Nature Inspired JPEG Quantization Optimization

Pedro Henrique Guimarães Ferreira, Osmar Luiz Ferreira de Carvalho and Eduardo Peixoto

***Abstract*— The JPEG standard is the most used algorithm for image compression, with billions of decoders deployed in either software or hardware. The standard allows for custom quantization tables, that are transmitted in the bitstream. This work examines the application of two promising nature inspired metaheuristics, Particle Swarm Optimization and Dual Simulated Annealing, in the generation of custom, image-specific, rate-distortion optimal quantization tables for the JPEG. Our results show that this approach is capable of producing a JPEG compliant, backwards compatible images that presents a compression rate about 9% smaller than standard JPEG.**

***Keywords*— Image Compression, Nature Inspired Heuristics, Numerical Optimization, JPEG Quantization Tables.**

I. Introduction

Image compression is an important research topic, essential for some applications such as digital photography and even internet browsing. Among many standards for image compression, the JPEG standard [1], [2] has stood the test of time, remaining for more than 25 years as both the most widely adopted image compression standard and the most used image format in the world [3]–[5].

In order to ensure cross-compatibility any JPEG file has to comply with a standardized format defined by JPEG's guideline [1]. Since JPEG started to become widespread, hardware implementations have been developed in various platforms and nowadays dedicated hardware accelerators for JPEG are present in most of modern devices [5], guaranteeing very fast encoding and decoding cycles, which gives it a huge advantage against newer compression standards, as they do not benefit from the existing hardware, being unable to match JPEG's compression speed.

Therefore, developing modifications on top of JPEG, so it can take advantage of the existing hardware, instead of developing new algorithms is still an efficient way to promote widespread enhancements on image compression with immediate application. In this regard, a lot of research has been conducted in determining and design custom quantization tables, ranging from designing quantization tables for specific purposes, as enhancing the feature detection [6], image retrieval [7] or the preservation of medical image properties [8] on the compressed images, to heuristics to generate a custom optimized table for each image [9], [10].

Since the quantization table design problem can be seen as a multi-objective, non-differentiable, non-smooth optimization problem, where one aims to minimize both the rate and the distortion of the compressed image in relation to the original image, nature-inspired heuristics have stood up as one of the leading approaches to this purpose [10], [11].

Departamento de Engenharia Elétrica, Universidade de Brasília.{pedroferreira, osmarcarvalho, eduardopeixoto}@ieee.org.

This work focuses in applying two of the most promising nature inspired heuristics, Particle Swarm Optimization (PSO) [12] and Dual Simulated Annealing (DSA) [13], in order to create custom, image-specific, quantization tables by directly optimizing a rate-distortion metric. Since custom quantization tables are a standard JPEG feature, the output of our encoder produces enhanced compressed images that are still JPEG compliant. Results show that our proposed algorithms outperform the standard JPEG algorithm by an average 10% in terms of bitrate used for the same quality.

II. Related Work

According to the JPEG standard [1], the JPEG encoding process is described by four major steps: first, for each color component, the image is divided into 8x8 tiles, then a reversible, linear transform, usually the Discrete Cosine Transform (DCT), is applied at each tile. Next, each transformed block is flattened into a 64 elements array and quantized according to an 8x8 quantization table, and last, the quantized coefficients are rearranged in a procedure called zigzag scan and submitted to entropy encoding.

As the only lossy step on the compression process occurs in the quantization stage, the quality of the compressed image is effectively controlled by the quantization table. Therefore, the quantization table should be explicitly (directly encoded on bitstream) or implicitly (via the declaration of the quality factor) provided as an input to the JPEG decoder,guaranteeing, through an universal support of custom quantization tables, a fine-grained control on the compressed image quality.

Multiple approaches for generating custom quantization tables were proposed to take full advantage of this feature, of which direct rate-distortion optimization and nature-inspired excelled themselves and gained ground as the leading paradigms in the literature [10], [11].

In terms of rate-distortion optimization, *Wu and Gersho* [14] proposed an optimization lying on minimizing the variation of the distortion in relation to the compression rate, obtaining good results, while *Ramchandran and Vetterli* [15] has proposed a fast rate-distortion optimization algorithm that works both for JPEG and MPEG based on thresholding the quantization scales, also obtaining real compression gain with relation to the standard JPEG quantization tables.

On the other hand, Particle Swarm Optimization (PSO) has also enjoyed considerable success in the literature. *Li et al* [16], *Snasel et al.* [17], *Fazli et al.* [18] and *Rabevohitra and Sung* [19] have all found success applying PSO to design custom quantization tables to improve the quality of hidden messages in steganography. On its turn, *Ma and Zhang* [20] have suggested a cultural-based multi-objective PSO algorithm for choosing the quantization tables, establishing a pareto front of possible quantization tables, achieving a significant visual enhancement, despite the fact that no evaluation was provided. More recently, *Abbood* [21] has also employed PSO for optimizing quantization tables, but in a context of minimizing compressed image distortion, obtaining state-or-the-art results.

Lastly, the Simulated Annealing algorithm was recently explored as an alternative to design quantization tables in *Hopkins et al.* [22], optimizing over an image dataset, finding static (not image specific) quantization tables that considerably outperforms JPEG standard tables.

III. Proposed Method

Differently from these other works, the goal of the proposed method is to optimize a quantization table specific for the image being encoded. This quantization table will be transmitted in the bitstream, as specified by the JPEG standard [1], producing a JPEG compliant bitstream. In order to verify this concept, two different high-performing swarm intelligence methods, PSO [12] and DSA [13], are employed to optimize both chrominance and luminance quantization tables.

Both of these methods usually work by minimizing a given fitness function. Unlike lossless compression methods, which can have their performance assessed in terms of a single, uni-dimensional metric, the compression rate, evaluating and comparing the performance of lossy compression methods can be particularly complicated, since lossy compression is, by its very nature, a multi-objective problem where we aim to minimize simultaneously the size of the compressed data and the distortion caused by the compression process, even though there is always a trade-off between them [23].

In terms of image compression specifically, the evaluation criteria has often lied in determining a cost function linking the compression and the distortion rates, allowing us to discriminate whether a given operation point gives us a better compromise between compression and distortion rates than other. For this purpose, multiple rate-distortion metrics were developed in the context of nature inspired computing, as exhaustively described in [11], with the traditional Lagrangian Cost Function standing out as the most common metric. But while these metrics succeed in establishing an exchange rate between compression rate - usually, expressed in bits per pixel - and distortion - often defined in terms of PSNR -, they do so by setting arbitrary, static exchanges rates based on mathematical or statistical assumptions.

This work proposes a new approach, replacing the traditional cost functions for a image-specific fitness function, generated by interpolating the base rate-distortion curve obtained by encoding the image using the provided default quantization tables. The proposed function, denominated Expected Rate Gain (ERG), is explained as follows.

A. The Expected Rate Gain (ERG)

From a theoretical, mathematical point of view, any JPEG compression process can be summarized as a relation between two ordered pairs, one composed by the original image and the quantization table and the second one, yielded by the compression process from the first pair, composed by the size of the compressed image and an appropriate distortion measure, such as the PSNR. As an abuse of notation, we can represent this idea using the following notation:

$$C : (I, Q) \to (S, PSNR) \tag{1}$$

Where C denotes the compression process, (I, Q) stands for the ordered pair original image, quantization table, respectively and $(S, PSNR)$ is the ordered pair for the size and PSNR of the compressed image, respectively.

Since $(S, PSNR)$ is an ordered pair, if we can ensure that for every value of S, there is only one value of $PSNR$, so we can also define a function $E : S \to PSNR$. Even though we cannot strictly guarantee that this condition holds for every quantization table for a given image, we can empirically verify that, for sufficiently different quantization tables, such as the quantization tables produced by varying the quality factor (q) of the compression, this condition holds, which is sufficient to allow us to define our intended function.

Moreover, since we supposed there is a (direct) relationship between the size of the compressed image, S, and the achieved $PSNR$, we can also estimate this relationship through a linear interpolation of the experimental points of the baseline quantization tables - the quantization tables produced varying the quality factor (q) from 1 to 100.

Therefore, we can chose a set $T = \{(S, PSNR) : C\,(I, Q\,(q)) \mid 0 \leq q \leq 100\}$ where it is possible to define the function $E : S \to PSNR$, which allows us to define $E\,(S)$ as following:

$$E\,(S) = interp\,(S, PSNR) \tag{2}$$

where $interp\,(S, PSNR)$ correspond to the linear interpolation considering the S points as the points for the x coordinate, while the $PSNR$ points are the points for the y coordinate, and $E\,(S)$ stands for the Expected PSNR provided the compressed image has size S.

Provided with the concept of Expected PSNR, we can finaly define our main metric, the Expected Rate Gain (ERG). For a given pair $(S_c, PSNR_c)$ submitted to the compression process C, with an Expected PSNR function $E(S)$, we can find an S_o that solves $E\,(S) = PNSR_c$, and thus, define the Expected Rate Gain as:

$$ERG\,(S_c, PSNR_c) = \frac{S_c}{S_o} \tag{3}$$

B. The Fixed Quality Expected Rate Gain (FQ-ERG)

We can intuitively understand the Expected Rate Gain as a measure of quantization table enhancement, since, for a custom quantization table, it effectively measures the ratio of the compressed image size and the size that an image compressed with the baseline, standard quantization tables would have to achieve the same distortion rate (PSNR).

For the same reason, it is pretty straightforward to realize that an attempt to optimize directly the ERG will lead to determining the operating point where the custom tables outperforms the standard tables by most. Although this result can be useful in some contexts, allowing an automatic, optimal choice of the operation point for compressing a image, its range of applications is very limited since it fails to provide support for a crucial feature: controlling the image target quality.

To solve this issue, we propose the Fixed Quality Expected Rate Gain (FQ-ERG), an adaption on the ERG to enable it to support image quality control. The FQ-ERG is defined as following:

$$FQ-ERG = \begin{cases} ERG(S_c, PSNR_c) & \text{, if } |PSNR_c - PSNR_t| \leq \epsilon \\ ERG(S_c, PSNR_c) + |PSNR_c - PSNR_t - \epsilon| \cdot P & , \\ \quad otherwise \end{cases} \quad (4)$$

where S_c and $PSNR_c$ are the same as in the Expected Rate Gain, $PSNR_t$ is the targeted PSNR for the compressed image and ϵ defines the PSNR drift tolerance, and P defines a penalty for PSNRs out of the tolerated PSNR range.

This new metrics expands the power of Expected Rate Gain allowing users to set a desired distortion rate, creating, therefore, a usable, high performing fitness function that requires very little parameter tuning, as we can derive a very intuitive heuristic to find good values for the hyper-parameters in an automatic fashion while also preserving its compatibility with the current default way to choose the targeted image quality, setting the value of the quality factor q.

For this, if a raw PSNR value is targeted, we can take advantage of the fact that, when establishing a desired image quality, we are not actually interested in an exact distortion rate, but rather in the visual experience that corresponds for that given rate. Thus, we can establish $\epsilon = 0.5$ to ensure the target image will not deviate more the 0.5 dB from the target quality. Alternatively, if the image quality is provided in terms of a strictly quality factor q, we can propose that $\epsilon = min(| PSNR(q-1) - PSNR(q) |, | PSNR(q+1) - PSNR(q) |)$ is also a suitable choice of value for ϵ, since it ensures that the desired optimization will take place only in the region that corresponds to the desired quality factor.

On the other hand, for most of the applications, a value of 2 for the penalty factor P already imposes a punishment harsh enough to guarantee that the solutions will likely lie in the desired quality range, but from a theoretical point of view, we can generalize this idea advocating that P should assume the value of twice the slope of the PSNR in the tolerange range, or, in other terms, we can find S_a and S_b that solves $E(S_a) = PSNR_t - \epsilon$ and $E(S_b) = PSNR_t + \epsilon$ and then assume a P given by the following equation:

$$P = 2 \cdot \frac{2\epsilon}{S_b - S_a}. \quad (5)$$

C. Particle Swarm Optimization (PSO)

The Particle Swarm Optimization is a population-based meta-heuristic where the candidate solutions, called particles, search a n-dimensional search space, varying their position (X) according to an stochastic parameter called velocity (V), which is influenced both by particle's (P) and swarm's (G) best known position [12], [24], [25], according to the following update rules:

$$\begin{aligned} V_i^{t+1} &= C_0 V_i^t + C_1 R_1 \left(P_i - X_i^t\right) + C_2 R_2 \left(G - X_i^t\right) \\ X_i^{t+1} &= X_i^t + V_i^{t+1} \end{aligned} \quad (6)$$

where R_1 and R_2 are random values sampled from a uniform distribution, and C_0, C_1 and C_2 are constants.

Since the Quantization Table Optimization problem can be seen as a 128-dimensional optimization problem on $(0, 255)$ bounded search space, where each of the first 64 dimensions of the search space correspond to the values of the chrominance quantization table, whereas the other 64 dimensions corresponds to the luminance quantization table, PSO can be directly applied to the Quantization Table Optimization problem given a fitness function, such as compression rate, lagragian rate-distortion cost function or even the newly defined FQ-ERG.

Despite the fact that PSO is a global optimization algorithm, being expected, therefore, that it will be able to find the desired global minimum regardless how it is initialized, if a distortion rate is targeted, we can take advantage of JPEG's standard tables to speed up the convergence, defining each particle in the initial population as a sum of the standard quantization table that yields the closest distortion rate to the desired one and with a discrete random vector of the same dimensions.

D. Dual Simulated Annealing (DSA)

The DSA is a stochastic global optimization metaheuristic [13], based on the combination of both Classical [26] and Fast Simulated Annealing [27] with local search improvements [28]. The idea behind SA is to minimize a fitness function in the same fashion as physical systems minimize their internal energy, Thus, for each step, the system stochastically decides between staying on its current position or transitioning to a new, neighbor position.

The acceptance probability of a transition is determined both by the value of the fitness function evaluated in the current and in the neighbor positions and a external, time-varying, hyper-parameter called Temperature, which controls the operation point of the exploration-exploitation by adjusting the tendency of the algorithm to accept uphill moves. For the quantization table optimization, we will employ the most frequent probability acceptance function, the Cauchy-Lorentz visiting distribution, with a hyperbolic decay temperature function, as comprehensively described in [29] and [30].

Analogously to the Particle Swarm Optimization, the Duan Simulated Annealing can also be applied for optimizing JPEG Quantization Tables in a very straight-forward fashion, representing chrominance and luminance tables by a 128-dimensions vector and accepting the closest standard quantization tables to the target image quality as the initial guess.

IV. Experimental Results

To evaluate the performance of the proposed methods, PSO and SA are employed to generate quantizations tables by optimizing both a typical Lagrangian Rate-Distortion Cost Function ($J = D + \lambda R$, where J is the cost, D is the distortion, measured as the mean squared error, R is the rate, amd λ is

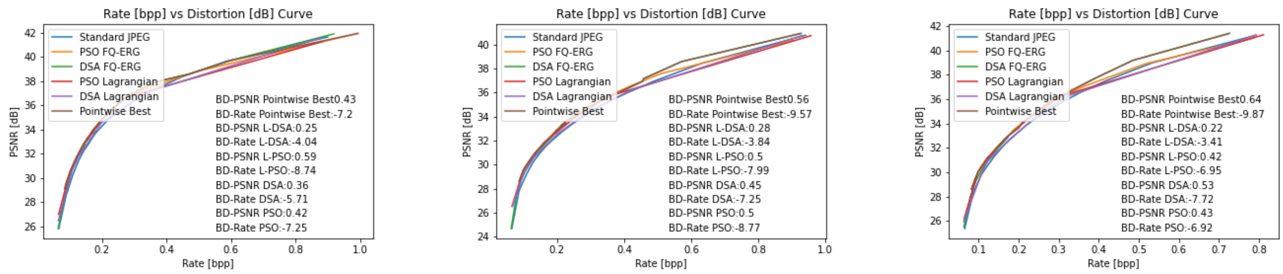

Fig. 1: Rate-distortion results for images: (a) kodim09, (b) kodim15, and (c) kodim20.

a parameter that controls the exchange between ditortion and rate) and the newly proposed Fixed Quality Expected Rate Gain (FQ-ERG).

The algorithms were benchmarked on the *Kodak Image Dataset*, presented on Fig. 2, a dataset composed by 24 uncompressed, 768 x 512 true color images. For each image, each algorithm generated a custom quantization table for 19 different target quality factors, from $q = 5$ to $q = 95$ in steps of 5. For the Lagrangian Rate-Distortion Optimization, the value of the Lagrangian multiplier λ was determined, for each point, through a grid search.

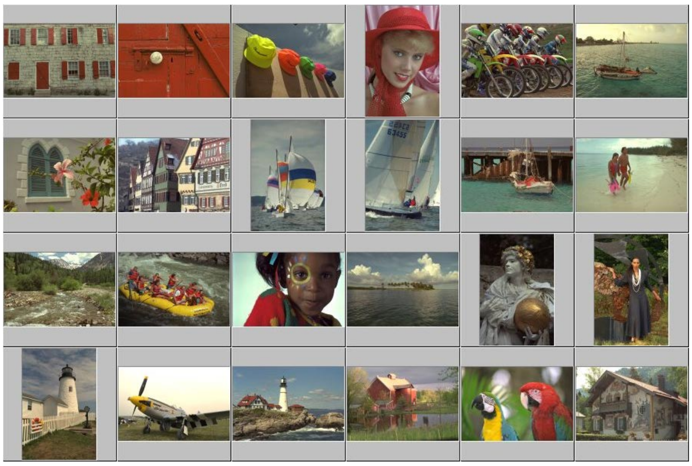

Fig. 2: The Kodak Image Dataset

For the PSO, our reference implementation was the one from DEAP library [31], with the following hyper-parameters:

- Population Size = 20
- Maximum Number of Generations = 50
- Maximum Local Update Factor = 2.0
- Maximum Global Update Factor = 2.0
- Minimum Speed = -3.0
- Maximum Speed = 3.0

On the other hand, the Scipy's Dual Simulated Annealing implementation [32] is taken as our reference implementation for DSA, with the default set of parameters, except for the maximum number of allowed function evaluations (maxfun), which is set as 1000. For the JPEG encoding, the popular Independent JPEG Group's JPEG still image codec v9 encoder is employed [33].

The described methods are compared with respect to size of the compressed image (bits per pixel), Peak Signal-to-Noise Ratio (PSNR) and both Rate and PSNR Bjontegaard Delta values [34].

Method	**BD-Rate** (%)	**BD-PSNR** (dB)
PSO FQ-ERG	-7.96	0.47
DSA FQ-ERG	-7.40	0.46
PSO Lagrangian	-7.92	0.49
DSA Lagrangian	-3.21	0.21
Pointwise Best	-9.11	0.56

TABLE I: Bjontegaard Delta for each Method

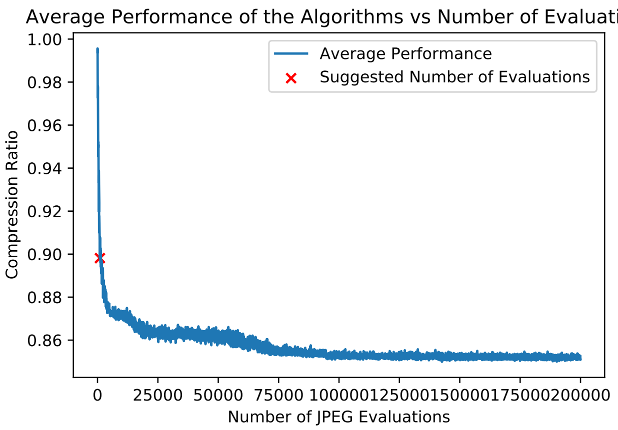

Fig. 3: Average Compression Ratio vs Number of JPEG Evaluations

As we can see in the table I, both PSO and DSA optimizations yield considerable enhancements in compression, providing file sizes about 8% smaller or PSNRs around 0. 5dB higher. More importantly, this enhancement can be obtained through a process that not only ensures backwards compatibility, producing a JPEG compliant bitstream, but also can be performed in near real-time in hardware due to the low number of required JPEG evaluations.

Furthermore, the support for near real-time implementations can be extended by appropriately tuning the population size and the number of generations for the PSO or the maximum number of iterations and functions evaluations for the DSA according to the available hardware capabilities. The expected behaviour of the compression ration with respect to the number JPEG of evaluations is described in the fig. 3. For the PSO, the number of JPEG evaluation is given by the product of the population size and the number of generations.

Another noteworthy observation is that our new metric, FQ-ERQ, performs, at average, no worse than the classical Lagrangian Cost Function, while demands far less hyper parameter tuning and adjusts itself automatically to any given image and quality factor, hence being a appropriate option for an automatic nature inspired JPEG encoder.

V. Conclusions and Future Work

This work presents a framework for generating custom, image-specific, rate-distortion optimized quantization tables in an automatic fashion, producing JPEG compliant images that does not require any change in deployed decoders and outperform the vanilla JPEG using standard quantization tables by about a 9% in terms of compression rate. The complexity of the proposed algorithm is relatively low, and can be further reduced depending on application requirements.

In terms of future work, a lot of research directions can be explored, but extending the nature inspired metaheuristic to jointly optimize both the applied transform and the choice of the quantization table seems promising. Similarly, there is a lot of room for improvements in the application of nature inspired metaheuristics for real-time or near real-time JPEG encoding. Further research could also investigate the design of new rate-distortion metrics for nature inspired quantization table optimization, since our results suggests that an appropriate choice for the fitness function may play a role in the obtained quantization tables, speeding up the convergence of the algorithms.

References

[1] ITU-T.81, "Information technology –digital compression and coding of continuous-tone still images: – requirements and guidelines."

[2] A. Skodras, C. Christopoulos, and T. Ebrahimi, "The JPEG 2000 still image compression standard," *IEEE Signal Processing Magazine*, vol. 18, no. 5, pp. 36–58, 2001. [Online]. Available: https://doi.org/10.1109/79.952804

[3] G. Hudson, A. Leger, B. Niss, and I. Sebestyen, "JPEG at 25: Still going strong," *IEEE MultiMedia*, vol. 24, no. 2, pp. 96–103, Apr. 2017.

[4] G. Hudson, A. Léger, B. Niss, I. Sebestyén, and J. Vaaben, "JPEG-1 standard 25 years: past, present, and future reasons for a success," *Journal of Electronic Imaging*, vol. 27, no. 04, p. 1, Aug. 2018.

[5] M. Elbadri, R. Peterkin, V. Groza, D. Ionescu, and A. E. Saddik, "Hardware support of JPEG," in *Canadian Conference on Electrical and Computer Engineering, 2005*. IEEE.

[6] J. Chao, H. Chen, and E. Steinbach, "On the design of a novel JPEG quantization table for improved feature detection performance," in *2013 IEEE International Conference on Image Processing*. IEEE, Sep. 2013.

[7] D. Edmundson and G. Edmundson, "Exploiting JPEG compression for image retrieval," in *2012 IEEE International Symposium on Multimedia*. IEEE, Dec. 2012.

[8] D. Onnasch, G. Prause, and A. Ploger, "Quantization table design for JPEG compression of angiocardiographic images," in *Computers in Cardiology 1994*. IEEE Comput. Soc. Press. [Online]. Available: https://doi.org/10.1109/cic.1994.470199

[9] Y. Jiang and M. S. Pattichis, "JPEG image compression using quantization table optimization based on perceptual image quality assessment," in *2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*. IEEE, Nov. 2011.

[10] S. N. and, "A literature review on quantization table design for the JPEG baseline algorithm," *International Journal Of Engineering And Computer Science*, Aug. 2016.

[11] S. Viswajaa, V. Balasubramanian, and G. Karpagam, "A survey on nature inspired meta-heuristics algorithms in optimizing the quantization table for the jpeg baseline algorithm," *International Advanced Research Journal in Science, Engineering and Technology*, vol. 2, pp. 114–123, 01 2015.

[12] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*. IEEE. [Online]. Available: https://doi.org/10.1109/icnn.1995.488968

[13] Y. Xiang, D. Sun, W. Fan, and X. Gong, "Generalized simulated annealing algorithm and its application to the thomson model," *Physics Letters A*, vol. 233, no. 3, pp. 216–220, Aug. 1997.

[14] S. Wu and A. Gersho, "Rate-constrained picture-adaptive quantization for JPEG baseline coders," in *IEEE International Conference on Acoustics Speech and Signal Processing*. IEEE, 1993.

[15] K. Ramchandran and M. Vetterli, "Rate-distortion optimal fast thresholding with complete JPEG/MPEG decoder compatibility," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 700–704, 1994.

[16] X. Li and J. Wang, "A steganographic method based upon jpeg and particle swarm optimization algorithm," *Inf. Sci.*, vol. 177, pp. 3099–3109, 08 2007.

[17] V. Snasel, P. Kromer, J. Safarik, and J. Platos, "JPEG steganography with particle swarm optimization accelerated by AVX," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 8, Jul. 2019.

[18] S. Fazli and M. Kiamini, "A high_performance steganographic method using JPEG and PSO algorithm," in *2008 IEEE International Multitopic Conference*. IEEE, Dec. 2008.

[19] F. H. Rabevohitra and J. Sang, "High capacity steganographic scheme for JPEG compression using particle swarm optimization," *Advanced Materials Research*, vol. 433-440, pp. 5118–5122, Jan. 2012.

[20] H. Ma and Q. Zhang, "Research on cultural-based multi-objective particle swarm optimization in image compression quality assessment," *Optik*, vol. 124, no. 10, pp. 957–961, May 2013.

[21] A. S. Abbood, "Design of jpeg image compression scheme with a particle swarm optimization-based quantization table," *International journal of innovation and scientific research*, vol. 25, pp. 22–31, 2016.

[22] M. Hopkins, M. Mitzenmacher, and S. Wagner-Carena, "Simulated annealing for jpeg quantization," 2017.

[23] H. Wang and S. Kwong, "Preferential multi-objective genetic algorithm for jpeg quantization table optimization," *Dynamics of Continuous, Discrete and Impulsive Systems, Series B: Applications and Algorithms 2 (Special Issue)*, pp. 608–613, 07 2005.

[24] Y. Zhang, S. Wang, and G. Ji, "A comprehensive survey on particle swarm optimization algorithm and its applications," *Mathematical Problems in Engineering*, vol. 2015, pp. 1–38, 2015.

[25] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, Aug. 2007.

[26] C. Tsallis, "Possible generalization of boltzmann-gibbs statistics," *Journal of Statistical Physics*, vol. 52, no. 1-2, pp. 479–487, Jul. 1988.

[27] C. Tsallis and D. A. Stariolo, "Generalized simulated annealing," *Physica A: Statistical Mechanics and its Applications*, vol. 233, no. 1-2, pp. 395–406, Nov. 1996.

[28] Y. Xiang and X. G. Gong, "Efficiency of generalized simulated annealing," *Physical Review E*, vol. 62, no. 3, pp. 4473–4476, Sep. 2000.

[29] K. M. Mullen, "Continuous global optimization inR," *Journal of Statistical Software*, vol. 60, no. 6, 2014.

[30] S. B. Xiang Y., Gubian S. and H. J., "Generalized simulated annealing for efficient global optimization: the gensa package for r." *The R Journal*, vol. 5, no. 1, 2013.

[31] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizeau, and C. Gagné, "DEAP: Evolutionary algorithms made easy," *Journal of Machine Learning Research*, vol. 13, pp. 2171–2175, jul 2012.

[32] P. Virtanen, , R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, and P. van Mulbregt, "SciPy 1.0: fundamental algorithms for scientific computing in python," *Nature Methods*, vol. 17, no. 3, pp. 261–272, Feb. 2020.

[33] I. Group, "The independent jpeg group's jpeg software," 03 2004.

[34] G. Bjøntegaard, "Improvements of the BD-PSNR model," ITU-T SG16/Q6, 35th VCEG Meeting, Doc.VCEG-AI11, 2008.