UNIVERSITÉ DE GENÈVE

stochastic information processing

A multi-layer image representation using Regularized Residual Quantization: application to compression and denoising

Sohrab Ferdowsi, Slava Voloshynovskiy, Dimche Kostadinov

Department of Computer Science, University of Geneva, Switzerland
{Sohrab.Ferdowsi, svolos, Dimche.Kostadinov}@unige.ch
http://sip.unige.ch

Overview

Applications: Joint compression/denoising of domain-specific images.

Contribution: Making RQ capable of quantizing high dim. data with arbitrary layers using regularization

Introduction

- RQ idea: Quantizing successively the residuals from a previous layer of quantization
- Popular in signal processing during the 1980's and 1990's (image and video coding)
- Problem with RQ: It gets over-trained quickly after a couple of layers.
- Our solution: Generating the codewords randomly from the optimal distribution

Related work: from dictionary learning to VQ

General inverse-problem formulation:

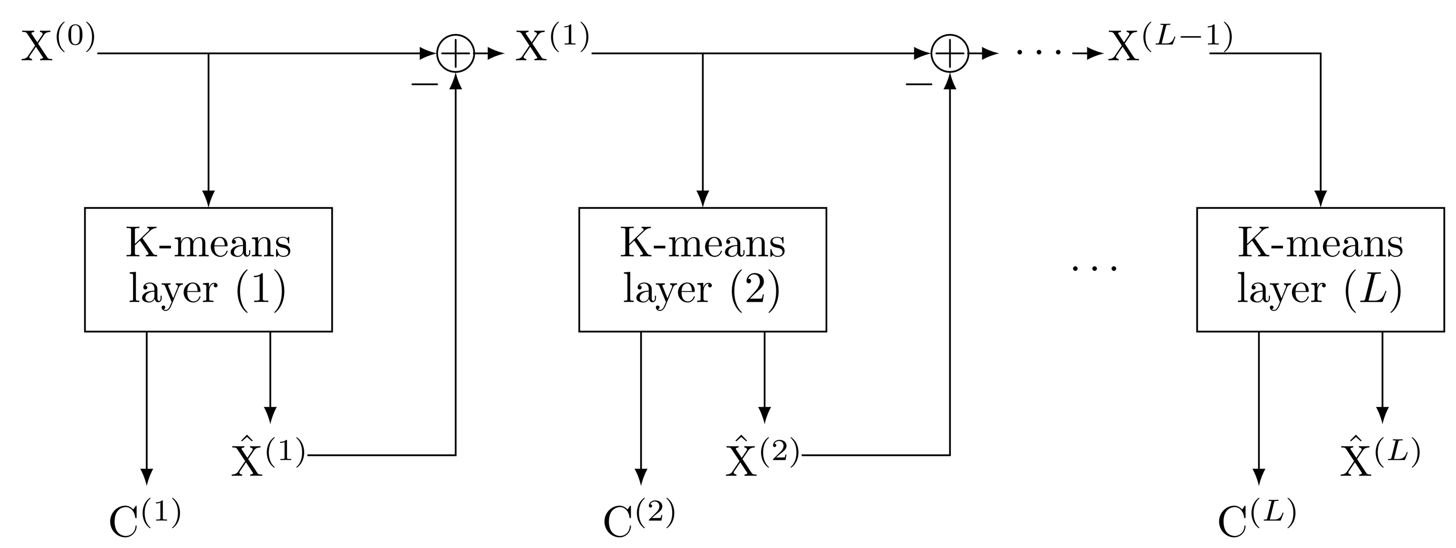
$$\begin{aligned} &\underset{C,A}{\text{minimize}} \ \|X - CA\|_F^2, \\ &\text{s.t.} \quad \Omega_C, \Omega_A \end{aligned} \tag{1}$$

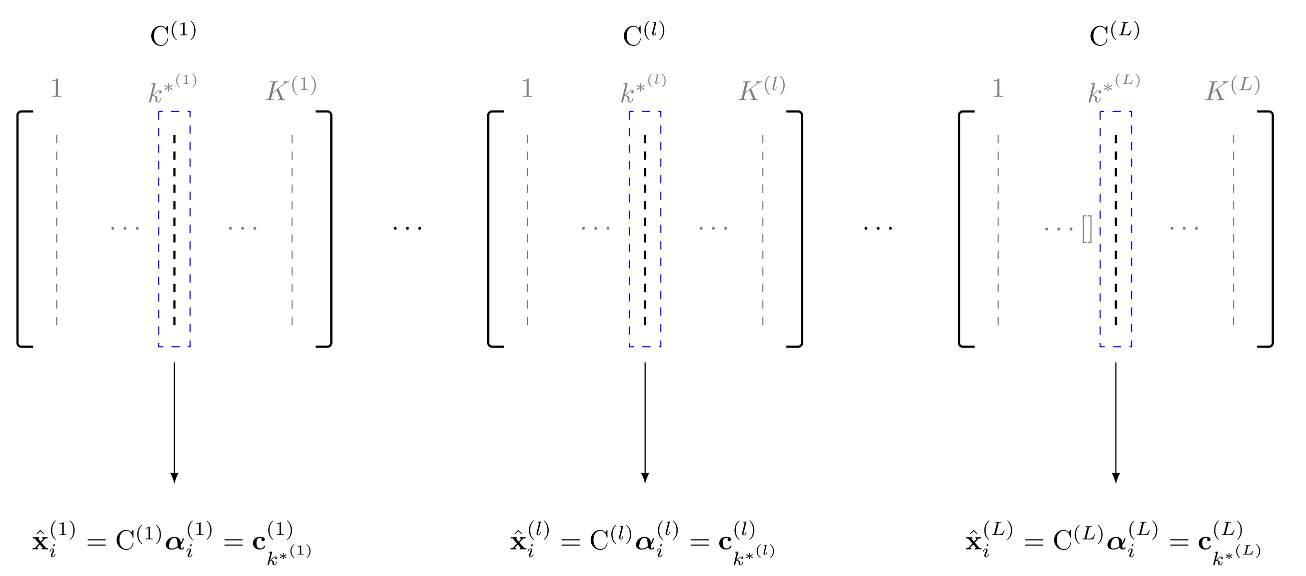
$X = [\mathbf{x}_1, \cdots, \mathbf{x}_i, \cdots \mathbf{x}_N]$: the training data samples
Ω_C: constraints on the construction of the codebook $C = [\mathbf{c}_1, \cdots, \mathbf{c}_k, \cdots, \mathbf{c}_K]$
Ω_A: constraints on the construction of the codes $A = [\boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_i, \cdots, \boldsymbol{\alpha}_N]$

- Sparse coding problem: $\Omega_A^{\ell_0}: \|\boldsymbol{\alpha}(i)\|_0 \leqslant s$ or $\Omega_A^{\ell_1}: \|\boldsymbol{\alpha}(i)\|_1 \leqslant s$
- K-means (basic VQ): $\Omega_A^{\mathsf{VQ}}: \|\boldsymbol{\alpha}_i\|_0 = \|\boldsymbol{\alpha}_i\|_1 = 1$.
- Product Quantization (PQ):

$$\Omega_A^{\mathsf{PQ}}: \|\boldsymbol{\alpha}_i^{(k')}\|_0 = \|\boldsymbol{\alpha}_i^{(k')}\|_1 = 1 \qquad \Omega_C^{\mathsf{PQ}}: C = \begin{bmatrix} C^{(1)} & 0 & \cdots & 0 \\ 0 & C^{(2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C^{(p)} \end{bmatrix}$$

- Residual Quantization (RQ): A multilayer structure on codebooks enforced during training:





$$\hat{\mathbf{x}}_i = \hat{\mathbf{x}}_i^{(1)} + \cdots + \hat{\mathbf{x}}_i^{(l)} + \cdots + \hat{\mathbf{x}}_i^{(L)}$$
$$k^* \triangleq \underset{1 \leqslant k \leqslant K^{(l)}}{\text{argmin}} \|\mathbf{x}(i)^{(l-1)} - \mathbf{c}_k^{(l)}\|_2$$

Preliminaries: Rate-Distortion behavior of n independent Gaussian sources

- Rate-Distortion theory: The trade-off between the compactness and the fidelity of representation
- A special setup: n independent Gaussian sources, $X_1, \cdots, X_j, \cdots, X_n$
- $X_j \sim \mathcal{N}(0, \sigma_j^2)$
- Expected distortion: $\mathcal{D} \triangleq \mathbb{E}[d(\mathbf{X}, \hat{\mathbf{X}})]$, where $d(\mathbf{a}, \mathbf{b}) \triangleq \frac{1}{n}\|\mathbf{a} - \mathbf{b}\|_2^2$
- Given a fixed D allowed ($\mathcal{D} \leqslant D$), what is the optimal way to allocate the distortion between these sources such that the overall allocated rate (distortion) is minimized?

$$\begin{aligned} &\min_{D_j} \sum_{j=1}^{n} \max\Big[0, \frac{1}{2}\log_2 \frac{\sigma_j^2}{D_j}\Big], \\ &\text{s.t.} \quad \sum_{j=1}^{n} D_j = D, \end{aligned} \tag{2}$$

- The solution is the **reverse water-filling**:

$$D_j = \begin{cases} \gamma, & \text{if } \sigma_j^2 \geqslant \gamma \\ \sigma_j^2, & \text{if } \sigma_j^2 < \gamma. \end{cases} \tag{3}$$

- Sources with variances less than γ should not be assigned any rate at all!

- **Optimal assignment** of the codeword variances:

$$\sigma_{C_j}^2 = \left(\sigma_j^2 - \gamma\right)^+ = \begin{cases} \sigma_j^2 - \gamma, & \text{if } \sigma_j^2 \geqslant \gamma \\ 0, & \text{if } \sigma_j^2 < \gamma. \end{cases} \tag{4}$$

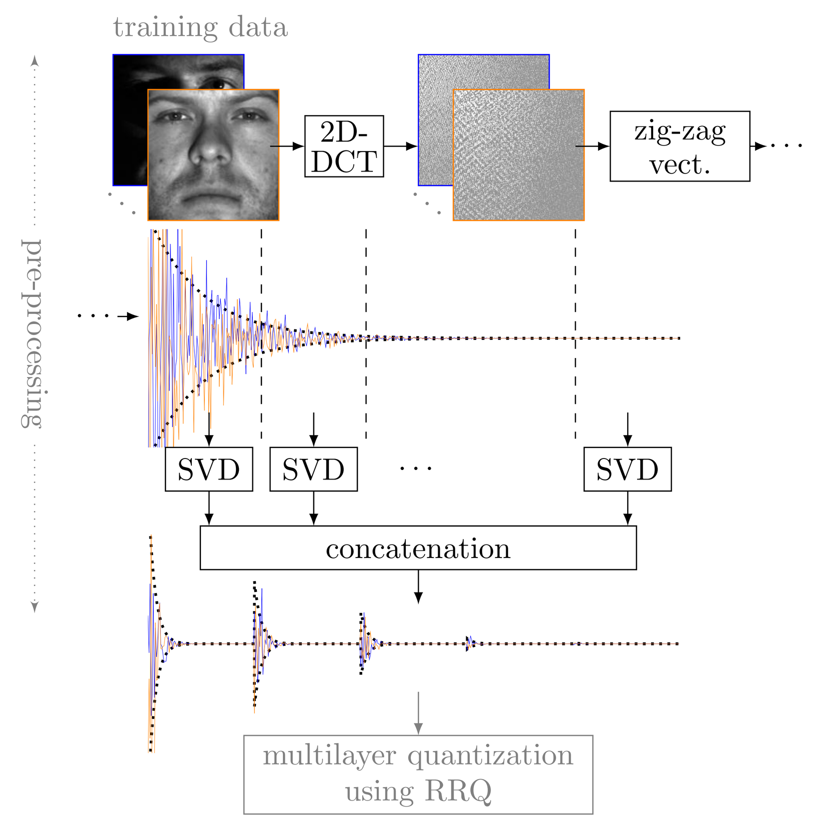
- Natural sparsification of codewords; to be enforced as regularization in learning

The proposed framework: RRQ

- K-means does not respect the optimal assignment of codeword variances.
- Instead, we generate codewords with the optimal distribution.

Pre-processing:

- To make the data un-correlated
- Be consistent between training and test sets



RRQ algorithm:

- Calculating the threshold level:
$\gamma^* = \underset{\gamma}{\text{argmin}}\Big(\big|\log_2 K^{(l)} - \sum_{j \in \mathcal{A}_\gamma} \frac{1}{2}\log_2 \frac{D_j^{(l-1)}}{\gamma}\big|\Big)$
- Soft-threshold the data variance at layer l (quantization distortion at layer $l-1$) using γ^*
$\sigma_{C_j^{(l)}}^2 = \left(D_j^{(l-1)} - \gamma^*\right)^+$
$S^{(l)} = \text{diag}(\sigma_{C_1^{(l)}}^2, \cdots, \sigma_{C_n^{(l)}}^2)$
- Generate $\mathbf{c}_k^{(l)}$'s randomly from $\mathcal{N}(\mathbf{0}, S^{(l)})$.
- Concatenate $\mathbf{c}_k^{(l)}$'s to get $C^{(l)}$.

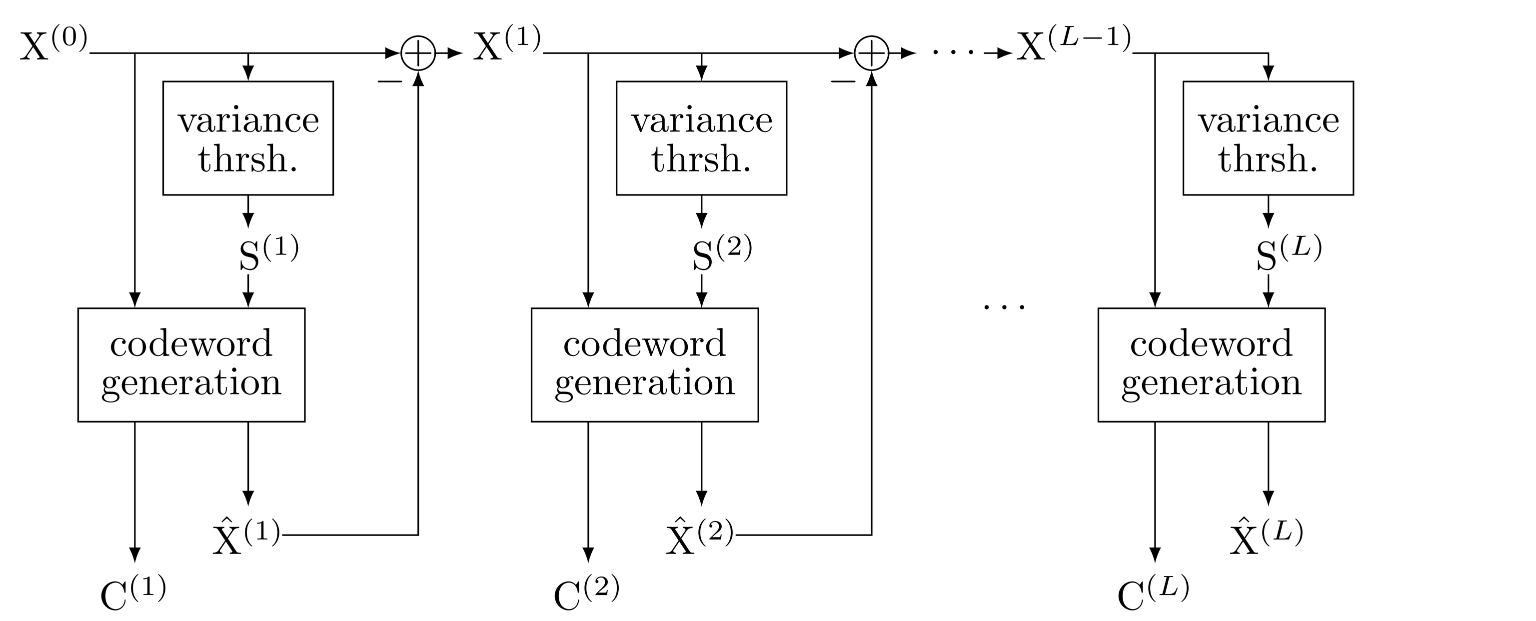
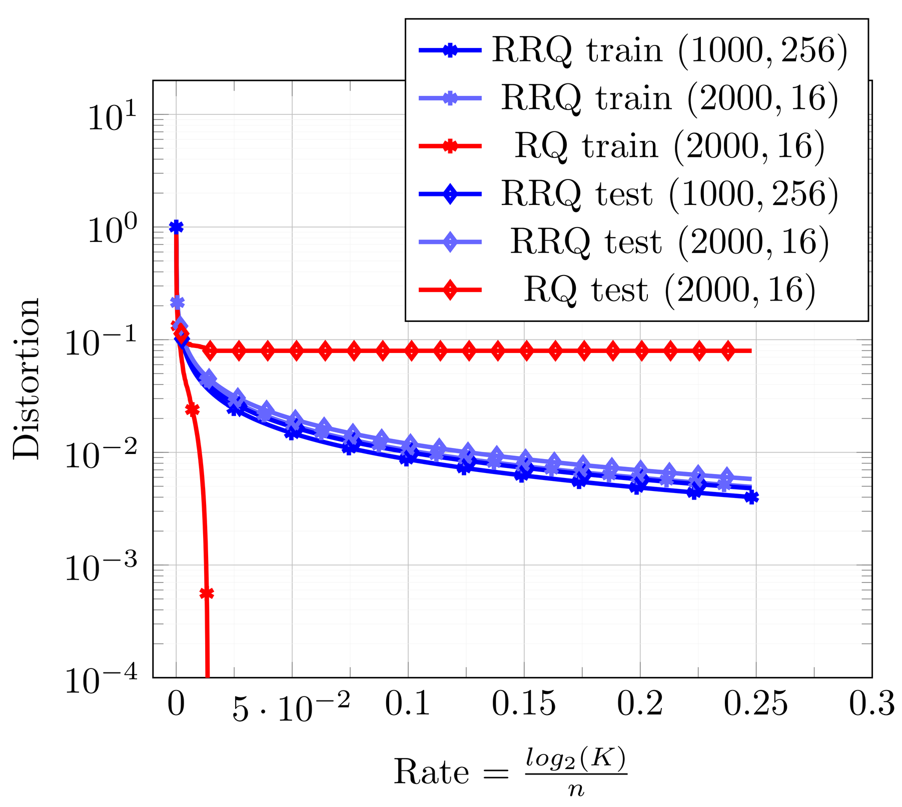


image compression

- The *CroppedYale-B* set is used which contains 2408 images of size 192×168 from 38 subjects.
- RQ over-trains at $n = 32256$, even from the first layer. RRQ outperforms JPEG-2000.



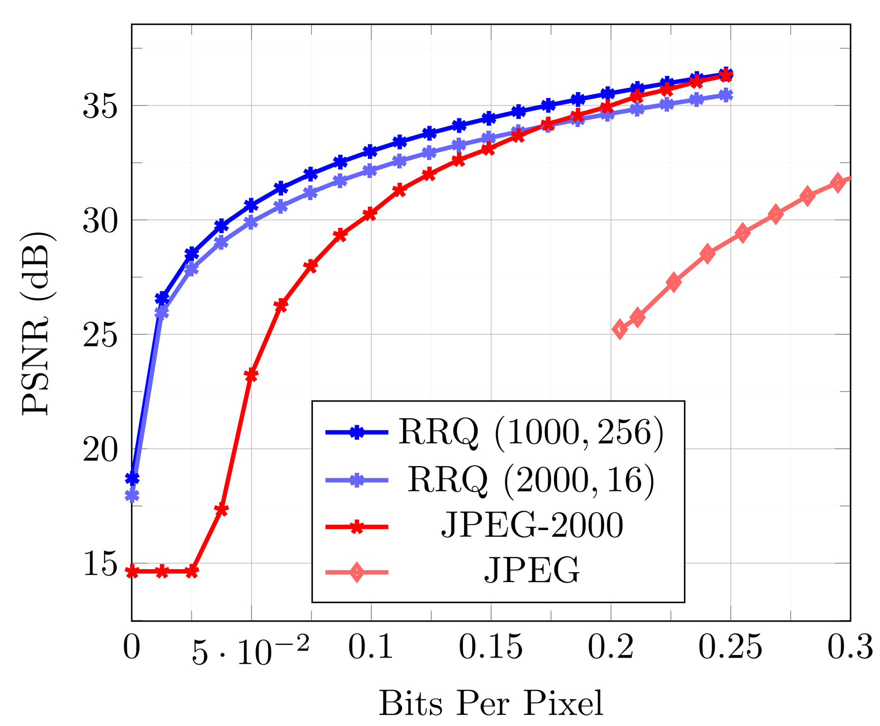


Image denoising

- Same setup as above, reconstruction using noisy test images.