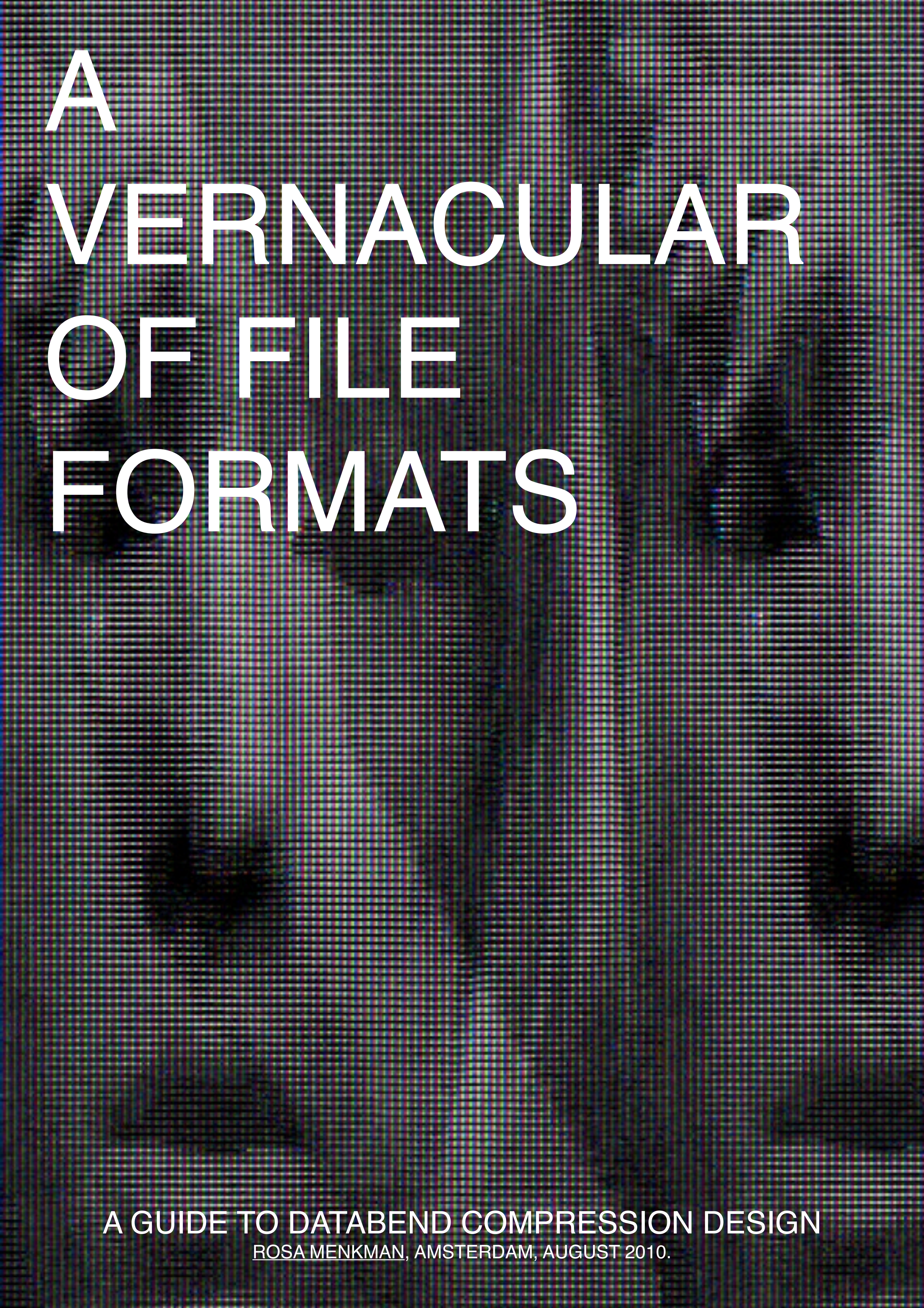


A VERNACULAR OF FILE FORMATS

A GUIDE TO DATABEND COMPRESSION DESIGN

ROSA MENKMAN, AMSTERDAM, AUGUST 2010.

Tango between a corrupt format and its user.

There is not sufficient data. Please enter data

You have too much character! Apply gaussian blur.

You still have quirks! Clone stamp for smooth surface

Your file has invalid markers. Enter new markers

Your dimensions do not correspond. Change dimensions

ERROR

Goto data therapy and repair your registry

Your keyframes are missing. Your codecs are not supported.

Noise Art > Filter art > when Cool becomes Hot >>>>>>>

Glitches are hot; proof can be found on MTV, Flickr, in the club and in the bookstore. While the coffee table book *Glitch: Designing Imperfection* (2009) has introduced the glitch design aesthetic to the world of latte drinking designers, and Kanye West used glitches to sing about his imperfect love life, the awkward, shy and physically ugly celebrate under the header "Glitched: Nerdcore for life".
Glitch has transformed from cool to hot. Its no more then a brightly colored bubblegum wrapper that doesn't ask for any involvement, or offers any stimulus. Inside I find gum that I keep chewing, hoping for some new explosion of good taste. But the more I chew, the less tasty / rubbery it gets. Glitch design fulfills an average, imperfect stereotype, a filter or commodity that echoes a "medium is the message" standard.

Luckily (or naturally?), the "No Content - Just Imperfection" slogan of this kind of hot glitch design is complimented by cool glitches. In *The Laws of Cool* (2004), Alan Liu asks himself What is "Cool"? He describes that cool is the ellipsis of knowing whats cool ...and withholding that idea. Those that insist on asking, are definitely uncool.

Unruly and defiant as it thus is, I will give my take on cool glitches. Cool glitches are the glitches that do not just focus on a static end product, but (also) on a process, a personal exploration or a narrative element (that often reflects critically on a medium).

Cool is in a constant state of flux, as is the *genre* of "cool glitch art", which finally exists as an assemblage that relies on first of all the construction, operation and content of the apparatus (the medium) and secondly the work, the writer/artist, and the interpretation by the reader and/or user (the meaning). In the end, there is “no one definition of cool” - glitch art.

In an effort to make what was once cool now hot, and visa versa, and to take what happened in the Designing Imperfection book a step further, I made this Vernacular of File Formats, in which I study ways to exploit and deconstruct the organizations of file formats into new, brutalist designs.

…I am waiting for the first "Glitchs not dead" hoodie in H&M. And because *fans are as bad as the ignorant*, for the sake of being bad, I will definitely wear the hoodie. Hoera!

Contents

Static
- Uncompressed
- Lossless
- Lossy

Moving image
- MOV
- DV
- WMV
- AVI

Glitch art is a practice that studies ... researches the ... manners to ... audio)visual works. ... formal ... not exists ... have a more ... mutating; it exists ... on the one hand the ... the apparatus (the ... rk, the writer/artist, ... and/or user (the

not (just) the digital file formats, nor the constantly changing ... ions between text, ... dynamics and the ... rent actors make

... aterials thus also ... the medium in a ... and its inherent

... se the term "glitch ... hat the question of ... genre should be ... Studies. Also, in the ... a need to research ... e point where the ... t unknown, new ... new commodities

... ust a vernacular of ... y but also includes ... e history of the

... ng consistent within ... it is the critical use ... al or designed. And ... or any other glitch

Please respect the software. Now it is too little too late - you did not try hard enough - you are just not good enough. [system shut down indefinitely] Dear mr compression I write a 1000 poems to you is this what they call progress? Warmly yours, the noxious angel of history

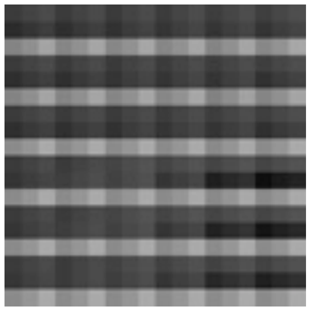
work, this is what I find most interesting to look for: what critical elements play a role in the work - does the work criticize something, or does it show the technology in a critical state?

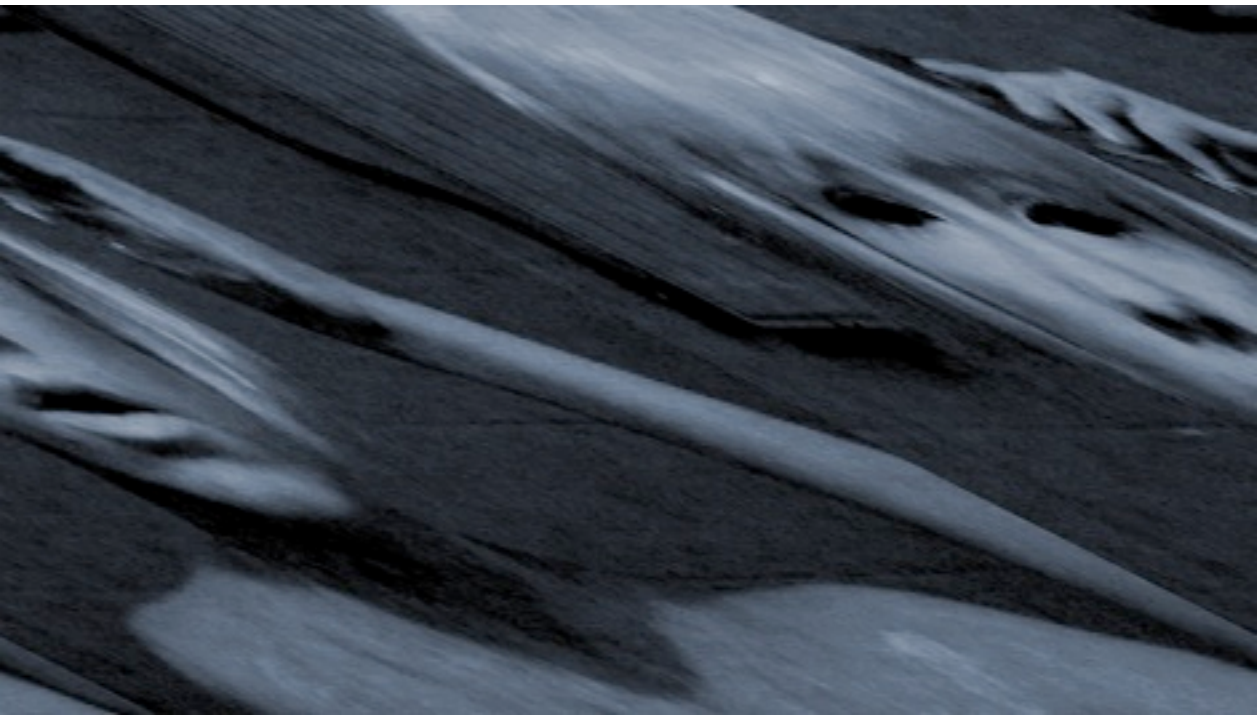
UNCOMPRESSED RAW

<-- Photoshop .RAW 3channel, 2020x1138. (with scan lines from the original video)

Raw image files contain minimally processed data (pixels) from the image sensor of either a digital camera, image scanner, or motion picture film scanner. The file header of a RAW image typically contains information concerning the byte-ordering of the file, the camera sensor information and other image metadata like exposure setting, camera/scanner/lens model, date (and, optionally, place) of shoot/scan, format, size, number of colors, and other information needed to display the image. It is possible to save a RAW image file without a header (choose header=0). When the interleaved RAW image is saved without a header the computer doesn't know the dimensions or any other crucial information that is needed to reconstruct the image out of the image data.

<-- Photoshop .RAW, (h=0) I opened a 3 channel interleaved RAW document as a 1 channel (1 color) interleaved document. (reversible databend)

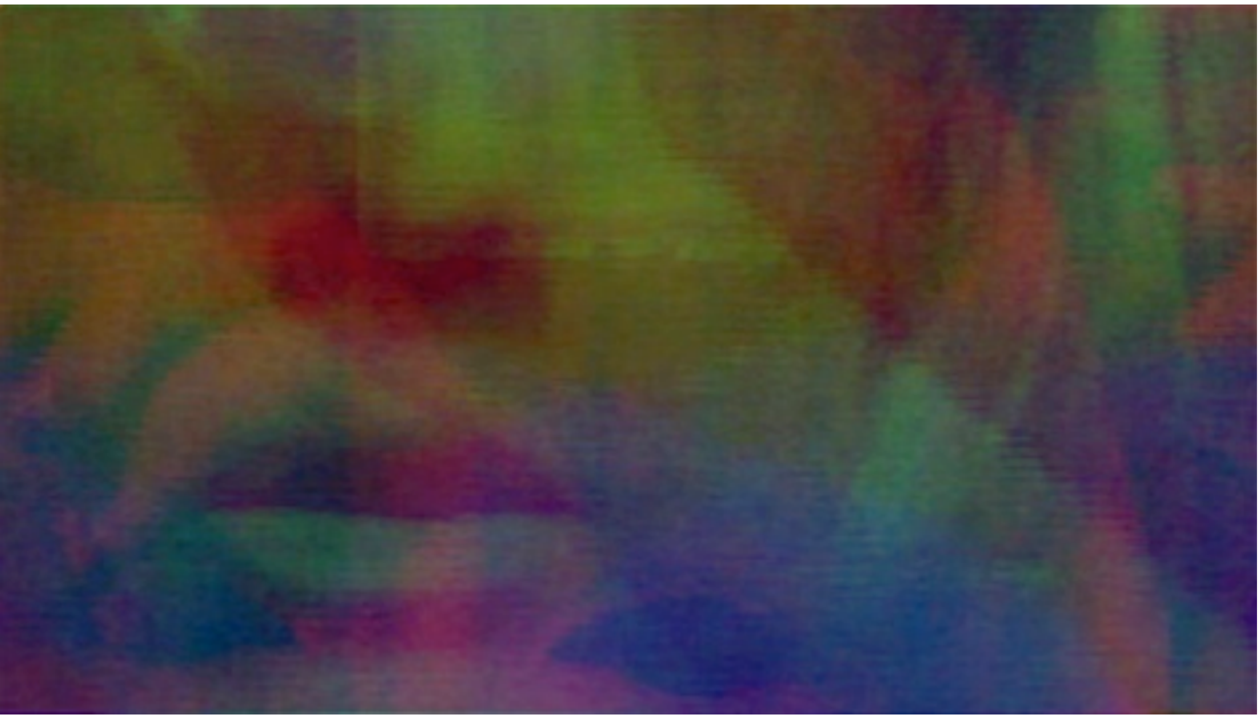




When you open the image, softwares like for instance photoshop will ask you for this data, giving you the opportunity to "bend" (reversible) the image.

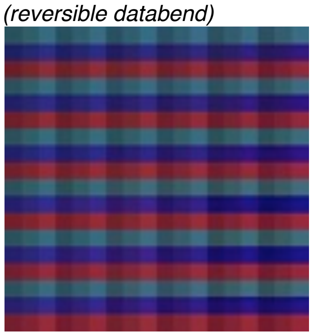
At this moment, you will be able to choose the dimensions, the amount of channels and if the image will be displayed interleaved or non-interleaved. By entering another value then the original, the image will be displayed in a distorted way.

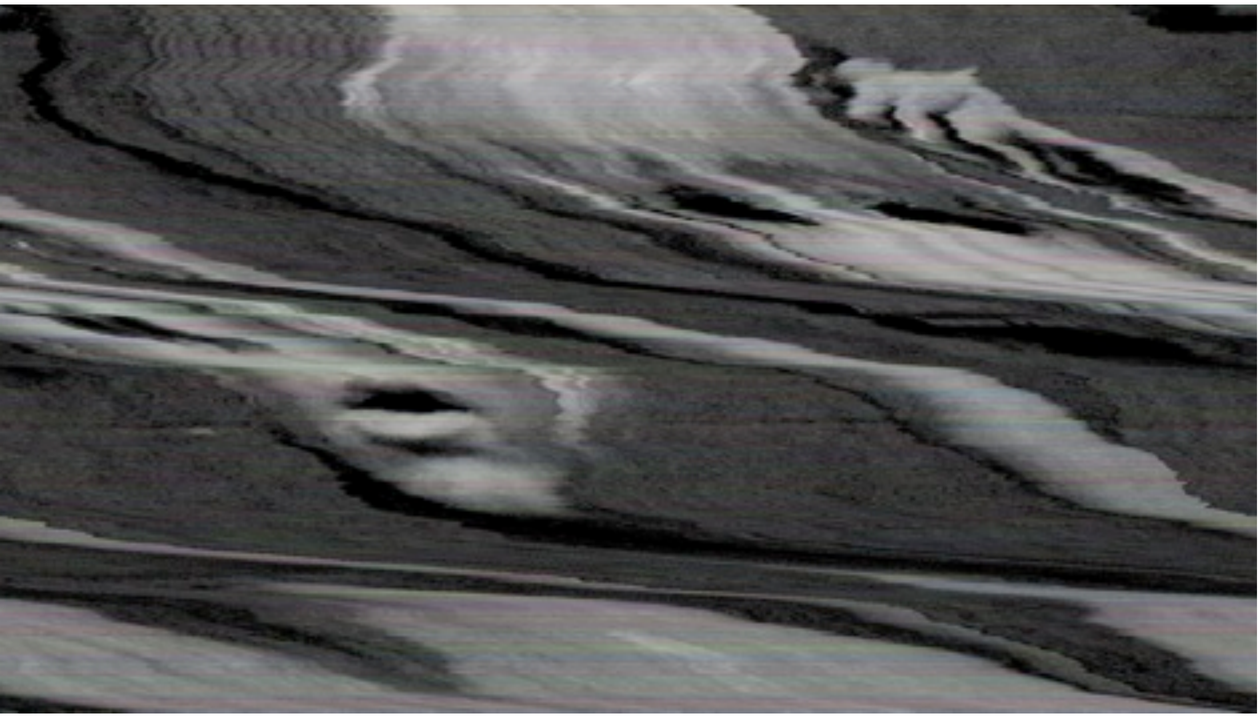
<-- Photoshop .RAW, (h=0) interleaved, I opened a 3 channel interleaved document but entered a slightly smaller value for the width. (reversible databend)



In the case of a RAW image file, interleaving and non-interleaving refer to the order in which the RGB color values of every pixel are stored.
In an interleaved Raw image, the data is stored in a RGBRGBRGB sequence.

<-- Photoshop .RAW, (h=0)
I opened the 3 channel interleaved RAW document as a 3 channel non-interleaved 8 bit document. (reversible databend)

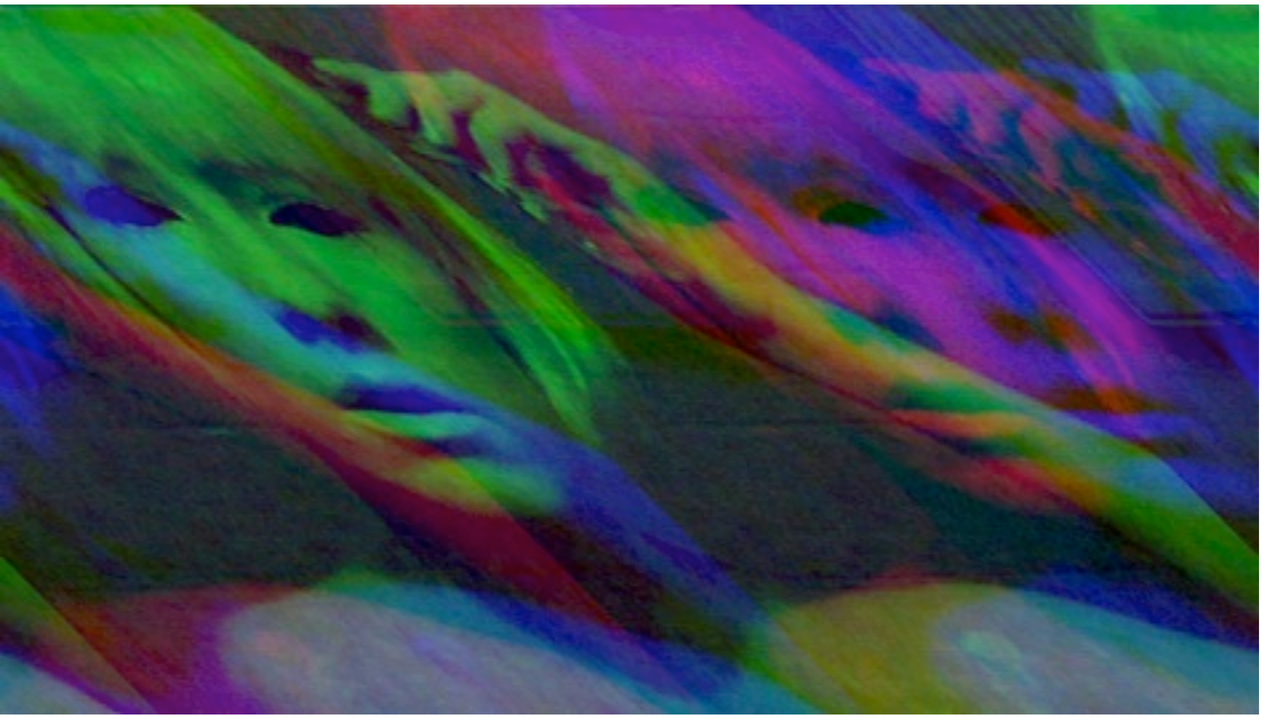




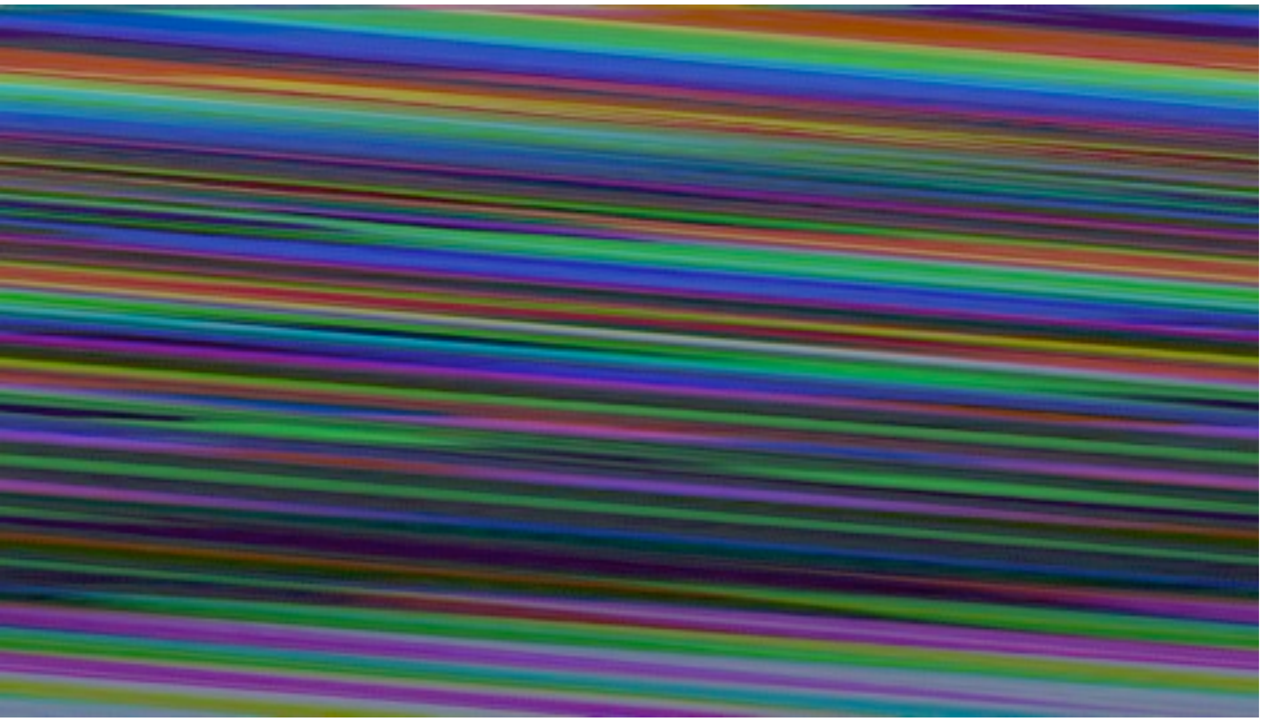
<-- Photoshop .RAW, (h=0)
I opened a 3 channel interleaved RAW image in Microsoft Word (Convert to Text Only) and saved it. This technique reformats the image data into a Microsoft Word document, changing some of the values and adding / deleting some extra data. In the image this shows by abrupt discolorations, and image general image shifts. (Stallio's Wordpad effect) (irreversible databend)

<-- Photoshop .RAW, (h=0)
I opened a 3 channel non-interleaved raw document with smaller hight (databend, reversible)

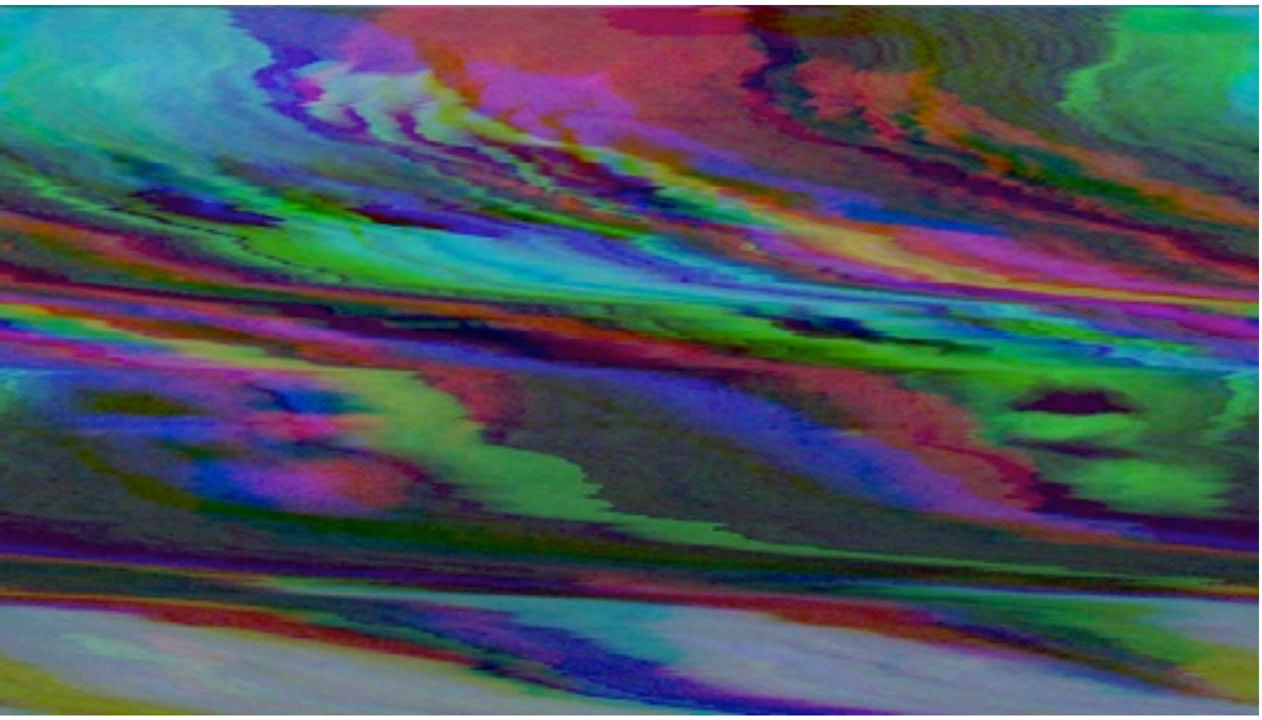
When the image is saved in non-interleaved array, the RGB values are not ordered sequentially but have their own 'layers' with



<-- Photoshop .RAW, (h=0) I opened a 3 channel non-interleaved document but entered a slightly smaller value for the width (1 pixel) (reversible databend)



<-- Photoshop .RAW, (h=0) I opened a 3 channel non-interleaved RAW document with a much smaller width (reversible databend)



<-- Photoshop .RAW, (h=0) I opened a 3 channel non-interleaved RAW image in Microsoft Word (Convert to Text Only) and saved it (wordpad, textpad and other text editors could also work). This technique reformats the image data into a Microsoft Word document, changing some of the values and adding / deleting some extra data. In the image this shows by abrupt discolorations, and image general image shifts. (Stallio's Wordpad effect) (irreversible databend)

BMP

<-- Bitmap (.bmp) bend by copy pasting a bit of the image data over and over. (irreversible databend)

The BMP file format is uncompressed; every bit that indexes a bitmap pixel value is packed within a linear row, "upside-down" with respect to normal image raster scan order, starting in the lower right corner, advancing row by row from the bottom to the top.

<-- Low quality Bitmap (.bmp) Random data replacement changed the values of the indexed colors in the color palette (irreversible databend)

This is why, when you copy-paste some of the image data the lower part of the image will still be intact, while the upper part of the image only shifts (and sometimes discolors)

In BMP files, and many other bitmap file formats the color palette consists of a block of bytes (a table or palette) listing the colors available for use in a particular indexed-color image.
Each pixel in the image is described by a number of bits (1-32 bit color depth) that index a single color from the color palette, that is described right after the header.

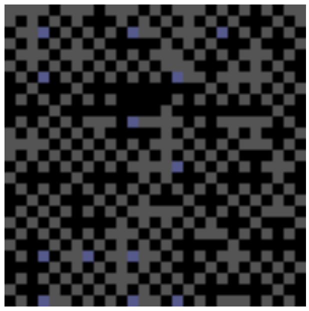
The BMP color palette uses the interleaved RGB color model. In this model, a color depends on different intensities (from 0 to 255) of the primary RGB colors. A color is thus defined by the final intensities of R+G+B. When you copy-paste the image data, the intensity data From B can (for instance) shift to the R, creating sudden discoloration.

<-- .BMP [Wordpad effect] In comparison to the RAW image, the image now waves towards the right. This is because the BMP raster format is saved from left to right, top to bottom. (irreversible databend)

LOSSLESS GIF

<-- Graphics Interchange Format (.gif), 8 colors restricted pattern (with dither)

Graphics Interchange Format is a bitmap image format that supports 8 bits per pixel and can therefore consist of no more then 256 colors.
The format supports animation.
Dither (the grainy blocky artifacts) is an intentionally applied form of noise used to



“randomize quantization error”; the difference between the actual analog value and quantized digital value. This error is caused by truncation (the discarding of less significant information). Dither thus helps to prevent from large-scale patterns such as "banding" (stepwise rendering of smooth gradations in brightness or hue). Moreover, the not available colors are approximated because the human eye perceives the diffusion as a mixture of the colors. This creates the illusion of color depth.

<-- Graphics Interchange Format (.gif), non-interlaced, 8 colors, restricted pattern (with dither) (irreversible databend)

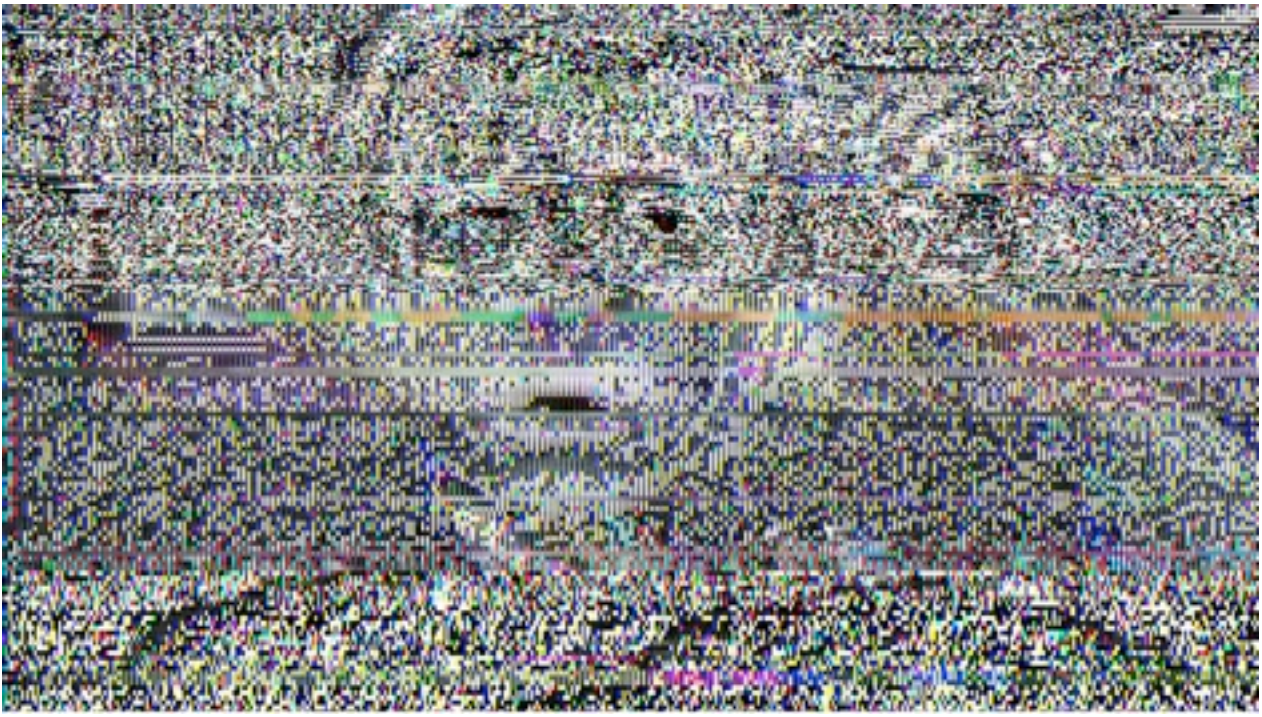
<-- Graphics Interchange Format (.gif), interlaced, animated 8 colors, restricted pattern (with dither) (irreversible databend)

The gif format uses a 4 pass one dimensional interlacing strategy. This means that one half of the image, consisting of every other row of pixels is rendered after the other half. In the image on the left this shows through a gradual displacement during weaving (the putting together of the two layers), which resulted in a second “ghost image” (or combing artifacts with jagged edges).

PNG

<-- Portable Network Graphics (PNG). interlaced (irreversible databend) pre-compression: filtering (prediction) 1-2 of the 7 stages before the image is reconstructed. (irreversible databend)

Like GIF PNG is a lossless compressed raster format, which means that it represents an image as a two-dimensional array of colored pixels.



PNG is a bitmapped image format that employs lossless data compression and offers a 7-pass 2-dimensional interlacing scheme—the Adam7 algorithm.

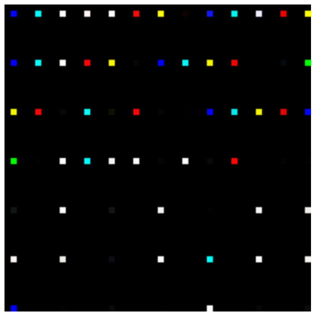
This is more sophisticated than GIF's 1-dimensional, 4-pass scheme, and often allows for a clearer low-resolution image to be visible earlier in the transfer. This is visible in image 1 which just passed its first stage of the 7 part interlacing scheme. In this stage a part of the image is rendered almost flawless, while the further it gets rendered, the more the corrupted data becomes visible.

<-- PNG. interlaced. stage 3/4. (irreversible databend)

<-- PNG. interlaced. stage 5/6.(irreversible databend)

<-- *PNG. interlaced (irreversible databend)*

The 7-pass de-interlacing of the corrupted image sometimes stops early (as early as the first pass).



<-- *PNG. interlaced (irreversible databend)*

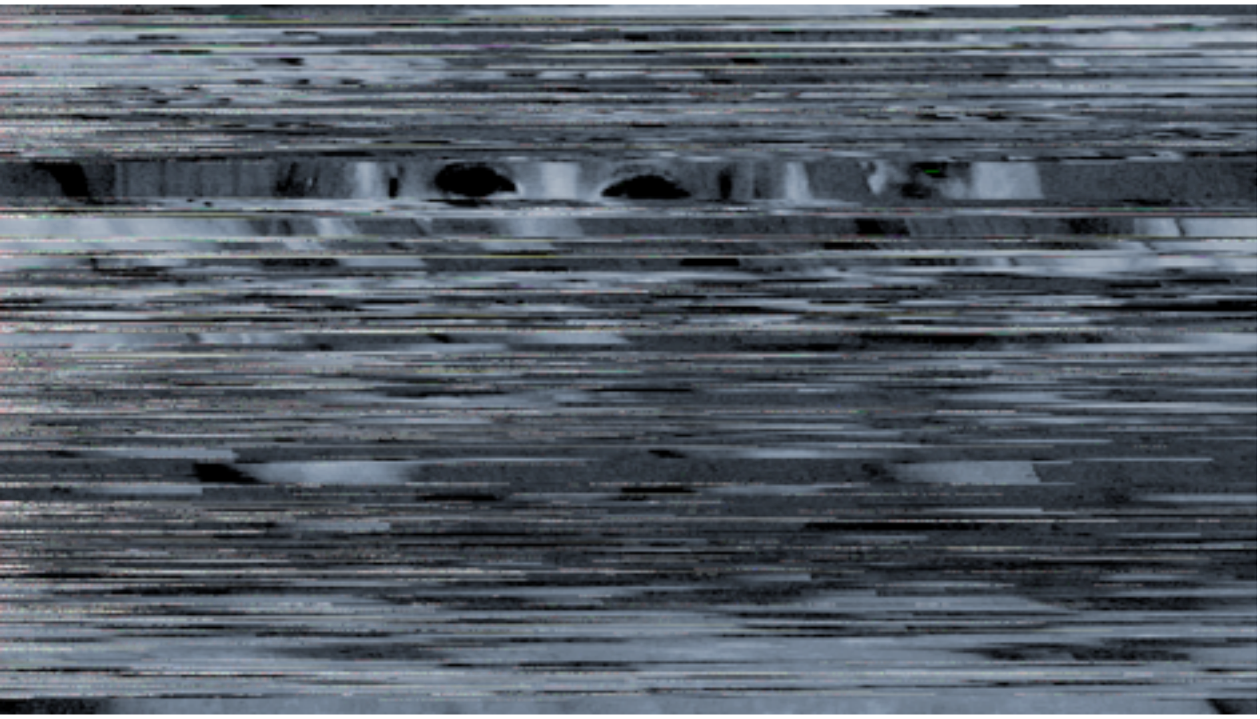
<-- *PNG. non-interlaced (irreversible databend)*

TARGA

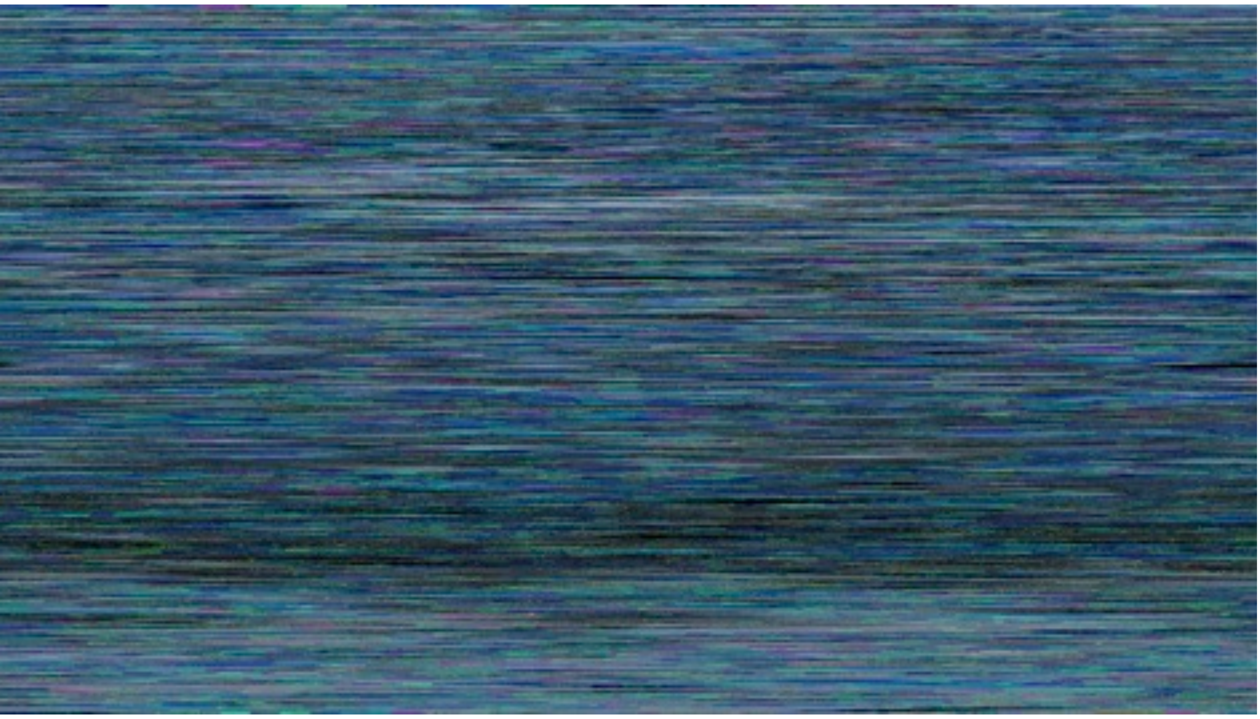
<-- Truevision Advanced Raster Graphics Adapter; (.TGA) is an initialism for Truevision Graphics Adapter. (can be lossy or lossless) 16 bits Lossless RLE compressed (irreversible databend)

Targa recognizes over half a dozen image file formats, some of which are more widely used than others.
I have databend a Tagra compressed file by searching and replacing + adding some data. Apparent are shifts of some blocks and some small color shifts.
To understand how these shifts went the way they went, is however to complex and I am not going to try to understand or explain this in this pdf.

<-- Targa (.tga) 24 bits Lossless RLE compressed (irreversible databend)



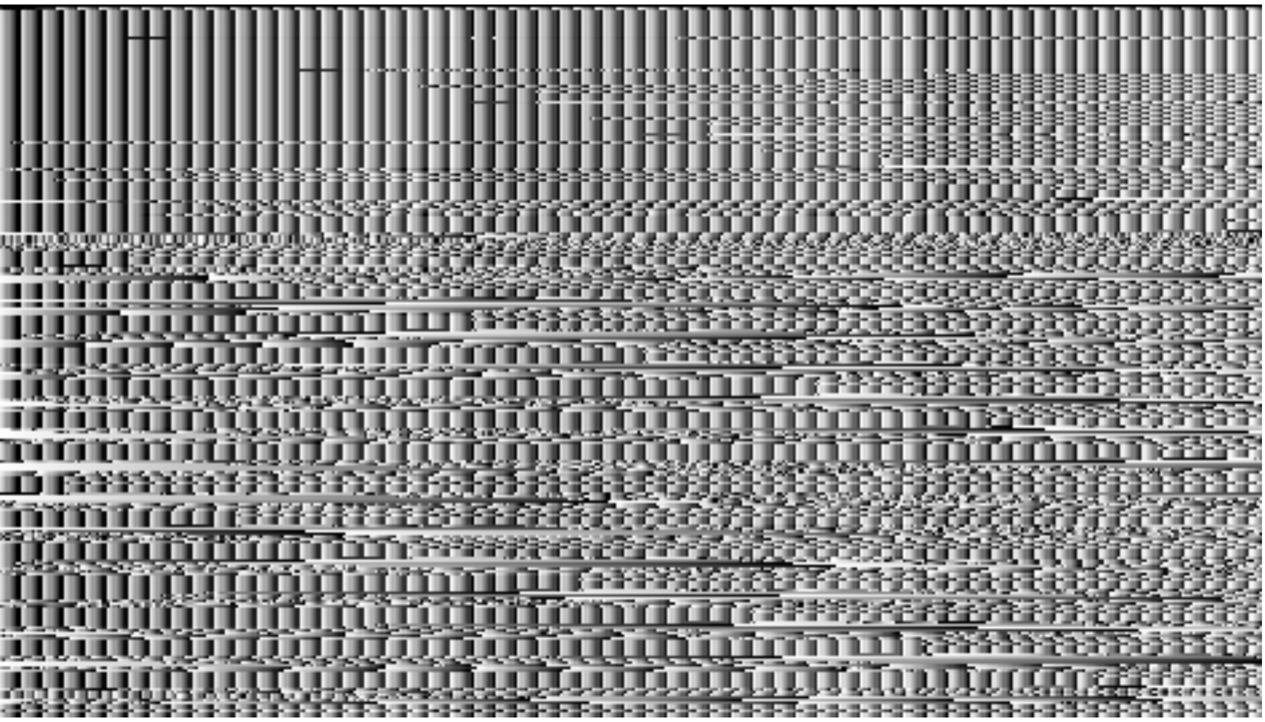
<-- Targa (.tga) 32 bits Lossless RLE compressed (irreversible databend)



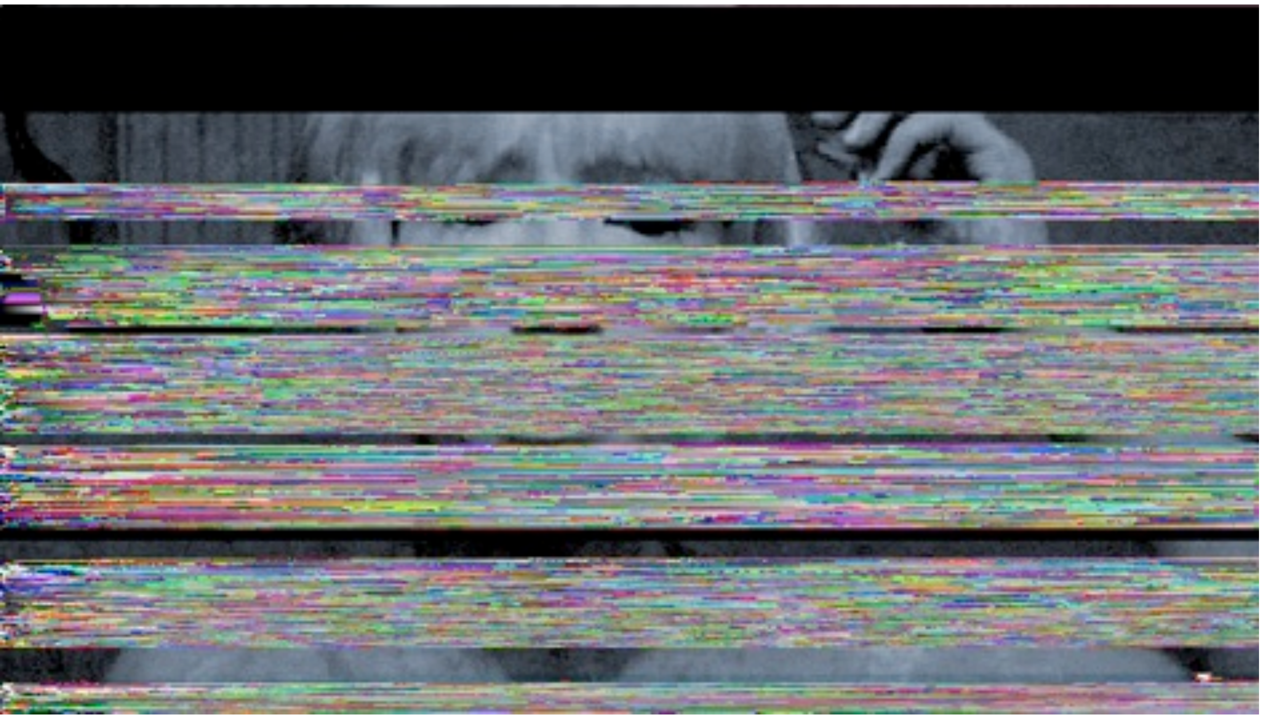
<-- Targa (.tga)
32 bits Lossless RLE compressed
(irreversible databend)

<-- Tagged Image File Format (.TIF) or in short TIFF, with white background layer
(irreversible databend)

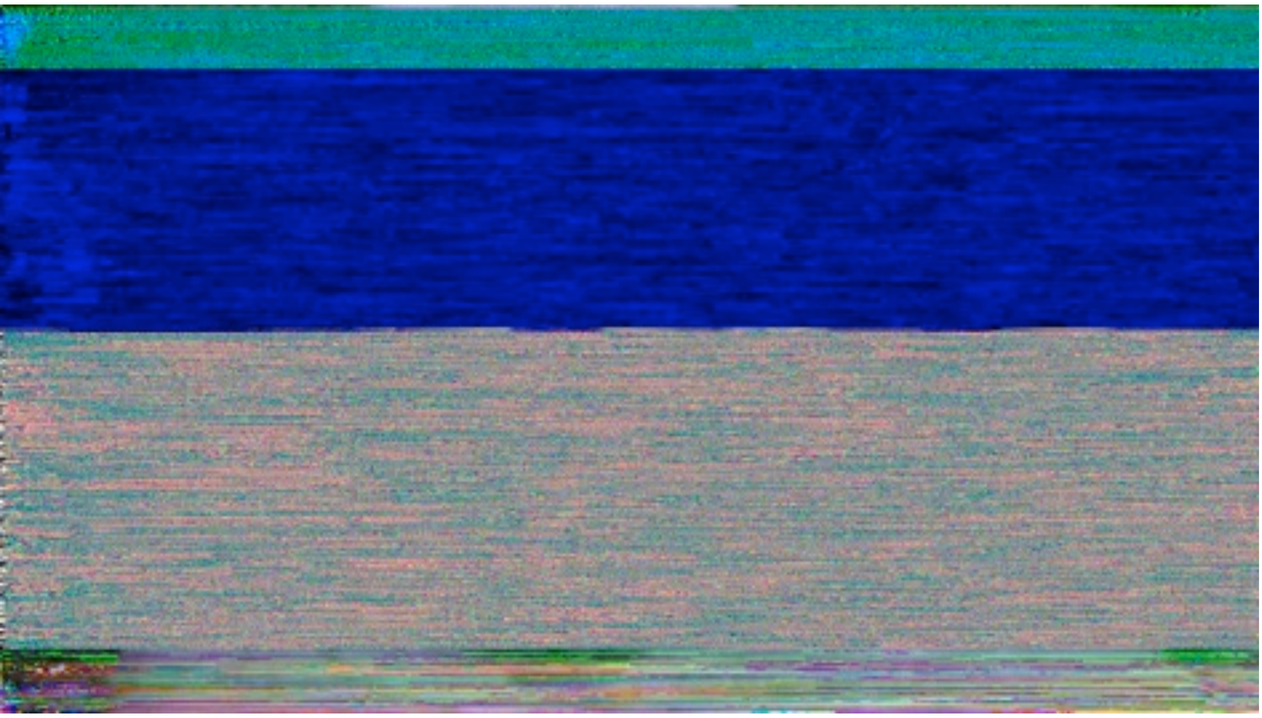
Just like Targa, TIFF is a very complex compression. I have had some really interesting diverse experiences with this file format, but I find it very had to get grips on the reason why they come to the surface the way they do. Thats why these are real glitch bends to me.



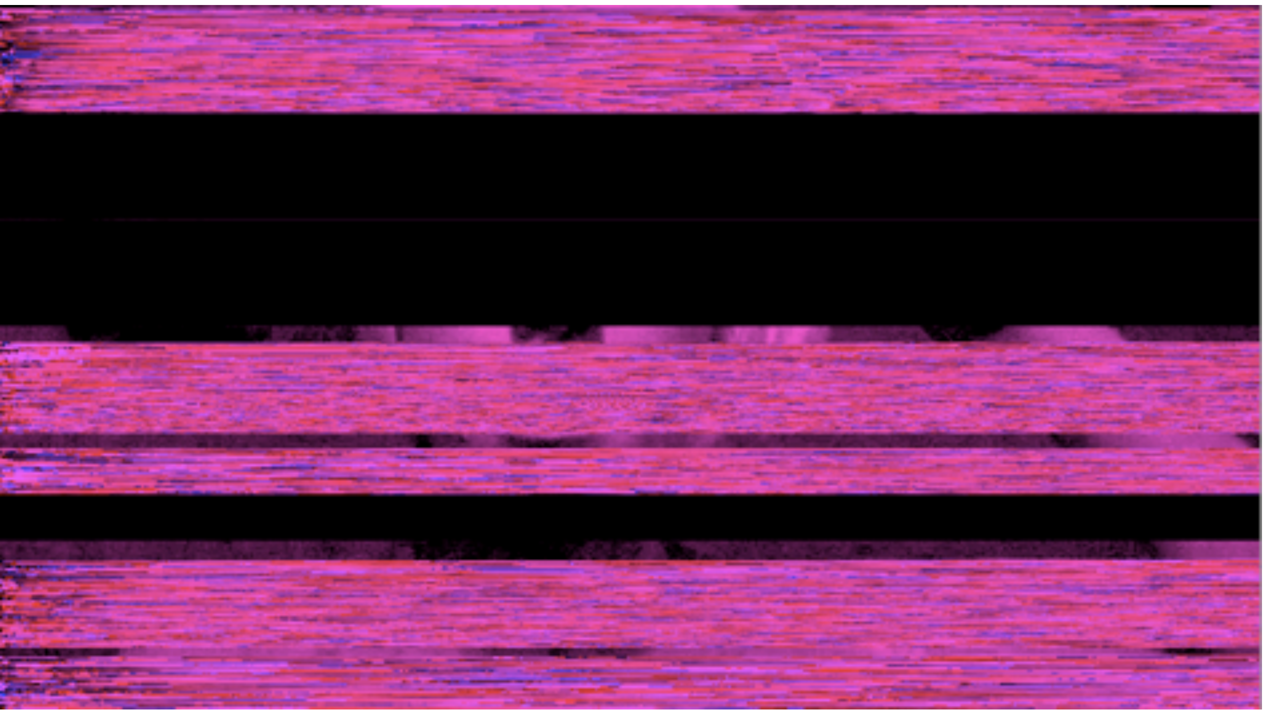
<-- TIFF
With white background layer
(irreversible databend)



<-- TIFF
interleaved macintosh Save
Image Pyramid
(irreversible databend)



<-- TIFF
interleaved macintosh Save
Image Pyramid
(irreversible databend)



<-- TIFF
interleaved macintosh Save
Image Pyramid
(irreversible databend)

PSD

<-- *PSD document (irreversible databend)*

Color channel shift displacement

<-- *PSD document (irreversible databend)*

<-- *PSD document (irreversible databend)*

LOSSY JPEG

<-- Joint Photographic Experts Group (.JPG) (lossy)
severely downsampled so that the 8x8 macroblocks (and quantization error) are apparent.
(irreversible databend)

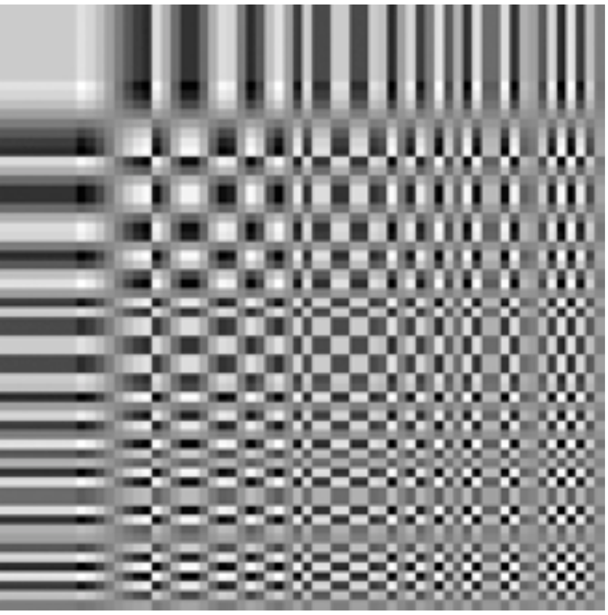
A JPG compression consists of 6 subsequent steps:

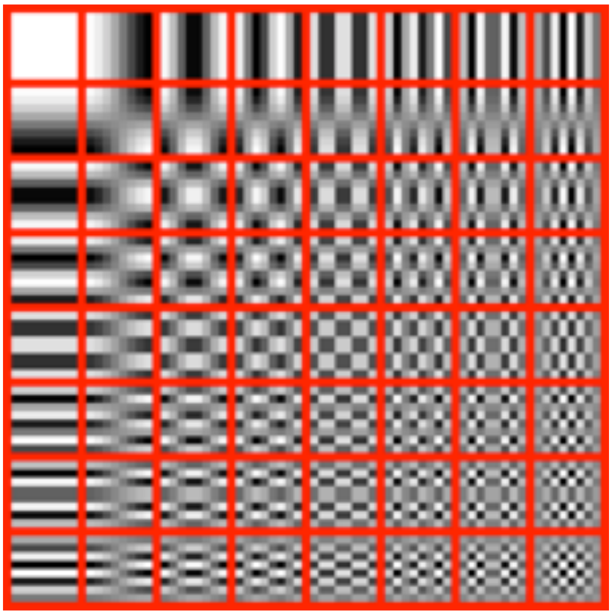
1. Color space transformation
2. Downsampling
3. Block splitting
4. Discrete cosine transform
5. Quantization
6. Entropy coding

1. Initially, images have to be transformed from the RGB color space to another color space (called Y′CbCr), that consists of three components that are handled separately; the Y (luma or brightness) and the Cb and Cr values (chroma or color values, which are divided into hue and saturation).

2. Because the human eye doesn't perceives small differences within the Cb and Cr space very well, these elements are downsampled.

3. After the color space transformation, the image is split into tiles or macroblocks. Rectangular regions of the image that are transformed and encoded separately.





<-- 8 x 8 DCT basis patterns of a JPG.

4. Next, a Discrete Cosine Transform (which works similar to the Fourier Transform function, exploited in datamoshing and macroblock studies) is used to create a frequency spectrum, to transform the 8x8 blocks to a combination of the 64 two-dimensional DCT basis functions or patterns (as differentiated by the red lines).

5. During the Quantization step, the highest brightness-frequency variations become a base line (or 0-value), while small positive and negative frequency differentiations get a value, which take many fewer bits to represent.

<-- high frequency mapping from which basic values are derived.

6. finally, entropy coding is applied. Entropy coding is a special form of lossless data compression that involves arranging the image components in a "zigzag" order. This allows the quantized coefficient table to be rewritten in a zigzag order to a sequence of frequencies. A run-length encoding (RLE) algorithm groups similar frequencies together and after that, via "Huffman coding" organizes what is left.

Because the RGB color values are described in such a complex algorithms, some random data replacement often results into dramatic discoloration and other effects.

<-- low res JPG, Baseline standard.
(irreversible databend)

<-- low res JPG, Progressive
(irreversible databend)

The very high compression ratio of this jpg effects the quality of the image and the size of the artifacts.

When using quantization with block-based coding, as in these JPEG-compressed images, several types of often unwanted artifacts can appear, for instance ringing or ghosting. In the bend image to the left, the low quality and corruption have made these artifacts more apparent.

<-- low res JPG, Progressive (irreversible databend)

Ringing is (often) the result of the loss of high frequency components, but can also be used to “enhance” the image, because this artifact put an emphasis on the edges (in moving image these artifacts can also be referred to as mosquito noise).

Ghosting is an artifact that appear when a part of the image is somehow doubled (often by refraction of a television signal within the atmosphere).

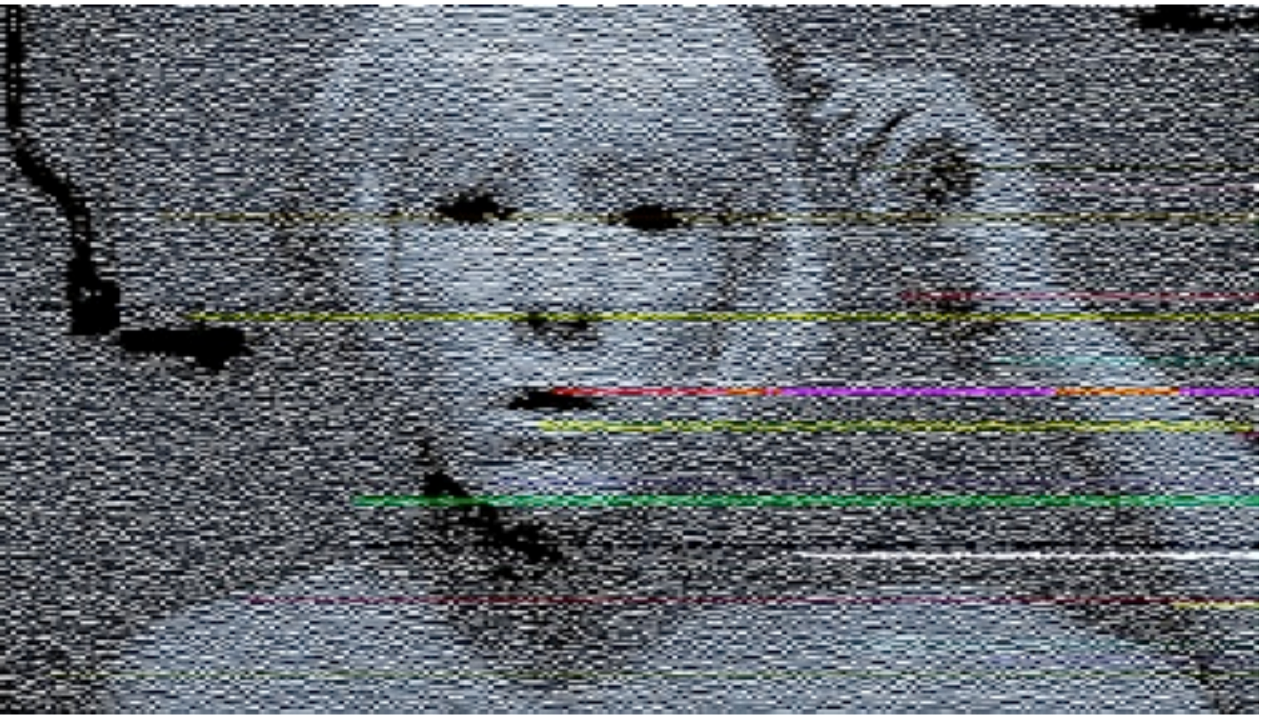
'Blocking' / Staircase artifacts appear most clearly along the curving edges, as a result of the 8×8 jpg blocks. Blockiness in "busy" regions is sometimes also referred to as quilting or checker-boarding.

The JPG to the left shows some typical color distortion (color mismatching) after random data replacement.

It also shows some less typical JPG artifacts called jaggies.
"Jaggies" is the informal name for artifacts in raster images. They are often the result from poor aliasing, which happens when a JPG signal reconstruction after downsampling has produced only high frequency outcomes.

<-- JPG, Progressive (irreversible databend)

<-- JPG, Progessive (irreversible databend)



<-- JPG, Baseline standard (irreversible databend)

<-- JPG, Baseline (irreversible databend)

<-- Joint Photographic Experts Group committee in 2000; low res JPEG 2000. (irreversible databend)

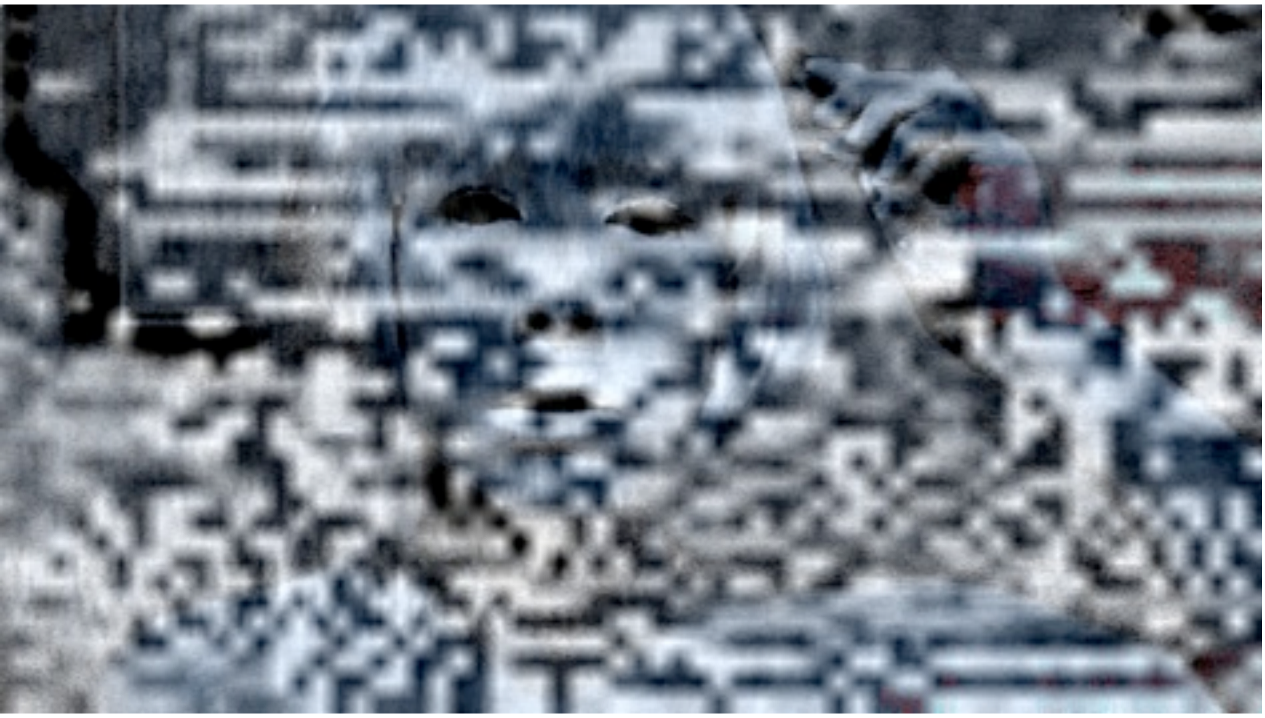
The JPEG 2000 standard was mainly developed because of the many edge and blocking artifacts of the JPG format. JPEG 2000 has improved scalability and edit-ability.
In JPG 2000, after the color transformation step, the image is split into so-called tiles, rectangular regions of the image that are transformed and encoded separately.
Tiles can be any size, and it is also possible to consider the whole image as one single tile. This results into a collection of sub-bands which represent several approximation scales.

<-- JPEG 2000.
(irreversible databend)

A sub-band is a set of coefficients that represent aspects of the image associated with a certain frequency range as well as a spatial area of the image. The quantized sub-bands are split further into precincts, rectangular regions in the wavelet domain.
A wavelet is a wave-like oscillation with an amplitude that starts out at zero, increases, and then decreases back to zero. It can typically be visualized as a "brief oscillation" like one might see recorded by a seismograph or heart monitor.
Precincts are split further into code blocks. Code blocks are located in a single sub-band and have equal sizes.

The chrominance components can be, but do not necessarily have to be, down-scaled in resolution; in fact, since the wavelet transformation already separates images into scales, downsampling is more effectively handled by dropping the finest wavelet scale.

<-- JPEG 2000
(irreversible databend)



<-- Low quality JPEG 2000
(irreversible databend)

MOVING IMAGE MOV

<-- Quicktime movie (.mov) Compression: none 256 colors, least quality

The low quality and the compression to fewer colors introduced posterization artifacts.

Posterization of an image means the conversion of a continuous gradation of tones to several regions of fewer tones, with abrupt changes from one tone to the another.

Posterization may be deliberate or may be an unintended artifact of color quantization.

<-- Quicktime movie (.mov) Compression: Animation 256 greys, least quality codec

displaced scan lines and posterization artifacts.
(I don’t know why the scan lines displaced themselves)

<-- Quicktime movie (.mov) Compression: Animation 1000 colors, least quality codec

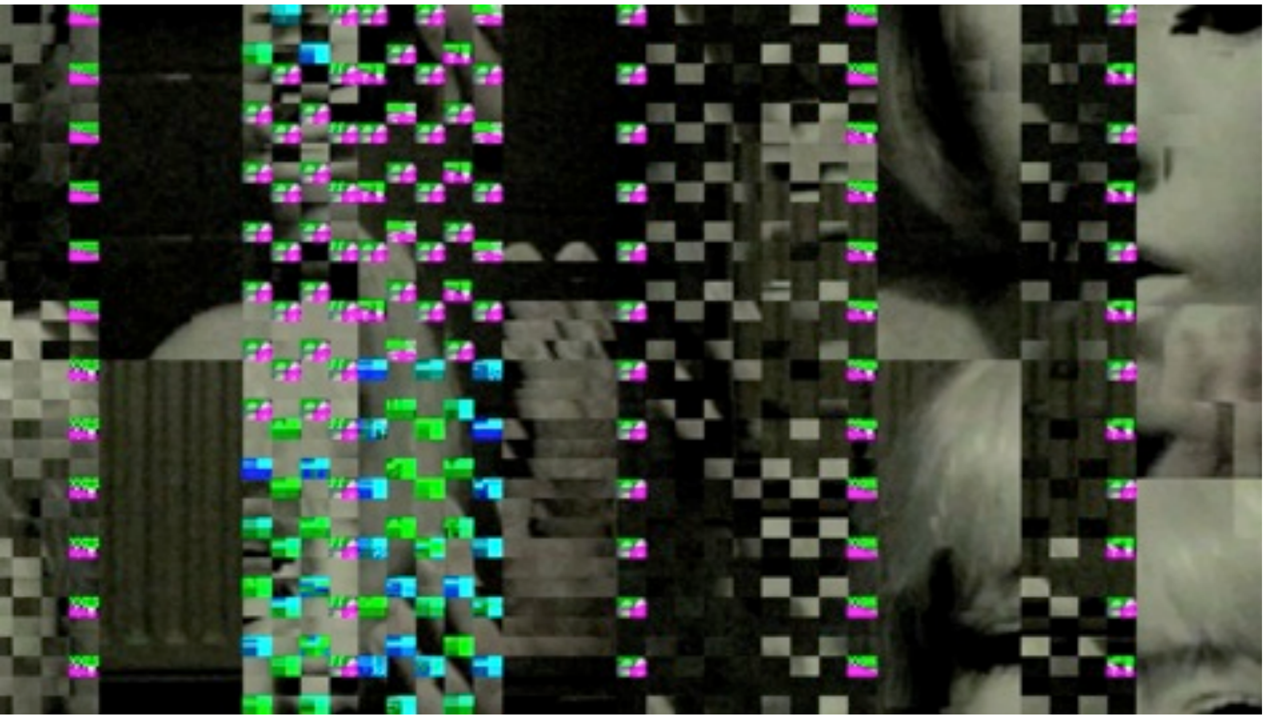
"Banding artifacts," or “false contours” result from color quantization within digitally compressed images.

The lines in the image can also be referred to as 10-bit Banding artifacts.

<-- DV (Digital Video) tape rewind (shows macroblocks)

The artifact to the left shows the macroblocks the footage consists of. This artifact is also referred to as screen tearing (a video artifact that appears when information from two or more different frames is shown a single screen draw).

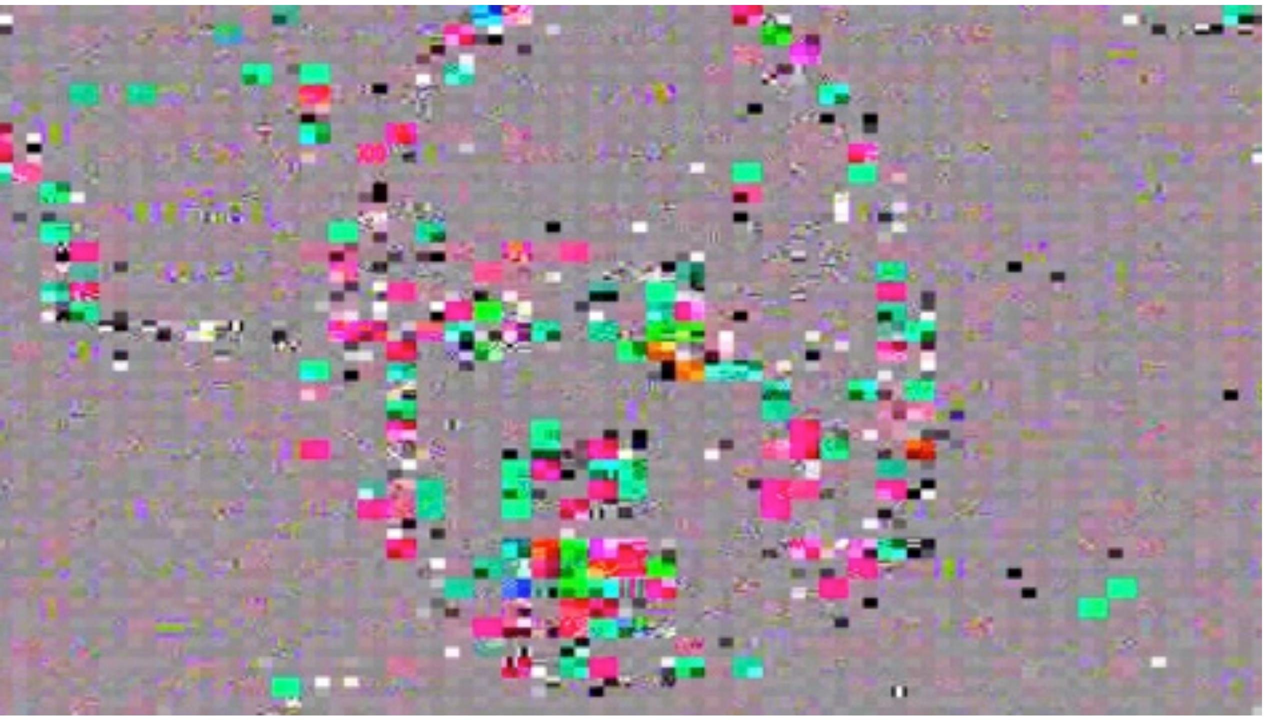
The lossy compressed video image is framed fundamentally different from analog or RAW video footage. First of all, the frames no longer rely upon raw pixels. Instead, macroblocks have become one of the elementary components of the lossy compressed moving image (at least under current standard codecs).



Lossy compressed video often depends on luminance (brightness) and chrominance (coloring) thresholds arranged within 16x16 pixel (more or less) macroblocks within the keyframes (the I-frames) of an image sequence. The thresholds (or frequencies) of chrominance and luminance depend on an oscillating cosine function (following Fourier Transform).

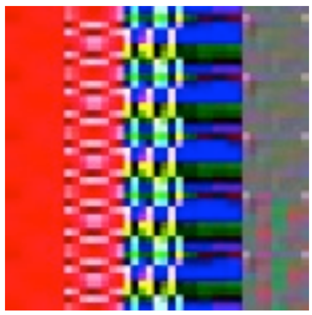
The exploitation/bending of this FFT and its chrominance and luminance values shows in the discolored macroblocks, that now show the otherwise mostly obscured macroblock structures of digital video.

<-- DV, pro50 DV, macroblock bend (rearranged macroblock data) "checkerboarding"

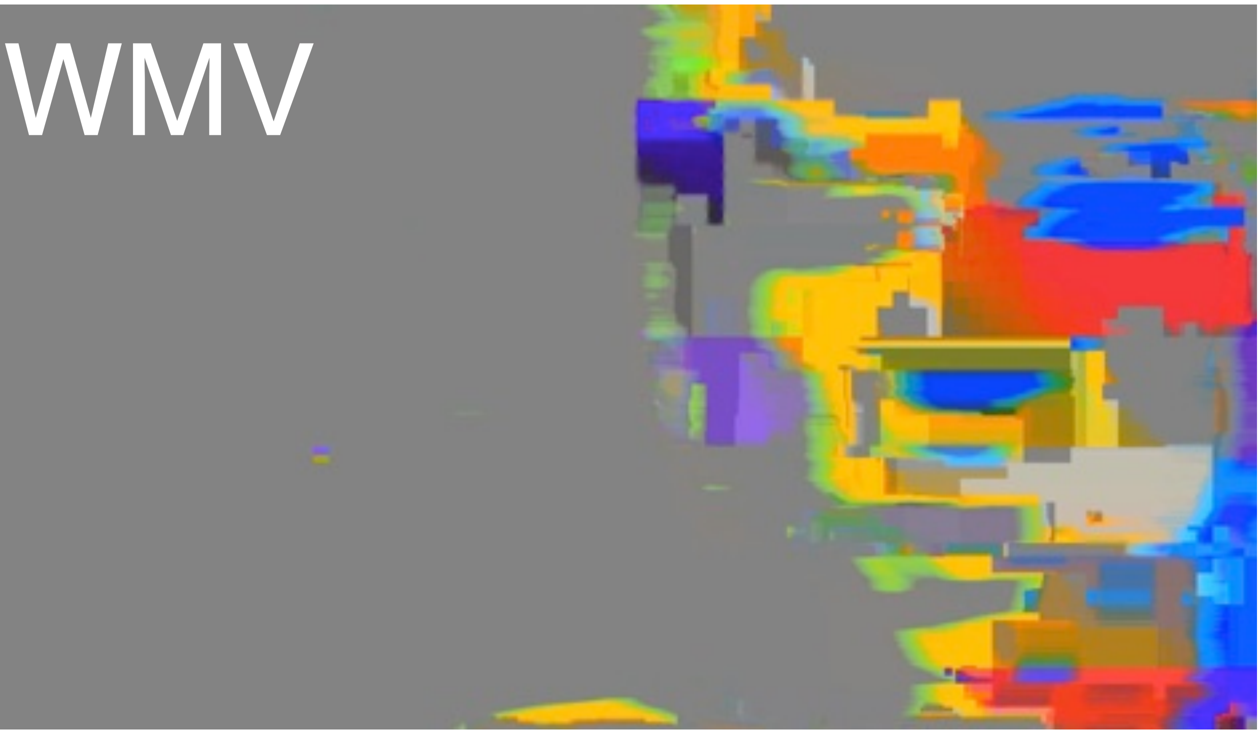


<-- DV pro50 DV, compression: pro50 DV bend luminance (brightness) and chrominance (coloring) values.

Re-organizations of color (chrominance) structures. The matrix of macroblocks is still completely intact but the chrominance values are off. Also referred to as "mosaicking", "pixelating" or "quilting" artifacts.



WMV



<-- Windows Media Video (.wmv) compression: two pass VBR Constrained bitrate: 5kbps (datamoshed)

The handling of space and time within video is significantly different between the linear analog or RAW footage and lossy compressed footage. Besides the macroblock structures within digital film material, the footage is also no longer based on a linear series of discrete images (a sequence); instead the video consists of different kinds of frames (I-frames or reference/key frames, P-frames or forward-predicted frames and B-frames or bi-directional frames), of which only the keyframe possesses a complete matrix of macroblocks.

When a video is encoded, each frame is stored as an I frame (the keyframce) or a P/B frame. An I frame is like a JPEG image, it holds the still image in its entirety. P and B frames are the smart frames that allow videos to be compressed. They store only the differences between the current frame and the last frame.
A motion prediction algorithm is used to calculate the difference between the P- and B-frames. These frames consist of motion vectors that index only the difference in position (the offset) of the macroblocks between the original and the next frame.

<-- Windows Media Video (.wmv) compression: two pass VBR Constrained bitrate: 5kbps (datamoshed)

<-- Windows Media Video (.wmv) compression: two pass VBR Constrained bitrate: 5kbps (datamoshed)

A recently popularized wave of video artworks was based on the deletion of keyframes and the exploitation of the vector motion of P-frames. Which was also referred to as“datamoshing”, “pixel bleeding” or simply “compression art”.

The effect you see in the datamosh videos is what happens when you store only the differences between frames, ie. when all I-Frame or keyframe references have been deleted.

<-- Windows Media Video (.wmv) Compression: two pass VBR Constrained bitrate: 5kbps (datamoshed)

In this type of compression, artifacts tend to remain on several generations of decompressed frames, and move with the optic flow of the image, leading to a peculiar effect, part way between a painting effect r smear that moves with objects in the scene.

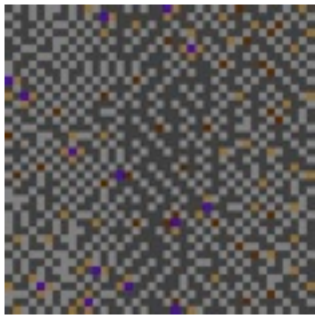
The DIVX file compression follows a similar encoding as the WMV compression. And is more often used.

<-- AVI (.avi) Compression: DIVX (datamoshed)

<-- AVI (.avi) Compression: DIVX (datamoshed)

<-- AVI (.avi) compression: none, 16 colors, least quality (dither)

Color quantization performs a multi-scale analysis on the neighborhood of each pixel, to determine the presence and scale of banding artifacts, and probabilistically dithers the color of the pixel. The overall effect is to "break down" the false contours



<-- AVI (.avi) compression: Cinepak, 256 colors lowest quality

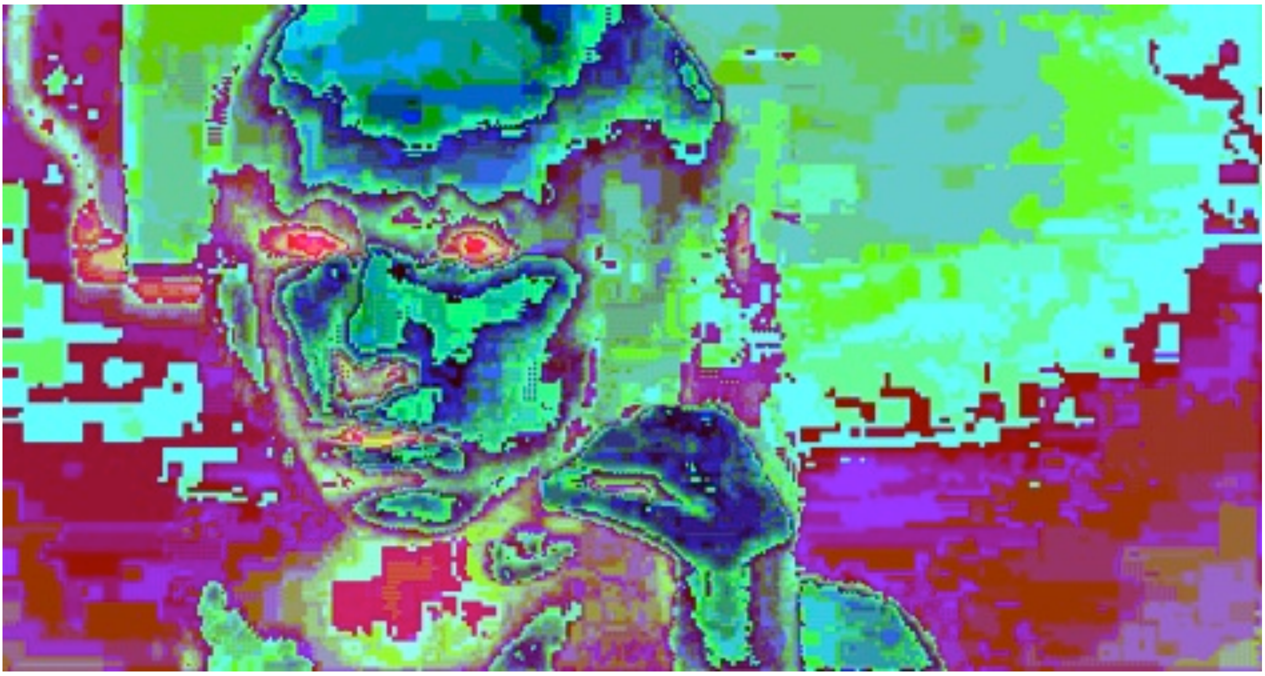
Color quantization

decreasing color depth

<-- AVI (.avi) compression: none, black and white best quality

The image shows a lot of jaggies.

<-- AVI (.avi) compression: none, 16 grays least quality

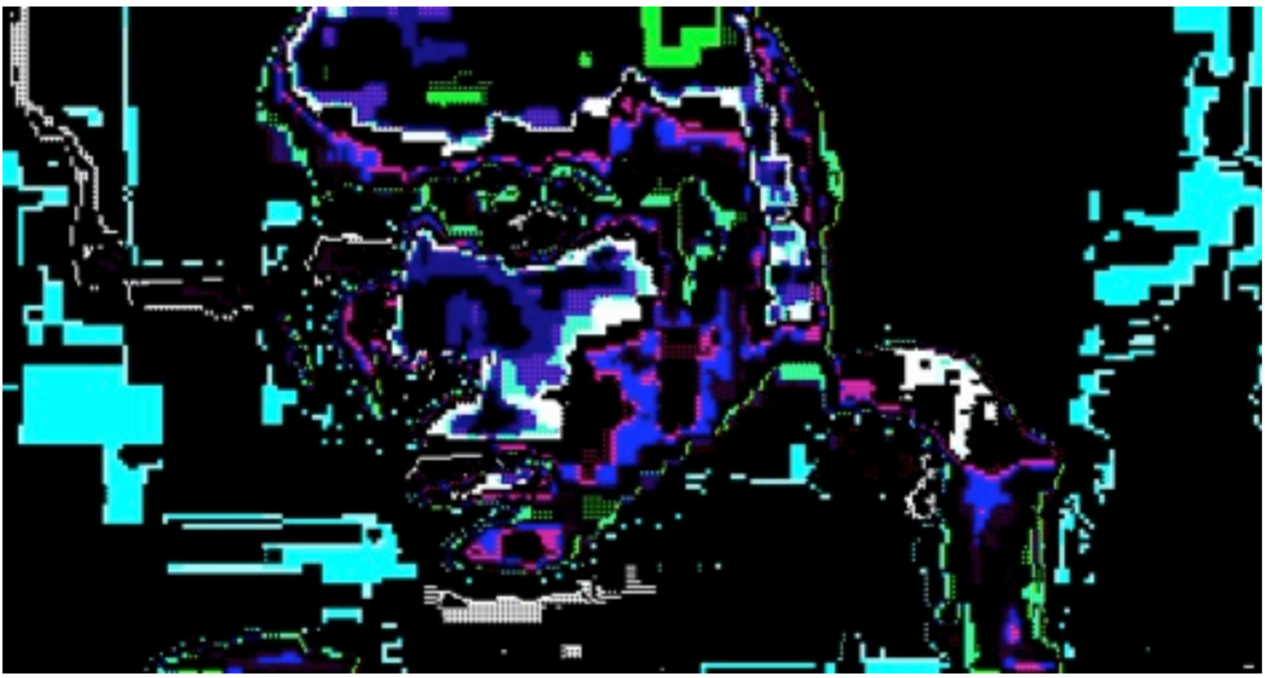


<-- AVI Cinepak compression 256 grays, lowest quality.

Cinepak is my most favorite compression. The compression is based on vector quantization, which results in blocky artifacting at low bitrates.

Cinepak divides a movie into key images and intra-coded images. Each image is divided into a number of horizontal bands which have individual 256-color palettes transferred in the key images. Each band is subdivided into 4x4 pixel blocks. The compressor uses vector quantization to determine the one or two band palette colors which best match each block and encodes runs of blocks as either one color byte or two color bytes plus a 16-bit vector which determines which pixel gets which color.

The cinepak compression is not widely supported anymore, and this is why, when opening the image in a player, the indexed colors of the image are often associated with an incorrect color palette, which results into very unexpected colorations of the image.



<-- AVI Cinepak compression, 256 grays, lowest quality. Opened in VLC player that associated the video with a purple/black/turquoise palette.