Acta Scientifica Malaysia (ASM)

DOI: http://doi.org/10.26480/asm.02.2021.85.91

ISSN: 2521-5051 (Print)
ISSN: 2521-506X (Online)
CODEN: ASMCCQ

RESEARCH ARTICLE

EXAMINATION OF PNG IMAGE OF CORONA VIRUS BASED ON BENFORD'S LAW

M.Z.A.M. Jaffar[a]*, A.H.H. Hamid[a], L.F.L. Joey[b]

[a]Institute For Mathematical Research, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia.
[b]Department of Mathematical and Statistics, Faculty Of Science, 43400 Serdang, Selangor, Malaysia.
[*]Corresponding Author Email: maizurwatul@upm.edu.my

ARTICLE DETAILS

Article History:

Received 19 October 2021
Accepted 23 November 2021
Available online 29 November 2021

ABSTRACT

The global crisis caused by the Corona virus is making everyone search for new ways to eliminate this danger that threatens people's lives. The addition of new methods for detecting the infection of people makes the prevention process fast and not expensive at all. Waiting hours or days for people's test results is both expensive and dangerous. The use of the JPEG format for images (the most common in our daily lives) and Benford's law when they work together makes this possible. This method has been validated by the discovery of digital image manipulation and its effective use in forensic evidence and is known to be highly sensitive. Moreover, the availability of high-resolution microscopes with optical zoom made it possible to take real and clear images of the Corona virus, and therefore the use of this method saves time, effort and risk.

KEYWORDS

Corona virus, PNG, Benford's law

1. INTRODUCTION

As people continue to get infected with the Coronavirus (Covid-19) and it turns into a global pandemic, we have witnessed things we did not know before, such as the closure of the year and the interruption of public life. Scientists were able to take real pictures of the Coronavirus using a high-resolution electron microscope. We took these images in different formats and analyzed them, for example, images (PNG) and images (JPEG). Fortunately, there are several ways to compress images that are available today. These fall into two general categories: Compression of images without loss and loss. The JPEG process is a widely used form of lossy image compression centered on discrete cosine transform, which converts images from the spatial domain to the frequency domain. DCT technology works by separating images into parts with different frequencies. During a step called quantization, in which part of the compression occurs, less significant frequencies are ignored, hence the term "loss". After that, only the most important remaining frequencies are used to restore the image in the decompression process. As a result, the reconstructed images contain some distortion, but as we will soon see, these distortion levels can be adjusted during the compression phase. The JPEG method is used for both color and black and white images, but the focus will be on analyzing and comparing coronavirus images in different formats using Benford's law.

Many of us are familiar with Benford's law, and we've used it to evaluate data and verify it hasn't been tampered with. In 1938, Frank Benford proposed Benford's law (Benford, 1938). It's also known as the rule of first numbers or the law of 'odd numbers,' because it only takes the most significant number (MSD) into account (Benford, 1938). By MSD, we mean that 0.567 equals 5 and that 5000 equals 5 as well. The probability distribution of the first number x (x = 1, 2, 9) in a collection of natural numbers is logarithmic, according to this rule (Shi, 2013). When studying the probability distribution of the first number from 1 to 9 for a set of normal data, Hill characterized it as a logarithmic distribution (Hill, 1996).

It should be emphasized that naturally created data is expected to follow this law, whereas modified data or random guesses are expected to break it (Fu et al., 2007). Many applications of Benford's rule have been proven in the forensic literature, such as financial statements (Nigrini, 1996). This law affects the image's gradient as well as the Laplacian pyramid code (Jolian, 2001). A group researcher demonstrated that images in the pixel field do not follow Benford's law, but that images converted to discrete cosine transform (DCT) do (Jolian, 2001). This is a significant breakthrough in the application of Benford's law. They also used Benford's generalized law to expose data that was buried naturally (Gonzalez et al., 2007). A group researcher applied this legislation (Fu et al., 2007).

2. MATERIALS AND METHODS

A general description of the JPEG process is given below. Later, we'll take the reader on a full tour of JPEG's approach so that they may have a better grasp of the process, and how these results are used to apply Benford's law to them.

1. The image is divided into 8x8 pixel pieces.

2. The DCT is applied to each block from left to right, top to bottom.

3. Quantization is used to compress each block.

4. The image's compressed block array is saved in a fraction of the space it would otherwise take up.

5. We use a method known as (Zig-Zag) to convert the image to binary data (0,1), dear reader, the results of this step we used to apply Benford's law as will be shown later.

6. If required, the picture is rebuilt using decompression, which is a technique that takes advantage of the Transform of the Inverse Discrete Cosine (IDCT).

Quick Response Code	Access this article online	
	Website: www.actascientificamalaysia.com	**DOI:** 10.26480/asm.02.2021.85.91

Cite The Article: M.Z.A.M. Jaffar, A.H.H. Hamid, L.F.L. Joey (2021). Examination of PNG Image of Corona Virus Based on Benford's Law. *Acta Scientifica Malaysia, 5(2): 85-91.*

7. Use Benford's law to analyze images results.

2.2 The DCT Formula

$$f_{x,y} = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7}\alpha(u)\alpha(v)F_{u,v}\cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right] \quad (8)$$

- X is the pixel row, for the integers 0≤x<8.
- y is the pixel row, for the integers 0≤y<8.
- ⍰(u) is defined as about, for the integers 0≤u<8.
- F u,v is the reconstructed approximate coefficient at coordinates (u, v).
- f x,y is the reconstructed pixel value at ordinates (x, y).

The DCT equation calculates the u,vth entry of an image's DCT. The x,yth element of the picture represented by the matrix p is p(x,y). The size of the block on which the DCT is performed is N. From the pixel values of the original image matrix, the equation calculates one entry (u,vth) of the converted picture. N = 8 and many ranges from 0 to 7 for the typical 8x8 block used by JPEG compression. As a result, D(i,j) would be the same as in Equation. The resultant matrix is dependent on the horizontal, diagonal, and vertical frequencies since the DCT employs cosine functions. As a consequence, a black picture with a lot of frequency change has a very random appearing resultant matrix, but a single-color image matrix has a resulting matrix with a big value for the first element and zeroes for the remaining elements.

2.3 The Matrix of DCT

Before we continue, dear reader, I want to remind you of what we said in the introduction that we have images of the Coronavirus in PNG and other images in JPEG, as well as one unit in color and another in black and white as shown below:

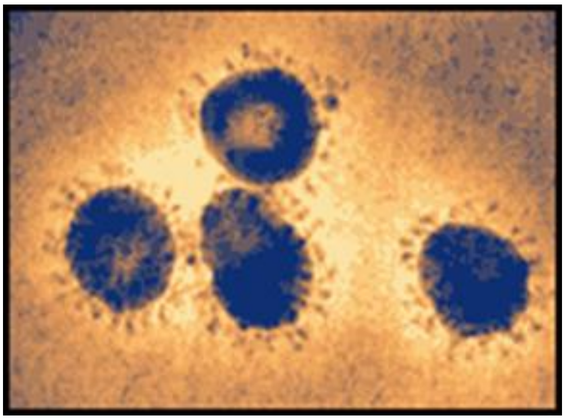


Image 1.PNG (https://www.cdc.gov/sars/lab/images.html.)

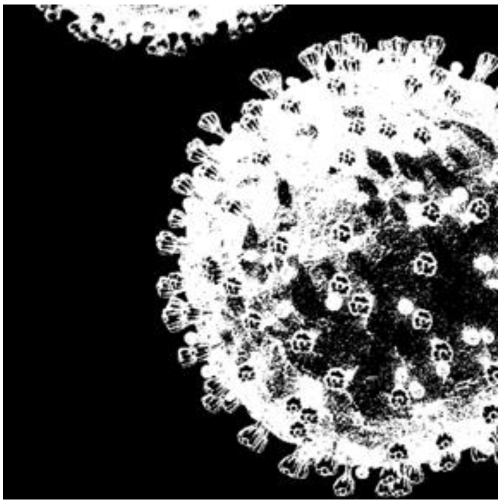


Image 2.PNG (https://commons.wikimedia.org/wiki/File:SARS-CoV-2_(2).png.)

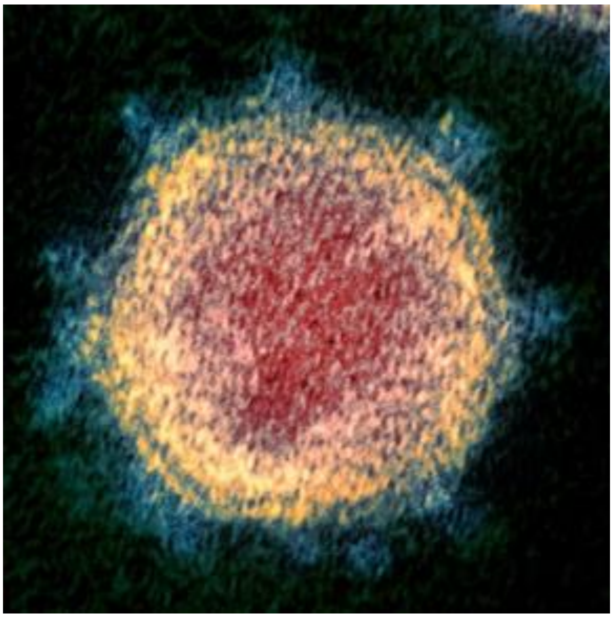


Image 3.JPEG (https://www.nih.gov/news-events/nih-research-matters/novel-coronavirus-structure-reveals-targets-vaccines-treatments)

2.4 Using an 8x8 block to do the DCT

First of all, keep in mind that the pixel values of a black-and-white image vary from 0 to 255 in increments of 1, with 0 representing pure black and 255 representing pure white. As can be seen, these 256 shades of gray may properly depict a photo, drawing, or another image (Angelopoulos, 2004). Because a picture is made up of hundreds or even thousands of 8x8 pixel blocks, the following explanation of what happens to one of them represents a microcosm of the JPEG process:

What is done to one block of picture pixels is applied to all of them in the same sequence.

Let's begin with a set of picture pixel values. This particular block was picked from an image's very bottom right-hand corner.

*2.4.1 Original 8*8 M=*

170	170	170	159	170	159	170	159
170	170	170	159	170	170	148	148
170	170	170	170	170	170	148	148
170	170	159	170	170	170	159	159
170	170	159	159	159	159	159	159
159	159	170	170	159	159	159	148
148	159	170	159	159	159	170	148
148	148	159	170	159	159	159	135

The original block is "leveled off" by removing 128 from each entry since the DCT is intended to deal with pixel values ranging from -128 to 127. As a consequence, the matrix below appears.

42	42	42	31	42	31	42	31
42	42	42	31	42	42	20	20
42	42	42	42	42	42	20	20
42	42	31	42	42	42	31	31
42	42	31	31	31	31	31	31
31	31	42	42	31	31	31	20
20	31	42	31	31	31	42	20
20	20	31	42	31	31	31	7

We begin by converting the picture from its spatial dimension to its frequency dimension using the discrete cosine equation. This block matrix now has 64 DCT coefficients, cy, where I and j are numbers between 0 and

7. The top-left coefficient, C00, corresponds to the original picture block's low frequencies. The DCT coefficients correlate to higher and higher frequencies of the picture block as we go away from C00 in all directions, with C77 corresponding to the highest frequency. It's worth noting that the human eye is particularly sensitive to low frequencies, which will be reflected in the quantization findings, the matrix below appears:

271.13	26.139	-23.06	9.5136	-9.875	3.0494	-1.135	2.3956
24.834	13.274	11.066	8.9424	6.6569	-17.23	5.9292	1.7442
-10.8	-5.379	-9.413	-2.613	-8.693	7.421	-2.288	10.993
3.128	-4.373	14.939	-8.729	-3.832	2.5624	-4.496	8.9781
1.125	-6.955	9.5278	3.2502	6.625	2.1717	-8.682	-1.383
9.1061	-8.998	1.3392	-4.077	-3.529	2.0613	-5.536	4.092
-0.263	4.3276	-7.788	-2.496	4.8184	6.758	0.6631	-2.358
-1.008	0.6051	7.382	-2.233	-3.462	-2.038	-1.856	3.393

2.4.2 Quantization

Our 8x8 DCT coefficient block is now ready for quantization compression. Variable degrees of picture compression and quality may be achieved at this phase by selecting particular quantization matrices, which is a unique and extremely valuable aspect of the JPEG process. This allows the user to choose from a range of quality levels ranging from 1 to 100, with 1 being the worst picture quality and most compression and 100 representing the best image quality and lowest compression. As a consequence, the quality/compression ratio may be customized to meet specific requirements. The JPEG standard quantization matrix is the outcome of subjective research using the human visual system. This matrix, which has a quality rating of 50, provides both great compression and good decompression image quality (Talukder et al., 2010).

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

However, scalar multiples of the JPEG standard quantization matrix can be utilized if a higher degree of quality and compression is required. The typical quantization matrix is multiplied by (100-quality level)/50 for a quality level larger than 50 (less compression, higher picture quality). The typical quantization matrix is multiplied by 50/quality level for a quality level less than 50 (more compression, poorer picture quality). The rounded and clipped scaled quantization matrix now has positive integer values ranging from 1 to 255. The following quantization matrices, for example, produce quality levels of 10 and 90 (Clausi, 2002).

$$Q_{10} = \begin{bmatrix} 80 & 60 & 50 & 80 & 120 & 200 & 255 & 255 \\ 55 & 60 & 70 & 95 & 130 & 255 & 255 & 255 \\ 70 & 65 & 80 & 120 & 200 & 255 & 255 & 255 \\ 70 & 85 & 110 & 145 & 255 & 255 & 255 & 255 \\ 90 & 110 & 185 & 255 & 255 & 255 & 255 & 255 \\ 120 & 175 & 255 & 255 & 255 & 255 & 255 & 255 \\ 245 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \\ 255 & 255 & 255 & 255 & 255 & 255 & 255 & 255 \end{bmatrix}$$

3. RESULTS AND DISCUSSION

$$Q_{90} = \begin{bmatrix} 3 & 2 & 2 & 3 & 5 & 8 & 10 & 12 \\ 2 & 2 & 3 & 4 & 5 & 12 & 12 & 11 \\ 3 & 3 & 3 & 5 & 8 & 11 & 14 & 11 \\ 3 & 3 & 4 & 6 & 10 & 17 & 16 & 12 \\ 4 & 4 & 7 & 11 & 14 & 22 & 21 & 15 \\ 5 & 7 & 11 & 13 & 16 & 12 & 23 & 18 \\ 10 & 13 & 16 & 17 & 21 & 24 & 24 & 21 \\ 14 & 18 & 19 & 20 & 22 & 20 & 20 & 20 \end{bmatrix}$$

The converted image matrix M is quantized by dividing each element by the corresponding element in the quantization matrix, then rounding to the nearest integer value. The quantization matrix Q50 is utilized in the next step.

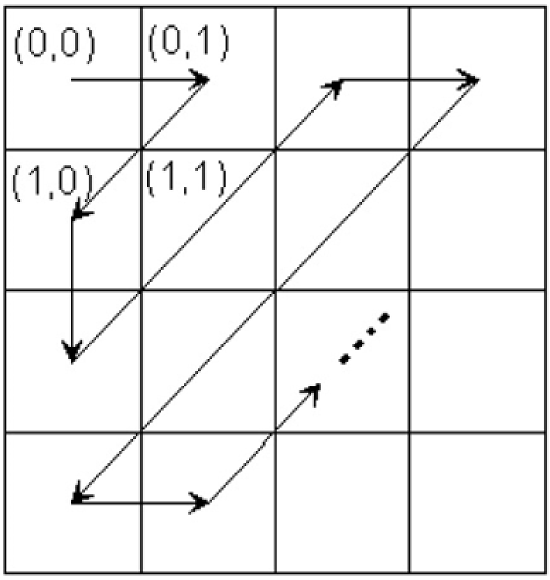
$$C_{u,v} = \text{round}\left(\frac{M_{u,v}}{Q50}\right)$$

17	2	-2	1	0	0	0	0
2	1	1	0	0	0	0	0
-1	0	-1	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Remember that the coefficients in the upper-left corner correspond to the picture block's lower frequencies, to which the human eye is most sensitive. Furthermore, the zeros indicate the higher frequencies that have been rejected as less significant, resulting in the lossy component of compression. Only the remaining nonzero coefficients will be utilized to rebuild the picture, as previously stated. It's also worth noting the impact of different quantization matrices; using Q10 would result in substantially more zeros for C, whilst using Q90 would result in relatively few.

3.1 Coding

The quantized matrix C is now ready for compression's last phase. Before being stored, an encoder converts all of C's coefficients into a stream of binary data (01101011). This article does not go into great detail on the coding process. We can, however, highlight one crucial feature that the reader will undoubtedly like. It is fairly typical for the majority of the coefficients to equal 0 after quantization. JPEG takes use of this by storing quantized coefficients in Figure 1's zig-zag sequence. The benefit comes from the aggregation of relatively long sequences of zeros that compress nicely. The 4x4 sequence seen in Figure 1 continues throughout the whole 8x8 block (Wallace, 1992).



This is the last step in the process of compressing an image, converting it from PNG format to JPEG format, and storing it inside your computer. Now, if we want to display the image again, what do we do?

3.2 Decompression

Decoding the bitstream representing the quantized matrix C is the first step in reconstructing our image. The appropriate element of the quantization matrix initially employed is multiplied by each element of C.

Ri,j = Qi,j × Ci.j

R=

272	22	-20	16	0	0	0	0
24	12	14	0	0	0	0	0
-14	0	-16	0	0	0	0	0
0	0	22	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Matrix R is next subjected to the IDCT, which is rounded to the nearest integer. Finally, each element of that result is added to 128 produce the decompressed JPEG version N of our original 8x8 picture block M.

N = IDCT + 128

3.3 Matrices are compared

Let's check how our initial pixel block compares in JPEG format.

Original 8*8 M=

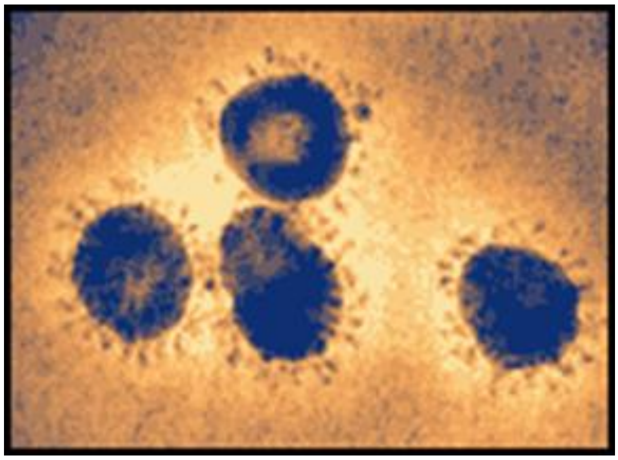
170	170	170	159	170	159	170	159
170	170	170	159	170	170	148	148
170	170	170	170	170	170	148	148
170	170	159	170	170	170	159	159
170	170	159	159	159	159	159	159
159	159	170	170	159	159	159	148
148	159	170	159	159	159	170	148
148	148	159	170	159	159	159	135

N 8*8=

174	169	165	163	163	163	159	156
170	168	167	167	168	165	159	153
168	167	168	170	171	167	159	152
170	167	166	166	168	166	161	156
171	167	162	160	162	163	162	160
166	162	158	158	160	162	160	157
155	155	158	163	165	162	154	147
145	150	159	168	171	164	149	138

Given that around 70% of the DCT parameters were lost before the picture block was decompressed/reconstructed, this is a remarkable result. It should come as no surprise that a black-and-white JPEG image recalls that there are 256 potential shades of gray in an image, and the difference, say, 10, is hardly noticeable to the human eye.

On a picture, we can do DCT and quantization (Image 1.PNG). To demonstrate the changes that occur, we display what occurred to the image from the previous phases.



The DCT multiplies each block 8*8. After that, each element in each block of the image is quantized using a quality level 50 quantization matrix. Many of the components are wiped out at this point, and the image takes up considerably less storage space. yielding the picture displayed in Figure 1.

Figure (1)

The inverse discrete cosine transform may now be used to decompress the picture. To achieve high compression, we'll transform the image to black and white at quality level 50. The quality declines dramatically at lower quality levels, but the pressure does not. as seen in Figure (2)

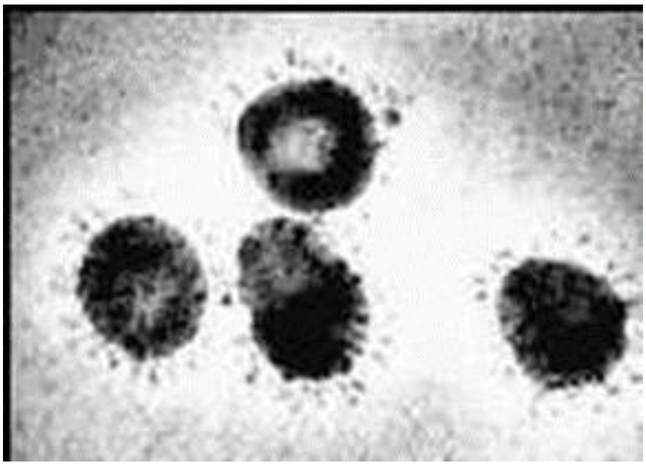


Figure (2)

Image 3.JPEG [9.1]
W&B
Quality 50 – 84%
Zeros

3.4 More Examples

We can observe the effects of compression on other photos. Photos having a lot of high frequencies, or images with a lot of contrast, do not compress as efficiently as smooth, low-frequency images.

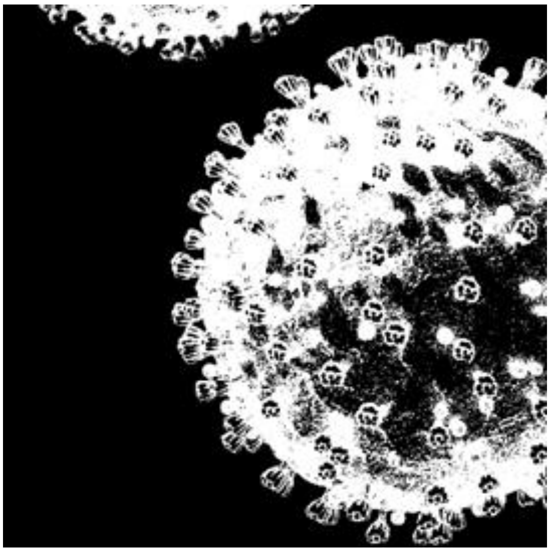


Image 2.PNG [10]

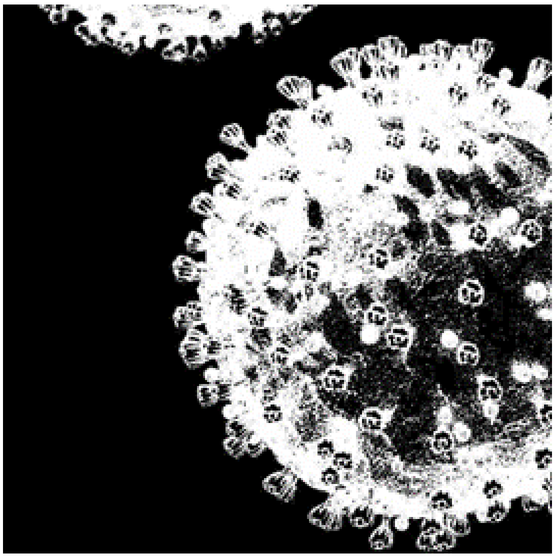


Image 2.JPEG [10.1]

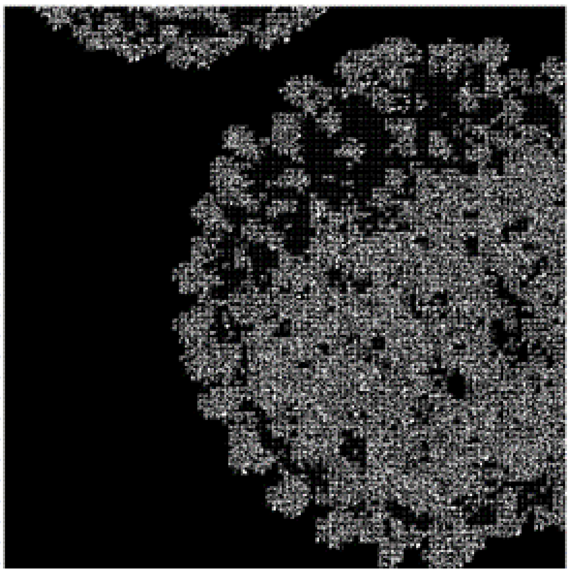


Image 2 is made up of 8 * 8 matrices

In this last image, it's different, because it's a color JPEG, and here we've recompressed it and converted it to black and white. In fact, we analyzed the results of this image to compare it with the results of other images of different formats.

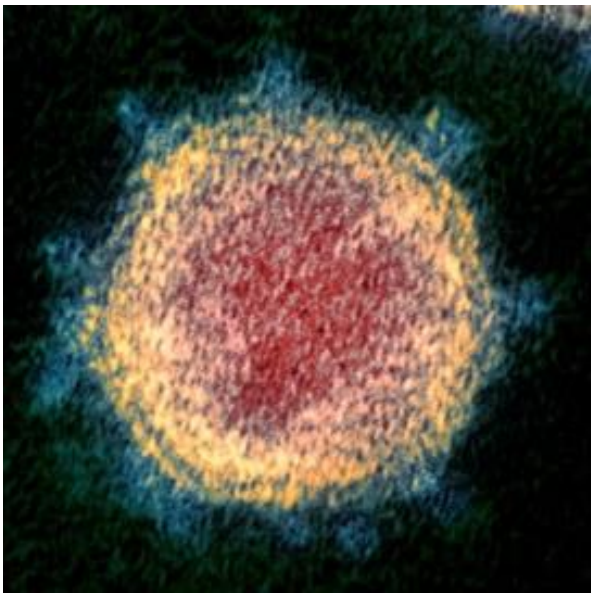


Image 3.JPEG [11]

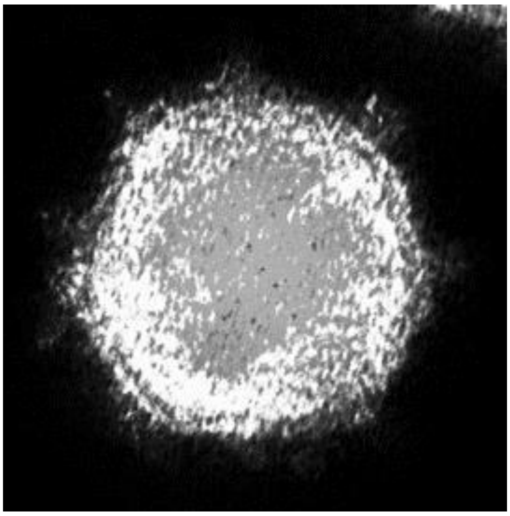


Image 3.JPEG [11.1] W&B

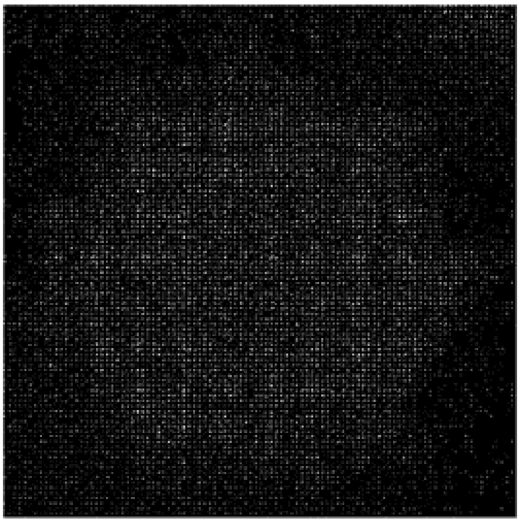


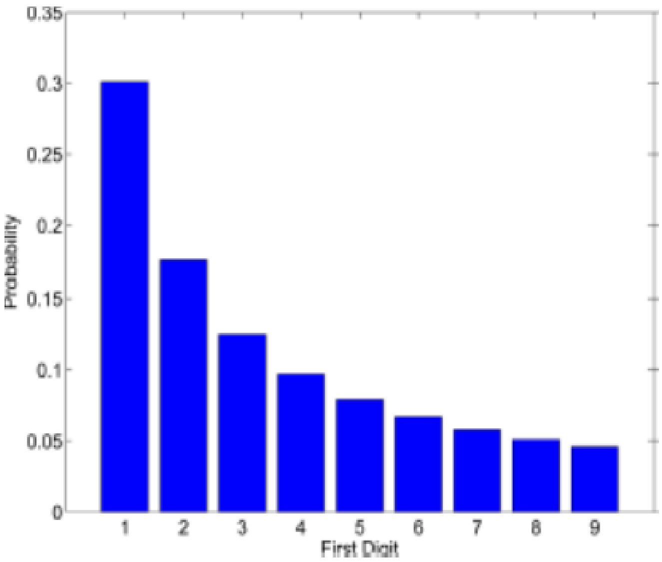
Image 3 is made up of 8 * 8 matrices

Following the compression and conversion of pictures to JPEG format in black and white, we go on to the analysis procedure, which employs Benford's law, as described in the introduction.

3.5 Benford's Law and Block-DCT Transactions: An Overview:

Figure 1 depicts the usual distribution of Benford's law. As a result, any data that closely resembles this pattern follows Benford's law.

When the most significant number (MSD) is taken into account, and the data sets meet Benford's rule, the law may be stated using equation (Taylor, 2019).



$$p(x)=\log_{10}\left(1+\frac{1}{x}\right),(x=1,2,\cdots,9)$$

where x is the number's initial digit and (px) is the probability distribution of x. The equation defines the generalized Benford's law, which was reported to closely follow a logarithmic law (Makrushin, 2018).

3.6 Block-DCT Transactions

For JPEG image compression, Benford's law has been extensively researched (Makrushin, 2018). A signal or an image is converted from the spatial domain to the frequency domain using a DCT. However, it must be emphasized that direct translation of a two-dimensional spatial function f (x,y) into a two-dimensional spatial frequency spectrum F(u,v) and vice versa does not result in signal or image loss of any information (Al-Bandawi and Deng, 2018). Since the images include 2D signals, a 2D DCT is used during their analysis. Compression of JPEG images is based on block-DCT, and has an 8*8 Block-DCT, quantum, and entropy coding (Cogranne, 2018). Based on the DCT, the probability distribution of the prime numbers was obtained. The first step is to split the original uncompressed image into 8 * 8-pixel non-overlapping parts. Block-DCT transactions are generated as a result of this technology. To convert each block into a frequency space, a 2D DCT is applied to it. We referred to this step earlier in the Zig-Zag process, the value in the upper left corner is referred to as the DC coefficient, while the remaining 63 values are referred to as the AC coefficients. As a result, the table of quantities is applied to each block of DCT transactions (Cogranne, 2018). JPEG parameters are generated as a result of this action. As a result, we are curious about the probability distribution of the first numbers of the AC component of the JPEG coefficients that do not include the DC coefficient.

4. Results and Discussion

The experiment aims to check the JPEG parameters of images taken with the coronavirus and determine whether these images are real or whether it is possible to determine infection with the virus from image analysis, as well as whether it is possible to distinguish between coronavirus and another virus of the same species by knowing and determine whether it follows Benford's generalized law; Image results were compared in different formats.

Now, the results of applying Benford's law to the all image

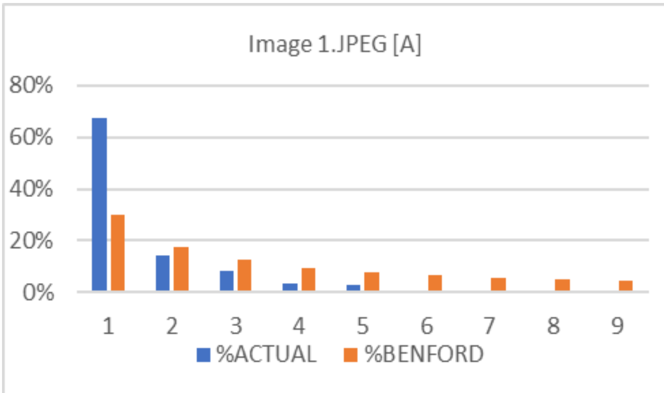


Image 1.JPEG [9.1]

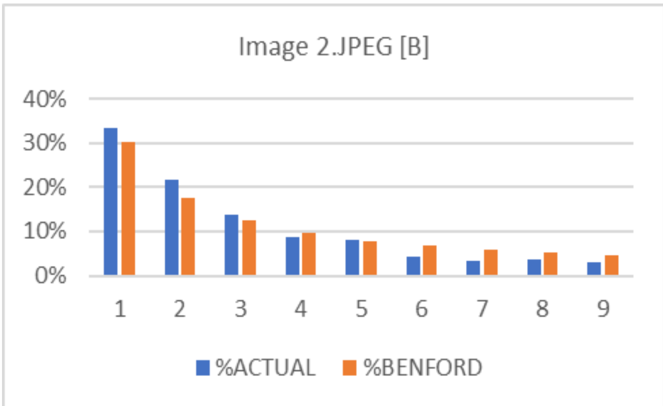


Image 2.JPEG [10.1]

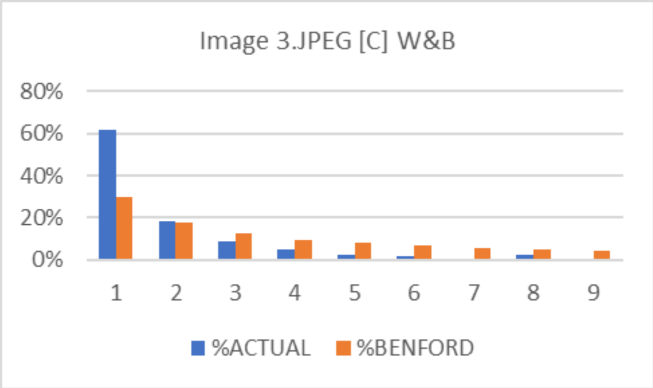


Image 3.JPEG [11.1] W&B

By observing the results, it becomes clear to you, dear reader, that the image closest to the real virus is the second image in Jart's address. However, through my review of the rest of the images and their credibility, I found that other factors effectively influence the results, which are all of the following:

Firstly, the image resolution is affected by the results. If you want to get accurate results, the number of pixels in the image must be a multiple of 8. We explain why, in the origin of the JPEG compression method, the image must be divided into 8 * 8. If the image pixel also does not meet this condition, Matlab will automatically By adding or subtracting some numbers to the main matrix of the image until it meets the condition of 8 * 8 blocks for compression. Undoubtedly, this is an obvious manipulation.

Secondly, if the captured image is in grayscale (black and white), then it will be better in terms of analysis and matching of results. The explanation for this is that many algorithms convert images from color to black and white, and these different methods lead to different results.

Thirdly, compressing the same JPEG image twice (double compression) reduces the size but distorts the original image and cannot be effectively verified. In addition, the 50% quantization matrix is better than the rest in terms of maintaining the image resolution on the one hand and the image size on the other.

5. Conclusion

A worldwide epidemic has paralyzed humanity, bringing people's everyday lives to a standstill. In this paper, we have discussed different images of Coronavirus in different shapes and colors and in black and white. We have suggested using the Benford Law Difference Scale to help separate three databases. We have shown that features of Benford's law can reveal the true form of the virus. In our future work, if scientists can take a real picture of the virus according to the conditions we mentioned, we will be able to identify infected people at a very high speed of no more than 60 seconds by applying the method mentioned in an article, surely this saves time, effort and risk as well. I think extending the search in this area to find the best pixels an image can be that satisfy Benford's law requirement and also the best quantization matrix that can satisfy the condition with one more thing added is to check the best algorithm for converting color images to black and white all of these things will make this possible in the future. Furthermore, lossless pressure must be thoroughly investigated. This will be discussed in more detail in the future. Finally, we'll look into full-color depth raw data processing and how it affects DCT parameters after JPEG conversion.

References

Al-Bandawi, H., Deng, G., 2018. Blind image quality assessment based on Benford's law. IET Image Processing, 12 (11), Pp. 1983-1993.

Angelopoulos, C., 2004. Digital panoramic radiography: An overview. In: Seminars in Orthodontics. WB Saunders, Pp. 194-203.

Benford, F., 1938. The Law of Anomalous Numbers. Proceedings of the American Philosophical Society, 78, Pp. 551-572.

Bhandari, A.K., Kumar, A., Padhy, P.K., 2011. Enhancement of low contrast satellite images using discrete cosine transform and singular value decomposition. World Academy of Science, Engineering and Technology, 79, Pp. 35-41.

Clausi, D.A., 2002. An analysis of co-occurrence texture statistics as a function of grey level quantization. Canadian Journal of remote sensing, 28 (1), Pp. 45-62.

Cogranne, R., 2018. Determining JPEG image standard quality factor from the quantization tables. arXiv preprint arXiv:1802.00992.

Fu, D., Shi, Y.Q. and Su, W., 2007. A Generalized Benford's Law for JPEG Coefficients and Its Applications in Image Forensics. Proceedings of SPIE, 6506, Pp. 1L1- 1L11.

Gonzalez, F.P., Heileman, G.L., and Abdallah, C.T., 2007. Benford's Law in Image Processing. IEEE International Conference on Image Processing, San Antonio, 16- 19, Pp. 405-408.

Hill, T.P., 1996. A Statistical Derivation of the Significant-Digit Law. Statistical Science, 10, Pp. 354-363. COGRANNE, Rémi. Determining JPEG image standard quality factor from the quantization tables. arXiv preprint arXiv:1802.00992, 2018. https://doi.org/10.1214/ss/1177009869

https://commons.wikimedia.org/wiki/File:SARS-CoV-2_(2).png.

https://www.cdc.gov/sars/lab/images.html.

https://www.nih.gov/news-events/nih-research-matters/novel-coronavirus-structure-reveals-targets-vaccines-treatments.

Jolion, J.M., 2001. Images, and Benford's Law. Journal of Mathematical Imaging and Vision, 14, Pp. 73-81. https://doi.org/10.1023/A:1008363415314

Makrushin, A., 2018. Generalized Benford's law for blind detection of morphed face images. In: Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security, Pp. 49-54.

Nigrini, M., 1996. A Taxpayer Compliance Application of Benford's Law. The Journal of the American Taxation Association, 18, Pp. 72-91.

Shi, Y.Q., 2013. Image Statistical Modeling and Its Applications to Information Forensics. New Jersey Institute of Technology.

Talukder, Kamrul, H., Harada, K., 2010. Haar wavelet-based approach for image compression and quality assessment of compressed image. arXiv preprint arXiv:1010.4084.

Taylor, Z., 2019. Pathways to legitimacy. Planning Theory, 18 (2), Pp. 214-236.

Wallace, G.K., 1992. The JPEG still picture compression standard. IEEE transactions on consumer electronics, 38, Pp. 1: xviii-xxxiv.