Efficient Storage of Images onto DNA Using Vector Quantization

Melpomeni Dimopoulou dimopoulou@i3s.unice.fr
Marc Antonini am@i3s.unice.fr
Labratoire I3S, Université Côte d'Azur, CNRS

Data Compression Conference

IEEE Signal Processing Society

THE PROBLEM:

- Conventional storage devices can endure for 5-10 years
- 80% of data is cold (very rarely accessed)
- High cost of reliable storage

THE SOLUTION: Use of DNA as a means of digital data storage

- Longevity
- High capacity

→ Need for efficient encoding to control the cost of DNA synthesis!

PURPOSE OF THIS WORK:

- To propose a new efficient method for the encoding of images into DNA using Vector quantization (VQ) improving the results obtained in our previous work** while controlling the DNA synthesis cost.

THE PROPOSED METHOD:

- DWT decomposition to reduce spatial redundancy
- VQ to encode each DWT subband
- Closed loop source allocation to optimally compress an image

1

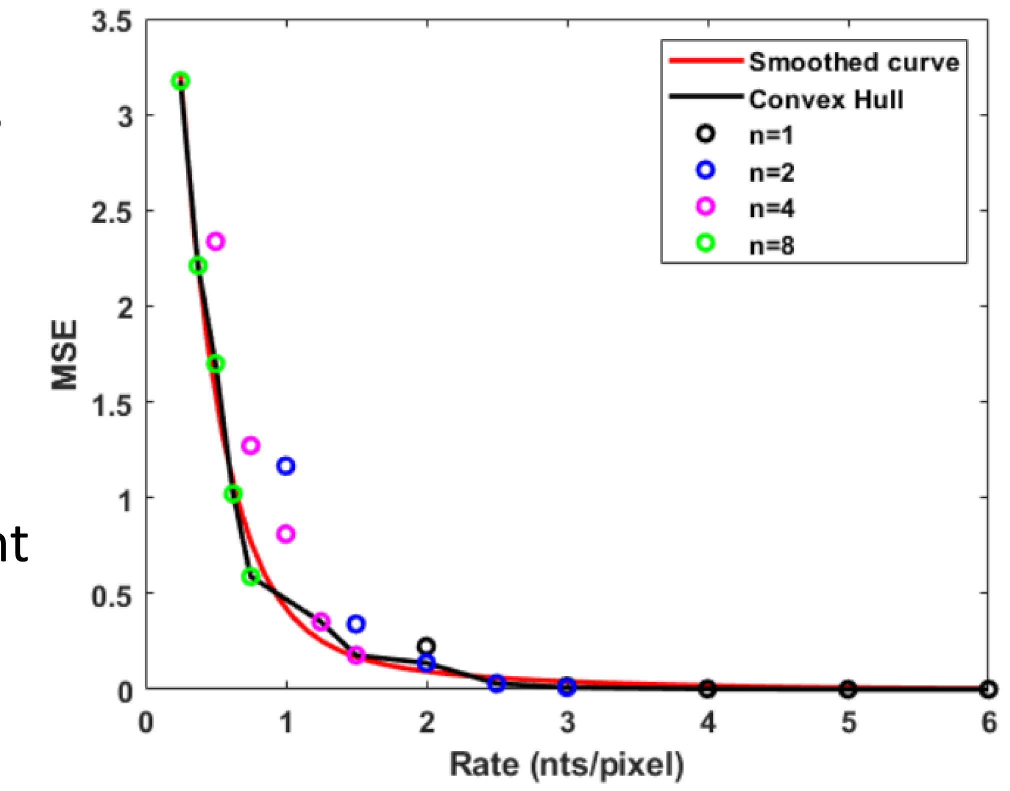
NUCLEOTIDE ALLOCATION:

Goal:

For a desired rate R_t, find optimal values of k (number of vectors) and n (length of vectors) for VQ that minimizes the distortion.

For each DWT subband:

- Build Rate-Distortion curve for different k and n
- Optimal points are lying on the convex hull



2

BIOLOGICAL CONSTRAINTS ON THE ENCODING:

(Reduction of sequencing error)

1) No homopolymers
2) %G,C ≤ %A,T
3) No repetition of short patterns

BUILDING A RESTRICTED QUATERNARY CODE:

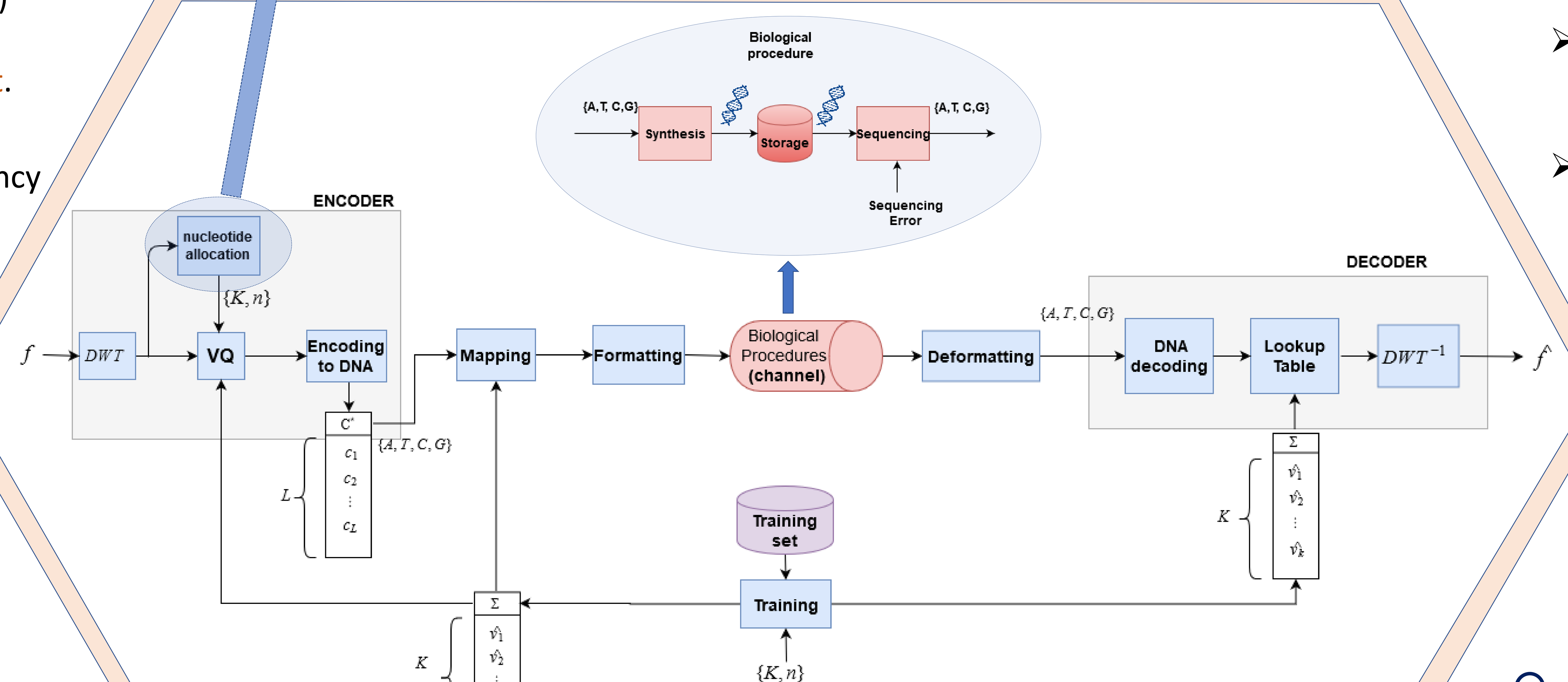
Two dictionaries of pair elements:

- $C_1 = \{AT, AC, AG, TA, TC, TG, CA, CT, GA, GT\}$
  - → *n picks:* $L = 10^n$ *codewords of length l=n*2*
- $C_2 = \{A, T, C, G\}$
  - → *adding a symbol from* C_2 *at the end of codewords:* $L = 10^n * 4$

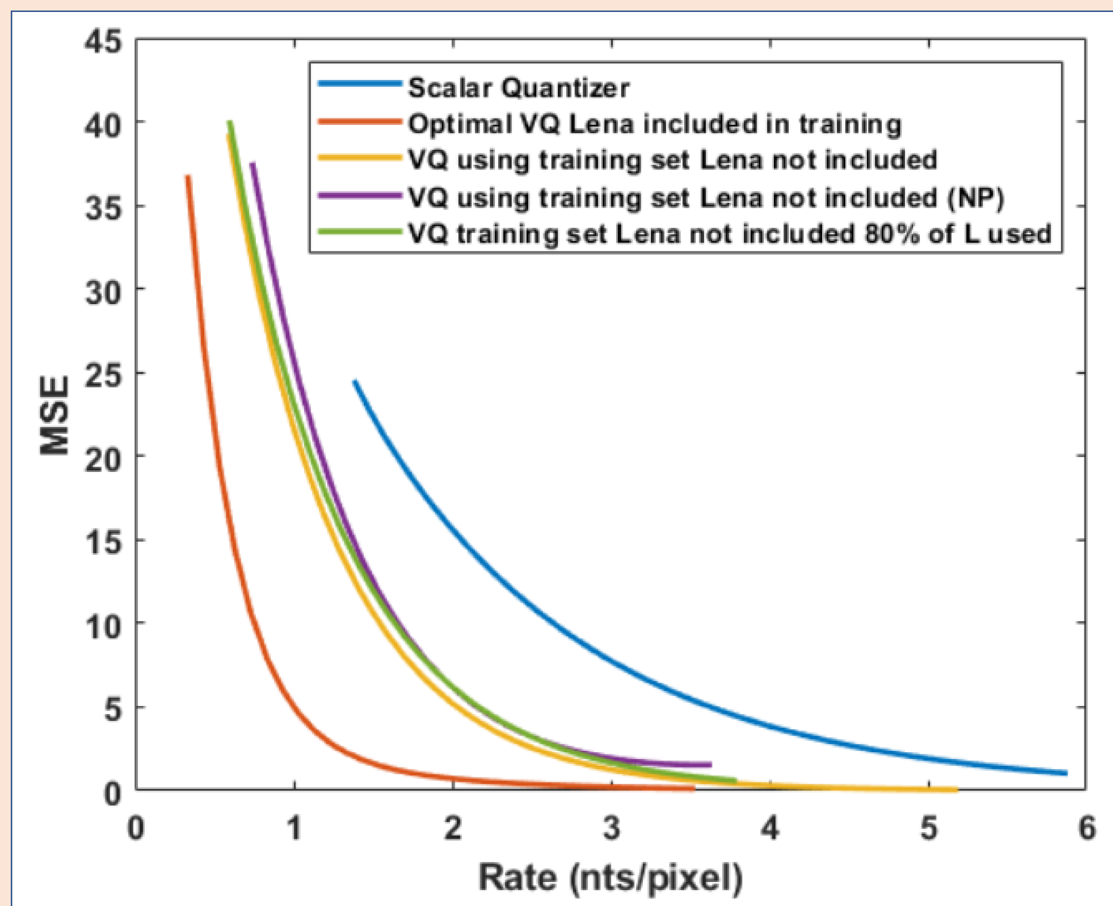
➢ Codewords of an even length l are built by picking $n = l/2$ pair-elements from C_1

➢ Codewords of odd length l codewords are built by picking $n = (l-1)/2$ pair-elements from C_1 and adding a pair element from C_2 at the end

3



The proposed encoding schema



6

EXPERIMENTAL RESULTS:

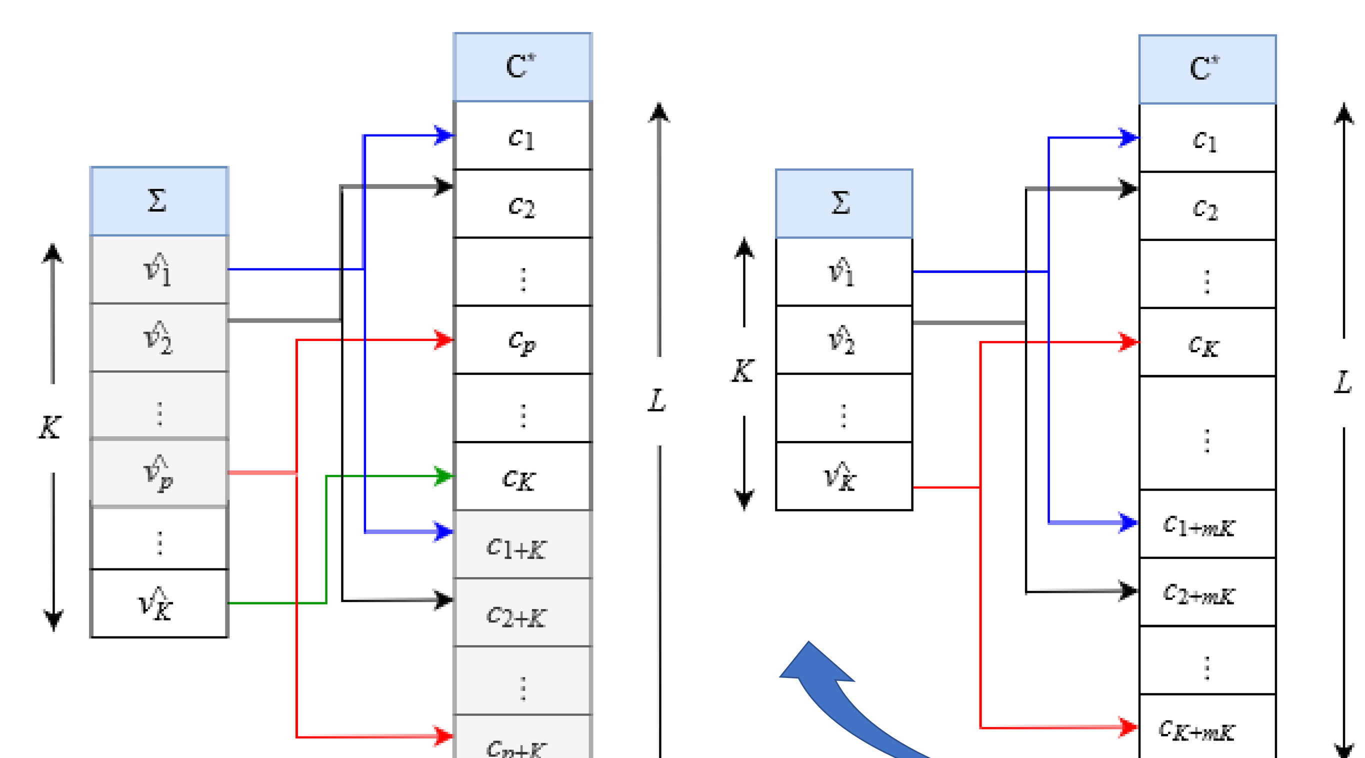
- Image: Lena 512x512
- Training set of different face images to obtain the vectors
- NP: No Patterns → $L = 2K$
- 80% of L used: K=80% L
- Results using VQ show great improvement compared to the results obtained in our previous work

** Dimopoulou, Melpomeni, et al. "A biologically constrained encoding solution for long-term storage of images onto synthetic DNA." *2019 27th European Signal Processing Conference (EUSIPCO)*. IEEE, 2019.

5' ATGT ATAT ATAT ATAT ATGT ATGT GTGT GTAT ATAT ATAT GTGT ATGT ATGT ATAT ACAT ATGT GTAT ATGT 3'
5' TGAAG TTGAA GCATA TGATG ACTCT GATCG AGCTC GTCGG TGCTT TGACT CTGAA TAAGC CTTCT TATAG 3'

Example of an oligo where k=L (top oligo) and an oligo were L=2K (bottom oligo).

The two ways for treating patterns. (1) on the left and (2) on the right



5

PATTERN REPETITIONS:

- VQ: efficient for compression
- More subband coefficients will be represented by the same vector
- Neighboring coefficients will be encoded to the same codeword → pattern repetitions!

Solution:

Increase code size L to allow double representation

MAPPING:

- To avoid patterns we need: $L \geq K$
- Mapping $\Gamma: \Sigma \rightarrow C^*$
- $m = \lfloor L/K \rfloor$
- $\Gamma(\hat{x}^i) = C^*(i + rand(0, m-1) * K)$

Two ways to treat pattern repetition:

1) If $K < L \leq 2K$: double representation of most frequent symbols , m=1 (left image)
2) If $L \geq 2K$: double representation of every word, m=2 (right image)

4