ISO/IEC JTC 1/SC29/WG1N100395
98th JPEG Meeting, Sydney, Australia, 16-20 January 2023

ISO/IEC JTC 1/SC 29/WG 1
(& ITU-T SG16)

Coding of Still Pictures

JBIG.
Joint Bi-level Image Experts Group

JPEG
Joint Photographic Experts Group

TITLE: **JPEG DNA Common Test Conditions version 1.1**
SOURCE: JPEG DNA AHG
EDITORS: Marc Antonini and Touradj Ebrahimi
PROJECT: JPEG DNA Exploration
STATUS: Draft
REQUESTED ACTION: Discussion
DISTRIBUTION: Public

Contact:
ISO/IEC JTC 1/SC 29/WG 1 Convener – Prof. Touradj Ebrahimi
EPFL/STI/IEL/GR-EB, Station 11, CH-1015 Lausanne, Switzerland
Tel: +41 21 693 2606, Fax: +41 21 693 7600, E-mail: Touradj.Ebrahimi@epfl.ch

Table of contents

JPEG DNA COMMON TEST CONDITIONS

version 1.1

1 Scope

The scope of JPEG DNA is the creation of a standard for efficient coding of images that considers biochemical constraints and offers robustness to noise introduced by the different stages of the storage process that is based on DNA synthetic polymers.

This document describes the Common Test Conditions (CTC) for the JPEG DNA image coding experiments. The main objectives of this document are:

- Define the common datasets that should be used in the evaluation of image coding solutions for storage on DNA support.
- Define the anchors (for direct encoding and transcoding) that should be used to comparatively evaluate the performance of image coding solutions for storage on DNA support.
- Define the coding conditions, in terms of source coding, error correction and biochemical noise simulators, as well as the target rates and compression ratios that anchors or alternative image coding solutions for storage on DNA support should demonstrate.
- Define the performance metrics for quality assessment that can be used to reliably evaluate the decoded images obtained from image coding solutions that produce streams in ACTG.
- Define the subjective evaluation procedure to perceptually evaluate all decoded images quality, namely the anchors and image coding solutions that produce ACTG.

In the current form, these common test conditions should be used to evaluate different aspects of image coding for storage on DNA support. The CTC are defined according with the use cases and requirements identified [17] and should be followed in all the experiments carried out by participants.

2 JPEG DNA Dataset

The JPEG DNA dataset is used for performance evaluation of image coding solutions for storage on DNA support. This JPEG DNA dataset is freely available to all JPEG DNA proponents which will be submitted in the framework of exploration experiments. The JPEG DNA dataset is organized according to:

- **Uncompressed dataset:** The uncompressed dataset provides a set of images to be used during the coding and decoding as defined in relevant exploration experiments.
- **JPEG compressed dataset:** The JPEG compressed dataset provides a set of already compressed images to be used during transcoding as defined in relevant exploration experiments.

The diversity of the images contained in the JPEG DNA dataset is high, namely in terms of their characteristics, such as content, color, and spatial resolution.

These datasets have the following characteristics:

- Format – PNG images (RGB color components, non-interlaced), JPEG 1 (including progressive and hierarchical modes), JPEG 2000 and JPEG XL compressed.
- Spatial resolution – TBD

Note: Link to the dataset and instructions on how to download to be included at a later date.

3 Evaluation Procedure

Objective and subjective quality evaluation will be performed by at least two independent organizations, following well-established procedures, and based on the decoded test images provided by each proponent. Proponents may perform encoding in any color space representation, but the input of the encoder and the output of the decoder must be in the PNG (in Y or in RGB color space) format as defined in the datasets. Objective image quality will be measured with luminance and color-based metrics and the RGB decoded images will be used for subjective quality evaluation.

4 Target Rates and Compression Ratios

The rate will be reported in two complementary and a third optional ways:

- The rate is expressed by the number of nucleotides (nts) per pixel (nts/pixel).
- The compression ratio is expressed by the number of input bits per nucleotide (bits/nts).
- The converted values in bits/pixel from the above two when possible.

An implementation of these rates can be found at: https://gitlab.com/wg1/jpeg-dna/jpeg-dna-metrics.

5 Objective Quality Evaluation

Objective quality testing shall be performed by computing several quality metrics, including $PSNR_Y$, $PSNR_{YUV}$, MS-SSIM, IW-SSIM, VMAF, VIFP, PSNR-HVS-M, NLPD and FSIM, between compressed and original images, at the target rates mentioned in the previous section. This section defines the objective image quality metrics that will be used for the assessment of learning-based image coding solutions. The reference implementation of $PSNR_Y$, $PSNR_{YUV}$ can be found https://gitlab.com/wg1/jpeg-dna/jpeg-dna-metrics and all other objective quality assessment metrics is available at: https://gitlab.com/wg1/jpeg-ai/jpeg-ai-qaf.

5.1 MS-SSIM Definition and Computation

Multi-Scale Structural SIMilarity (MS-SSIM) [1] is one of the most well-known image quality evaluation algorithms and computes relative quality scores between the reference and distorted images by comparing details across resolutions, providing high performance for learning-based image codecs. The MS-SSIM [1] is more flexible than single-scale methods such as SSIM by including variations of image resolution and viewing conditions. Also, the MS-SSIM metric introduces an image synthesis-based approach to calibrate the parameters that weight the relative importance between different scales. A high score expresses better image quality.

5.2 IW-SSIM Definition and Computation

Information Content Weighted Structural Similarity Measure (IW-SSIM) [2] is an extension of the structural similarity index based on the idea of information content weighted pooling. This metric assumes that when natural images are viewed, pooling should be made using perceptual weights that are proportional to the local information content. Moreover, advanced statistical models of the natural images are employed to derive the optimal weights which are combined with multiscale structural similarity measures to achieve the best correlation performance with subjective scores from well-known databases.

5.3 VMAF Definition and Computation

The Video Multimethod Assessment Fusion (VMAF) metric [3] developed by Netflix is focused on artifacts created by compression and rescaling and estimates the quality score by computing scores from several quality assessment algorithms and fusing them with a support vector machine (SVM). Even if this metric is specific for videos, it can also be used to evaluate the quality of single images and has been proved that perform reasonably well for learning-based image codecs. Since the metric takes as input raw images in the YUV color space format, the PNG (RGB color space) images are converted to the YUV 4:4:4 format using FFMPEG (BT.709 primaries). A higher score of this metric indicates better image quality.

5.4 VIF Definition and Computation

The Visual Information Fidelity (VIF) [4] measures the loss of human perceived information in some degradation processes, e.g. image compression. VIF exploits the natural scene statistics to evaluate information fidelity and is related to the Shannon mutual information between the degraded and original pristine image. The VIF metric operates in the wavelet domain and many experiments found that the metric values agree well with the human response, which also occurs for learning-based image codecs. A high score expresses better image quality.

5.5 PSNR-HVS-M Definition and Computation

The PSNR-HVS-M [5] is a simple and effective quality model which uses DCT basis functions and is based on the human visual system (HVS). The model operates with 8x8 pixel block of an image and calculates the maximum distortion that is not visible due to the between-coefficient masking. The proposed metric, PSNR-HVS-M, considers the proposed model and the contrast sensitivity function (CSF).

5.6. PSNR-Y and PSNR-YUV Definition and Computation

The *PSNR* between the original component, *I,* and the reconstructed component, *I',* (both n-bit) is computed as follows:

$$PSNR = 10\ log_{10}\frac{(2^n-1)}{MSE}$$

where the *MSE* between the two M×N images, *I* and *I',* is given by:

$$MSE = \frac{1}{MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}(I(i,j)-I'(i,j))^2$$

Once the *PSNR-Y*, *PSNR-U* and *PSNR-V* are individually computed, *PSNR-YUV* is computed using:

$$PSNR-YUV = \frac{6\,PSNR-Y + PSNR-U + PSNR-V}{8}$$

5.7 NLPD Definition and Computation

The Normalized Laplacian Pyramid Distance (NLPD) is an image quality metric [14] based on two different aspects associated with the human visual system: local luminance subtraction and local contrast gain control. NLPD exploits a Laplacian pyramid decomposition and a local normalization factor. The metric value is computed in the normalized Laplacian domain, this means that the quality of the

distorted image relative to its reference is the root mean squared error in some weight-normalized Laplacian domain. A lower score expresses better image quality.

5.8 FSIM Definition and Computation

The feature similarity (FSIM) metric [6] is based on the computation of two low level features that play complementary roles in the characterization of the image quality and reflects different aspects of the human visual system: 1) the phase congruency (PC), which is a dimensionless feature that accounts for the importance of the local structure and the image gradient magnitude (GM) feature to account for contrast information. The color version of the FSIM metric will be used. A high metric value expresses better image quality.

6 Subjective Quality Evaluation

To evaluate the selected coding solutions, a subjective quality assessment methodology should be used. Subjective quality evaluation of the compressed images will be performed on the test dataset.

The Double Stimulus Continuous Quality Scale (DSCQS) methodology will be used, where subjects watch side by side the original image and the impaired decoded image, and both are scored on a continuous scale. This scale is divided into five equal lengths which correspond to the normal ITU-R five-point quality scale, notably Excellent, Good, Fair, Poor, and Bad. This method requires the assessment of both original and impaired versions of each test image. The observers are not told which one is the reference image and the position of the reference image is changed in pseudo-random order. The subjects assess the overall quality of the original and decoded images by inserting a mark on a vertical scale. The vertical scales are printed in pairs to accommodate the double presentation of each test picture.

The subjective test methodology will follow BT500.13 [7] and a randomized presentation order for the stimuli, as described in ITU-T P.910 [8] will be used; the same content is never displayed consecutively. There is no presentation or voting time limit. A training session should be organized before the experiment to familiarize participants with artifacts and distortions in the test images. At least, three training images will be used before actual scoring.

To perform the tests, a semi-controlled crowdsourcing setup framework and/or a more controlled lab environment procedure can be used to show the images according to the DSCQS methodology. The semi-controlled crowdsourcing setup has been proven in the past its reliability, i.e. maintains a low variance of the scores [9]. The QualityCrowd2 [10] software and Amazon Mechanical Turk (or other similar platforms) will be used for crowdsourcing. The number of subjects will be large enough in order to draw conclusions in a statistically meaningful fashion.

7 Biochemical Coding Constraints

DNA data storage is a very error-prone process. The different components of the biochemical process that make possible the storage of data into synthetic DNA molecules, especially sequencing, generate a lot of errors. The error rates of these processes also depend on the DNA code that should be embedded in a molecule. Some patterns, motives and characteristics of the DNA codes have been identified as error generating and should be avoided to make the whole data storage process functional. This document aims at describing thoroughly these constraints. Any coder that aims at being a serious solution for DNA coding should follow these constraints, or at least approach a compliance of these constraints.

To identify if a coder respects or doesn't respect a constraint, it is not enough to check if the codebooks used in these algorithms are compliant with them. One should also check that ligatures between codewords respect them, for example. For that reason, a constraint-specific scope definition is necessary to check compliance. The document gives precisions on these scopes.

● Strand length limitations

- Definition

In the DNA data storage biochemical processes, it is very difficult to manage very long sequences of DNA, especially during synthesis. For that reason, the DNA codes are cut in subsequences of the same length. A DNA different molecule will be synthesized for each one of these subsequences that we call strands. A functional coder should cut the DNA coded in strands of a certain fixed length. Moreover, a strand index should be embedded in the strand's code to retrieve its content and to reassemble the full code in the correct order.

- Criteria

A satisfactory strand length should lie in the interval [100;300] bases. The strand length is generally fixed for all the coded data for ease of decoding. A random sequence can be padded to the ends of a strand to respect the constant strand length.

● Homopolymer runs

- Definition

A homopolymer run is the repetition of the same nucleotide several times in a row. When coding for DNA data storage, it is important to limit the number of repetitions of the same nucleotide several times in a row. The different sequencing technologies can allow different maximum lengths for homopolymers.

- Criteria

As a baseline, without any prior on the technologies that will be used in the biochemical process, a maximum homopolymer sequence of length 3 is asked for. All coders should avoid generating sequences of homopolymers of length more than 3.

- Scope

This criteria should be respected in the codebooks, in the possible ligatures of the different codewords used in the coder and in the different headers, indexes, identifiers and their possible ligatures embedded in the format.

● GC content balance

- Definition

The GC content describes the usage of the C and C bases in the coded data. More specifically, the GC content describes the percentage of all the bases in the code that are either a G or a C:

$$GC\_content = \sum_{i=0}^{l} \frac{\delta(c[i],\{G,C\})}{l} \delta(c[i],\{G,C\}) = \{1 \; if \; c[i] \; in \; \{G,C\}, 0 \; otherwise\},$$

l being the length of the code.

- Criteria

The accepted interval in which the GC content should fall in 40 to 60%. When over 50%, the error rate for nanopore sequencers significantly increases for some methods. A final satisfactory interval could be [40% 50%] [15].

- Scope

This criterion should be respected for every strand of the code.

● Repetition of patterns

- Definition

A short pattern is a sequence of bases of maximum length l.

- Criteria

The pattern should not be repeated too many times in the defined scope.

- Scope

The codewords used to encode the oligos should not be repeated forming the same pattern throughout the oligo length. [16]

● **Reverse complementarity**

- Definition

Reverse complements are pairs of sequences that are complementaries of each other's reversed sequence. In DNA, nucleotides are complementary: A and T are complementary base-pairs, and C and G are. This means that these two pairs can chemically bind together. Reverse complementary sequences are sequences of nucleotides that are one by one complementary when the two sequences are read in two different senses (one in direct, one in reverse). For example, ACTA and TAGT are complementary sequences. And ACTA and TAGT are reverse complementary sequences.

- Criteria

To be defined

- Scope

To be completed

8 Errorless Anchors Generation

8.1 Transcoder Anchor 1

Compressing using JPEG 1 and apply Goldman DNA coding [12]

The anchor 1 describes an encoding method that consists in cutting the already compressed JPEG binary file into bytes and encoding each byte separately. The byte wise encoder is composed of two blocks:

First block is a ternary Huffman coder that has the possible values of a byte as an alphabet. Each byte value has a ternary (written with 0s, 1s and 2s) codeword associated. The length of the codewords for each byte value will depend on the frequency of appearance of this value in the source. The more frequent a value, the shorter the codeword and the rarer the value, the longer the codeword.

The second block is a Goldman coder: it encodes the ternary bases (0s, 1s and 2s) of the ternary Huffman codewords with a rotating {A, T, C, G} alphabet into quaternary DNA-like codewords.

The reference implementation of this anchor can be found inside the Anchors repository at https://gitlab.com/wg1/jpeg-dna/jpeg-dna-anchors/-/tree/main/Compression%20Coding/Binary%20Source/Variable%20Length%20Direct%20Transcoder%20(Anchor%201)

8.2 Transcoder Anchor 2

Compressing using JPEG 1 and apply modified Goldman DNA coding using fixed-length ternary

The anchor 2 describes an encoding method that consists in cutting the already compressed JPEG binary file into bytes and encoding each byte separately. The byte wise encoder is composed of two blocks:

First block is a ternary fixed-length coder that has the possible values of a byte as an alphabet. Each byte value has a ternary (written with 0s, 1s and 2s) fixed length codeword associated.

The second block is a Goldman coder: it encodes the ternary bases (0s, 1s and 2s) of the ternary Huffman codewords with a rotating {A, T, C, G} alphabet into quaternary DNA-like codewords. In this method, each byte value is encoded with a codeword of length 6.

The reference implementation is available inside the Anchors repository at https://gitlab.com/wg1/jpeg-dna/Transcoder-Anchor-1

8.3 Transcoder Anchor 3

Compression using JPEG 1 and apply "Paircode-like" DNA coding [13]

The anchor 3 describes an encoding method that consists in cutting the already compressed JPEG binary file into bytes and encoding each byte separately. The byte wise encoder is composed of a fixed-length codebook that was generated with the aim of respecting the biochemical constraints of the DNA data storage channel. Each byte value has a fixed length codeword associated (a value is encoded with the codeword that has a codebook index equal to the value to encode). In this method, each byte value is encoded with a codeword of length 5.

The reference implementation of this anchor is available inside the Anchors repository at https://gitlab.com/wg1/jpeg-dna/jpeg-dna-anchors/-/tree/main/Compression%20Coding/Binary%20Source/Fixed%20Length%20Direct%20Transcoder%20(Anchor%203)

8.4 JPEG DNA Benchmark Codec (JPEG DNA BC)

The JPEG DNA python package is an implementation of the algorithm of the same name. The main purpose of the JPEG DNA algorithm is to encode the DCT coefficients into DNA-like quaternary code instead of binary code (see the figure 1). This work has been published in [11]. First, a quantization is applied, then a zigzag transform, the result is a sequence of integers in which the first element is the DC coefficient, then follow all the 63 AC coefficients.

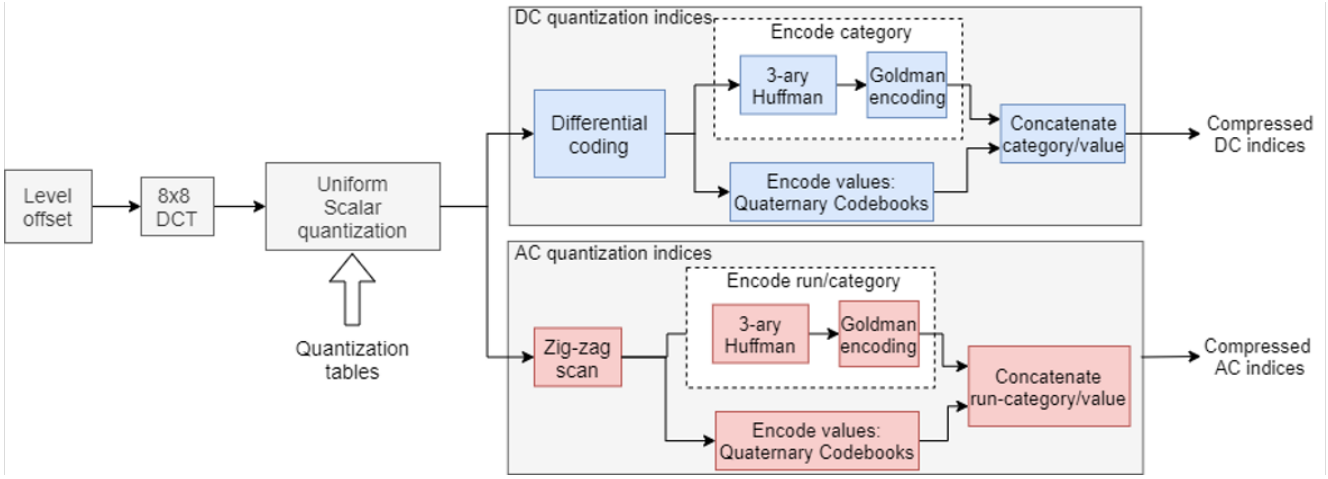

Figure 1: The JPEG DNA Benchmark codec.

The encoding system for DC and for AC coefficients share similarities with the usual JPEG algorithm but have their own specificity, especially in the coding of each value. Like in the JPEG algorithm, we use a system of categories and run/categories. The categories are used for the DC values and describe the interval in which the value falls. The run/categories in addition to the category of the coefficient also groups the number of zeros before this value.

The difference with the usual JPEG algorithm lies in the way categories, run/categories and values are coded. They are not anymore coded into binary data but into quaternary DNA-like data. This data will be synthesized into DNA molecules and the data content will be sequenced (read) from those molecules. The technologies are restricting the possibilities of quaternary sequences that can be encoded: we can't have homopolymers (repetitions of the same base), and the bases use must be balanced (G and C base must represent between 40% and 60% of the totality of the content).

To encode the categories and run categories, we use a combination of ternary Huffman coding with Goldman coding [12] on the ["A", "T", "C", "G"]. The Huffman coder generates codewords recursively with the alphabet {0,1,2} according to their frequency. The most frequent categories have a shorter codeword. The Goldman coder translates the ternary bases of these codewords with a rotating {A, T, C, G} alphabet into quaternary DNA-like codewords.

The values are coded using fixed-length codebooks that have been pre-generated and that respect the constraints previously described. We concatenate all the run/category and value codes to obtain a stream for the block. We concatenate the block codes to obtain the stream for the image.

The development of this algorithm has been oriented towards an easy-to-use, easy-to-modify paradigm. As previously described, the algorithm is working with a collection of coders (Huffman, Goldman), transforms (DCT, Zigzag). The library is composed of three main packages, one for coders, one for transforms and one for codecs. The latter one only contains the JPEG DNA codec. The coders and transforms used by these codecs are developed as modules in the packages of the same name. Also in the scripts package, one will find and may add general scripts for those codecs to perform compression, decompression, performance evaluation, etc.

A thorough explanation of the package can be found in the input document number M93103, and a wiki for documentation and link to the code can be found here www.i3s.unice.fr/~am/JpegDNA/. In order to access the code please contact xpic@i3s.unice.fr.
An implementation can be found at https://gitlab.com/wg1/jpeg-dna/jpeg-dna-benchmark-codec

8.5 JPEG DNA BC Transcoder

Existing JPEG files can be losslessly transcoded to JPEG DNA BC and vice-versa.
Implementation available at https://gitlab.com/wg1/jpeg-dna/JPEG-DNA-BC-Transcoder

8.6 Full Decoding/Re-encoding

This anchor consists in decoding an already compressed JPEG binary file into a raw image, then encode it into DNA with the JPEG DNA BC codec. To decode and retrieve the image, the JPEG DNA BC decoder is enough.

The quality chosen to re-encode the raw image into DNA with the JPEG DNA BC codec should be close to the parameter used during the previous JPEG compression.

9 Experimentation Workflow

This section defines a generic procedure to conduct experiments on any codec for DNA data storage. The general workflow of those experiments is proposed in the figure 2 and is composed of different tools that deal with specific processes. The main components of this workflow are:

- The codec (that is able to both encode the input image into a pool of formatted DNA-like sequences called oligos and decode a pool of formatted oligos into an image),
- The noise model (a simulator that alters an input pool of formatted oligos by introducing errors – insertions, deletions, substitutions – approximating the behavior of the real biochemical processes – synthesis, storage, amplification, sequencing –),
- The filtering system or oligo selector, that, from the noised oligos, gets rid of the oligos that contain too many errors,
- The consensus, that from a set of erroneous oligos will generate a pool of oligos most likely to be the non-erroneous one.

For each process in each component, it is important to identify the external parameters that need to be adjusted to conduct thorough experiments. They are:

- The coding rate, determined during the encoding process,
- The length of the formatted oligos, and the primers used for the oligos during formatting,
- The noise level for all the components of the noise model (synthesis, storage degradation, amplification and sequencing).

All the information coming from the consensus is enough for the decoder to decode data, no additional input is required for the decoder.

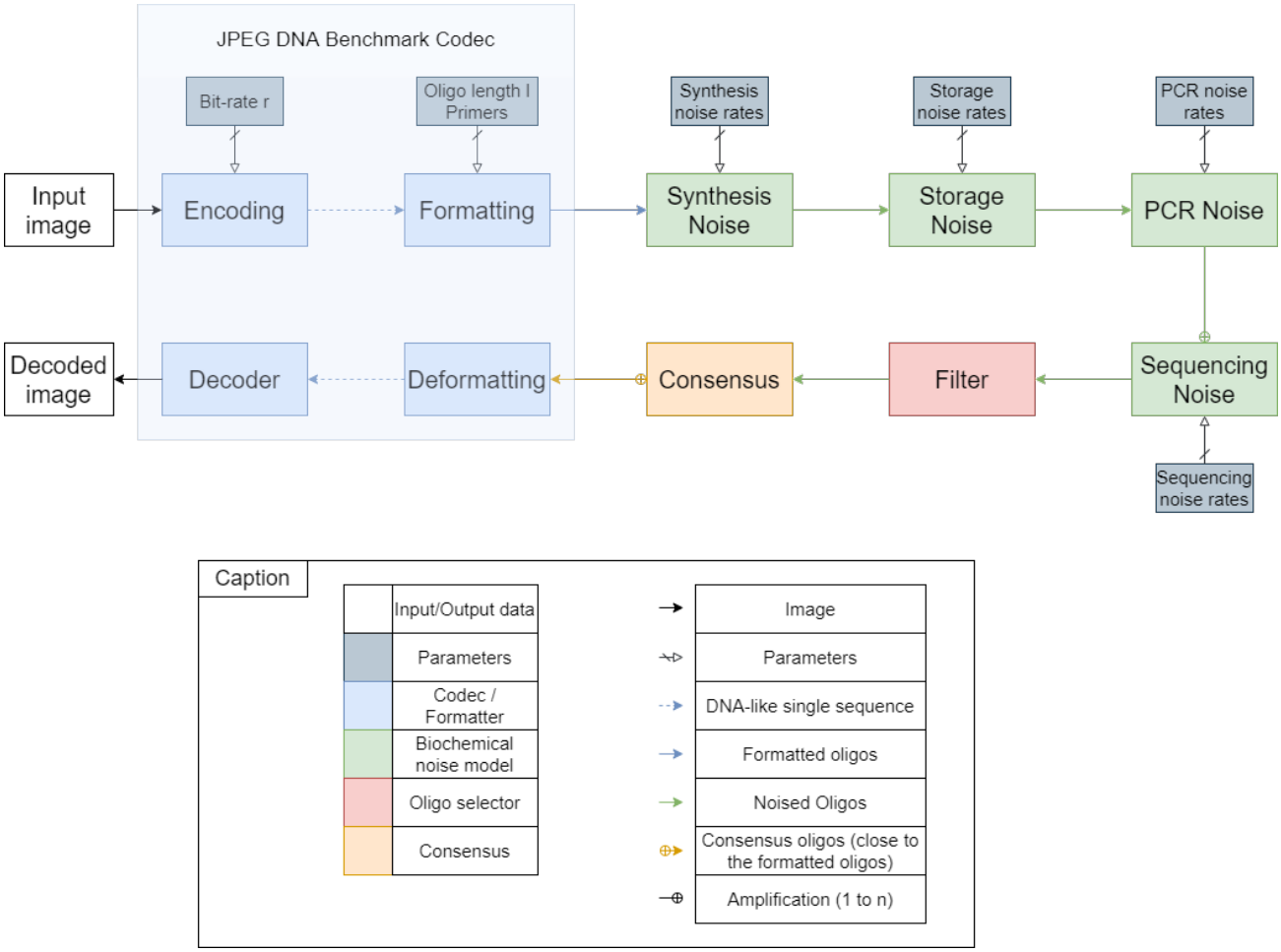

Figure 2: General workflow for image coding/decoding simulations.

10 Naming Convention for Decoded Images and Streams

To be completed

11 Evaluation Framework and Results Reporting Template

To be completed

13 References

[1] Z. Wang, E. P. Simoncelli, and A.C. Bovik, "Multiscale structural similarity for image quality assessment," ThirtySeventh Asilomar Conference on Signals, Systems & Computers, Pacific Grove, CA, USA, November 2003.

[2] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment", IEEE Transactions on Image Processing, Vol. 20, No. 5, May 2011.

[3] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy and M. Manohara, "Toward a practical perceptual video quality metric", [Online], Available here.

[4] H.R. Sheikh and A. C. Bovik, "Image information and visual quality" IEEE International Conference on Acoustics, Speech, and Signal Processing, Montreal, Canada, August 2004.

[5] N. Ponomarenko, F. Silvestri, K. Egiazarian, M. Carli, J. Astola, V. Lukin, "On between-coefficient contrast masking of DCT basis functions", Proceedings of the third international workshop on video processing and quality metrics (Vol. 4), 2007.

[6] L. Zhang, L. Zhang, X. Mou and D. Zhang, "FSIM: a feature similarity index for image quality assessment," IEEE Transactions on Image Processing, vol. 20, no. 8, pp. 2378-2386, August 2011.

[7] ITU-R Recommendation BT.500-13, "Methodology for the subjective assessment of the quality of television pictures," International Telecommunications Union, Geneva, Switzerland, 2012.

[8] ITU-T Recommendation P. 910, "Subjective video quality assessment methods for multimedia applications," International Telecommunication Union, Geneva, 2008.

[9] D. Saupe, F. Hahn, V. Hosu, I. Zingman, M. Rana, and S. Li, "Crowdworkers proven useful: a comparative study of subjective video quality assessment," International Conference on Quality of Multimedia Experience (QoMEX), Lisbon, Portugal, June 2016.

[10] C. Keimel, J. Habigt, C. Horch, and K. Diepold, "Qualitycrowd: a framework for crowd-based quality evaluation," Picture Coding Symposium, Krakow, Poland, May 2012.

[11] M. Dimopoulou, E. Gil San Antonio and M. Antonini, "A JPEG-based image coding solution for data storage on DNA", European Signal Processing Conference (EUSIPCO), Aug. 2021.

[12] N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E.M. LeProust, B. Sipos, and E. Birney, "Towards practical, high-capacity, low-maintenance information storage in synthesized DNA", Nature, vol. 494, no. 7435, pp. 77–80, 2013.

[13] M. Dimopoulou, M. Antonini, P. Barbry and R. Appuswamy, "A biologically constrained encoding solution for long-term storage of images onto synthetic DNA", 27th European Signal Processing Conference (EUSIPCO), Sep. 2019.

[14] V. Laparra, J. Ballé, A. Berardino, and E. P. Simoncelli, "Perceptual Image Quality Assessment using a Normalized Laplacian Pyramid", S&T Symposium on Electronic Imaging: Conf. on Human Vision and Electronic Imaging, San Francisco, CA, USA, February 2016.

[15] C. Delahaye and J. Nicolas, "Sequencing DNA with nanopores: Troubles and biases", PLOS ONE 16(10): e0257521, 2021

[16] T. J. Treangen and S. L. Salzberg, "Repetitive DNA and next-generation sequencing: computational challenges and solutions", Nature Reviews Genetics, vol. 13, no. 1, p. 36, 2012

[17] ISO/IEC JTC 1/SC29/N100252, Use Cases and Requirements for DNA-based Media Storage version 1.0, 96th Meeting, Online, 25-29 July 2022