \) denotes the empirical entropy of the \( i^{th} \) feature map in \( Y \),
- \( \gamma \) is a hyperparameter,
- \( D \) is the data loss,
- \( R \) is the reconstruction loss.
I – Autoencoder for image compression

\[ Q'(u) = 0 \quad \Rightarrow \quad \theta \text{ cannot be learned} \quad \text{via gradient-based methods} \]

approximating \( Q \) at training time.
II – Two learnings

\( Q = \{Q_1, Q_2, \ldots, Q_m\}, \delta_i \) = quantization step size for \( Q_i, i \in [1, m] \).

• Learning jointly \( \{\theta, \phi, \delta_1, \ldots, \delta_m\} \):

\[
\min_{\theta, \phi, \delta_1, \ldots, \delta_m} \mathbb{E} \left[ \|X - g_d(g_e(X; \theta) + \Delta \odot T; \phi)\|_F^2 \right] \\
+ \gamma \mathbb{E} \left[ \sum_{i=1}^{m} \left( -\log_2(\delta_i) - \frac{1}{h \times w} \sum_{j=1}^{h \times w} \log_2 \left( \tilde{p}_i (y_{ij} + \delta_i \tau_{ij}) \right) \right) \right]
\]

• Learning \( \{\theta, \phi\} \) while fixing \( \{\delta_1, \ldots, \delta_m\} \).
III - Experiments

- **H.265**
  - learning $\{\theta, \phi\}$ while fixing $\{\delta_1, \ldots, \delta_m\}$; one $\{\theta, \phi\}$ per $\gamma$
  - still fixing $\{\delta_1, \ldots, \delta_m\}$ at test time

- **JPEG2000**
  - learning $\{\theta, \phi\}$ while fixing $\{\delta_1, \ldots, \delta_m\}$
  - varying $\{\delta_1, \ldots, \delta_m\}$ at test time
  - learning jointly $\{\theta, \phi, \delta_1, \ldots, \delta_m\}$
  - varying $\{\delta_1, \ldots, \delta_m\}$ at test time
III - Experiments

reference

H.265  X  X  O  O  JPEG2000

rate ≈ 0.23 bpp
IV - Interpretation

Zero mean each feature map of $Y$ \quad \rightarrow \quad \text{DCT-like distribution in each feature map of } Y

$f(x; \mu, \lambda) = \frac{1}{2\lambda} \exp\left(\frac{|x - \mu|}{\lambda}\right)$

$\mu \in \mathbb{R}$

$\lambda \in \mathbb{R}^*_+$

$50^{th}$ feature map in $Y$

$125^{th}$ feature map in $Y$
Was a DCT-like transform learned? No!

$$j^{\text{th}} \alpha \in \mathbb{R} \xrightarrow{g_d(\cdot ; \phi^*)} \hat{X}$$

<table>
<thead>
<tr>
<th>$j, \alpha$</th>
<th>50, 8.0</th>
<th>50, -8.0</th>
<th>125, 20.0</th>
<th>125, -20.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>top-left patch of $\hat{X}$</td>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
</tbody>
</table>
Thanks you for your attention!

For further details,

www.irisa.fr/temics/demos/visualization_ae/visualizationAE.htm