Neural implementation of non-linear scalar quantization

Oleksandr Pankiv and Dariusz Puchala

Lodz University of Technology

Institute of Information Technology
Lodz University of Technology
Lodz, Poland

Data Compression Conference

IEEE Signal Processing Society
Technical Co-Sponsor
Microsoft
The University of Arizona

Introduction

Scalar quantization is an important component of modern lossy data compression systems including end-to-end approaches based on **artificial neural networks**.

In practical applications, uniform quantization is preferred for its simplicity, but in general **non-linear quantization** is optimal.

In this paper, we propose neural implementation of **companded quantization** which is the convenient tool to realize non-linear quantization.

Companded quantization

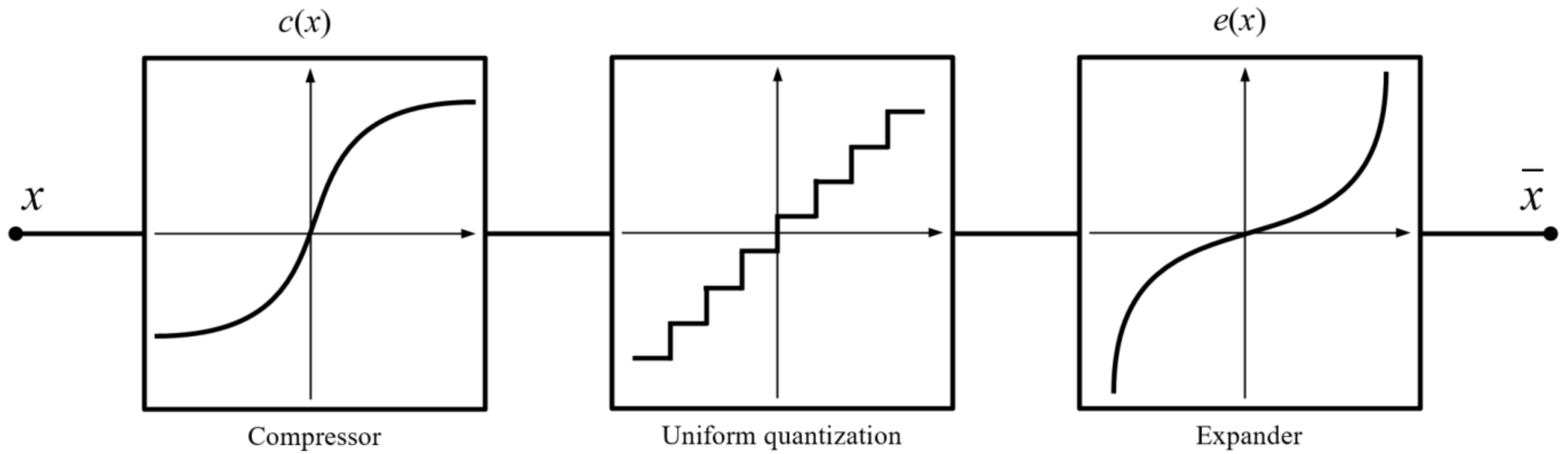

Figure 1: Block diagram of companded quantization.

Companded quantization

In this scheme, random variable x goes first through the **compressor** function $c(x)$. Its role is to properly **stretch** and **compress** the values of random variable depending on the probability density function $f_x(x)$ according to the following rule: “stretch” high probability regions and “compress” regions with low probabilities. In this way we can assign more reconstruction values to the regions of high-probability using uniform quantization.

As the next stage, we have **uniform quantization**. Finally, at the output of the scheme we apply the **expander** function $e(x)$ whose role is to reverse the effect of the compressor function.

Proposed method

In the proposed neural network based realization of non-linear companded quantization we follow the presented scheme but in place of compressor $c(x)$ and expander $e(x)$ we use **artificial neural networks**.

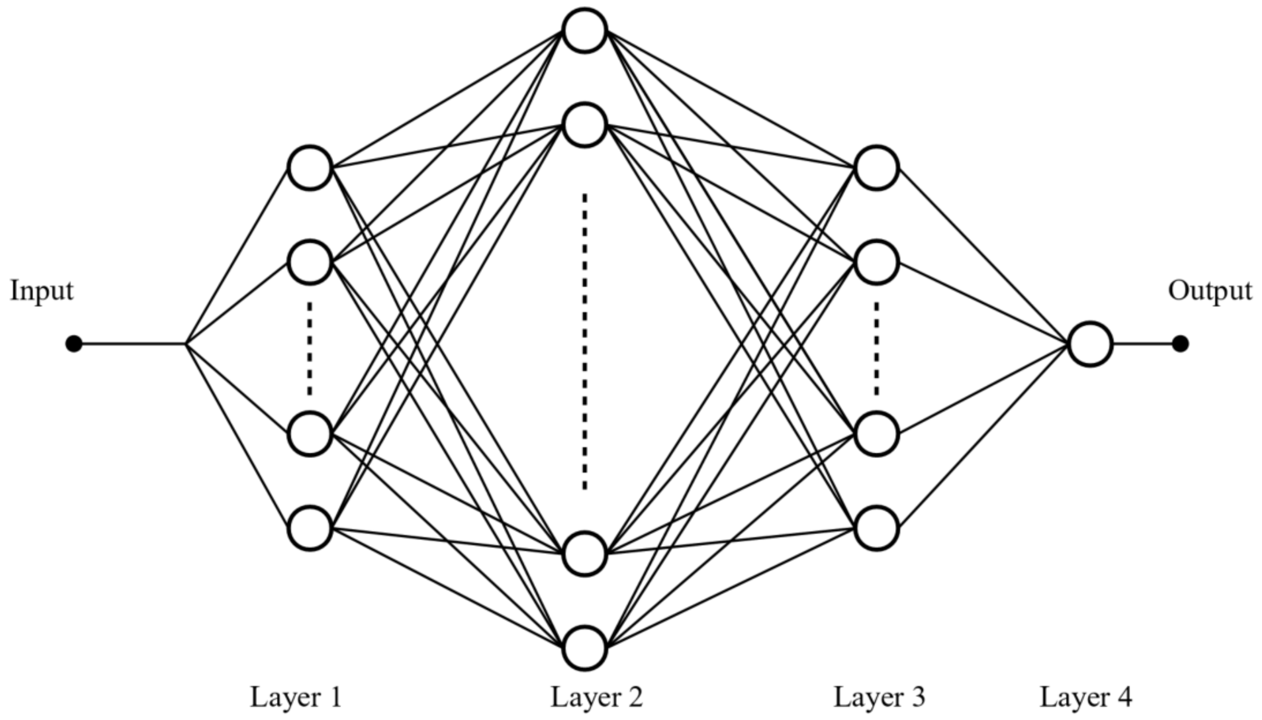

Figure 2: Neural realization of compressor $c(x)$ or expander $e(x)$ functions.

Proposed method

In the proposed method the uniform quantization is modelled using two different approaches:

- based on **rounding operation**,
- using additive and **uniformly distributed noise** with zeros expected value and properly selected variance.

Experimental results

Test images used during the experiments.

(a) (b) (c) (d)

Figure 3: Test images used during the experiments: (a) 'Lena.bmp', (b) 'Lake.bmp', (c) 'Pentagon.bmp', (d) 'Noise.bmp'.

Experimental results

Experimental results for „Lena.bmp" image.

Table 1: The experimental PSNR results in dB obtained for 'Lena.bmp' image.

M	Uniform	Lloyd-Max	Neural direct	Neural noise
2	16.85	19.75	19.70	19.32
4	22.88	26.02	25.98	24.42
8	28.90	31.58	31.89	31.40
16	34.67	36.21	37.44	37.03
32	40.73	41.31	42.83	42.63
64	46.37	47.21	48.27	48.39
128	51.15	54.15	53.91	52.55

Experimental results

Experimental results for „Noise.bmp” image (Gaussian noise).

Table 2: The experimental PSNR results in dB obtained for ’Noise.bmp’ image.

M	Uniform	Lloyd-Max	Neural real	Neural noise
2	15.44	22.49	22.49	22.49
4	22.76	27.29	27.29	27.03
8	28.81	32.31	32.45	32.37
16	34.80	36.12	37.47	38.05
32	40.72	41.02	42.60	43.45
64	46.36	47.17	47.68	48.66
128	51.13	54.15	53.16	53.82

Conclusions

Experimental results **proved** that the **proposed** neural realization of optimal scalar quantization **can be used** in modern “end-to-end” trained lossy data compression systems based on artificial neural networks.

The **proposed approach** allowed to obtain **better results** than **Lloyd-Max algorithm** which suffered from convergence issues.

The directions of the future research will be focused on **quantization** of many variables with a **bit budget constraint** and application to data compression based on **neural realization of Karhunen-Loève transform**.

Thank you for watching our video.