

GLITCHART

_técnicas_informações_nova categoria_

LUMANZIN

Pontifícia Universidade Católica de São Paulo

Luísa Manzin da Rocha

GLITCH ART
_técnicas_informações_nova categoria_

São Paulo
2017

LUÍSA MANZIN DA ROCHA

GLITCH ART

_técnicas_informações_nova categoria_

Memorial de Produto
Comunicação e Multimeios
FAFICLA PUC-SP
Orientada por: Prof. Dr. Marcelo Prioste

São Paulo
2017

Dedico à igreja católica que desde o maternal recebe dinheiro do meu pai e da minha mãe.
Dedico ao meu pai e à minha mãe por, além da vida em si, terem pagado essa faculdade cara e sem estrutura,
para um curso que tem capacidade de ser extremamente diferenciado e artístico, mas peca na falta de incentivo de uma instituição endividada, presa num sistema social baseado na troca monetária e não na troca justa,
que seria a humana.

Dedico à academia, já que é só assim que se prova que sabe fazer alguma coisa.
Dedico aos juízes, juízas e promotoras que sairão da PUC e rezo para não serem na justiça como são no bar
afinal, não há lugar melhor para conhecer os ideais de alguém

Dedico à todos os artistas de todas as áreas
vale lembrar que manipular um bisturi é arte também
que se permitem expressar e sentir.

Resumo:

Glitch Art é arte digital, trabalha com erros e corrompimentos de arquivos. Exalta detalhes que ficam escondidos em cada formato. Neste estudo tem tutoriais para produção de glitch, uma discussão sobre os tipos de glitch e a criação de uma nova nomenclatura e um produto: um vídeo com estética glitch.

Resume:

Glitch Art is digital art. It is based in data/file corruption. Shows up details that maintains hided in each format. This study have tutorials of glitch, a discussion about glitch types and the creation of a new nomenclature and a product: a glitch video.

#glitch #art #aleatoriedade #random #técnica #technique #estética #aesthetics

ÍNDICE

1 ___ /

6 ___ O GLITCH

_ 10 _____ PURE-GLITCH/GLITCH-ALIKE

_ 12 _____ GLITCH-SIM

13 ___ TÉCNICAS

_ 17 _____ AUDACITY

_ 25 _____ PROCESSING

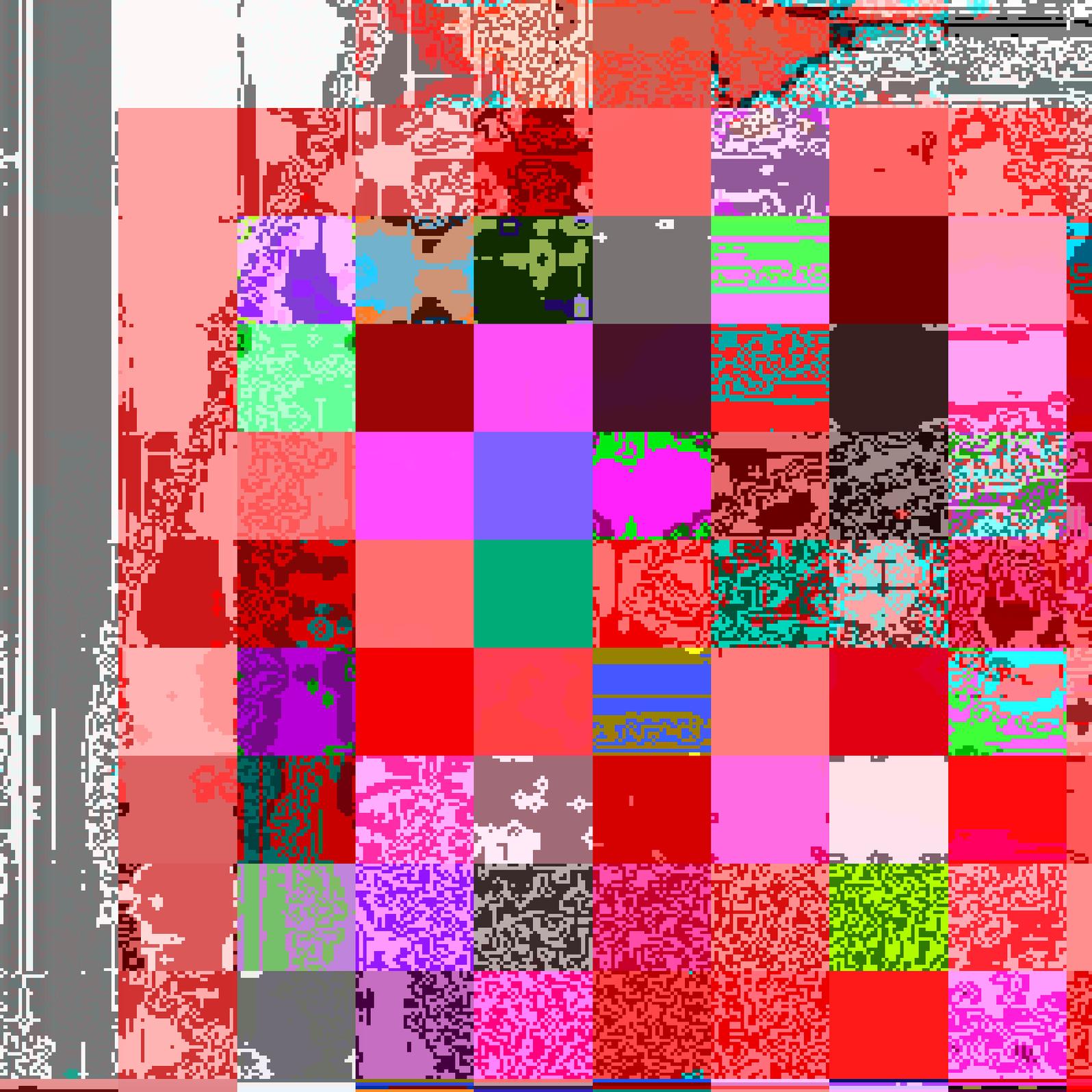
24 ___ SACANAGEM

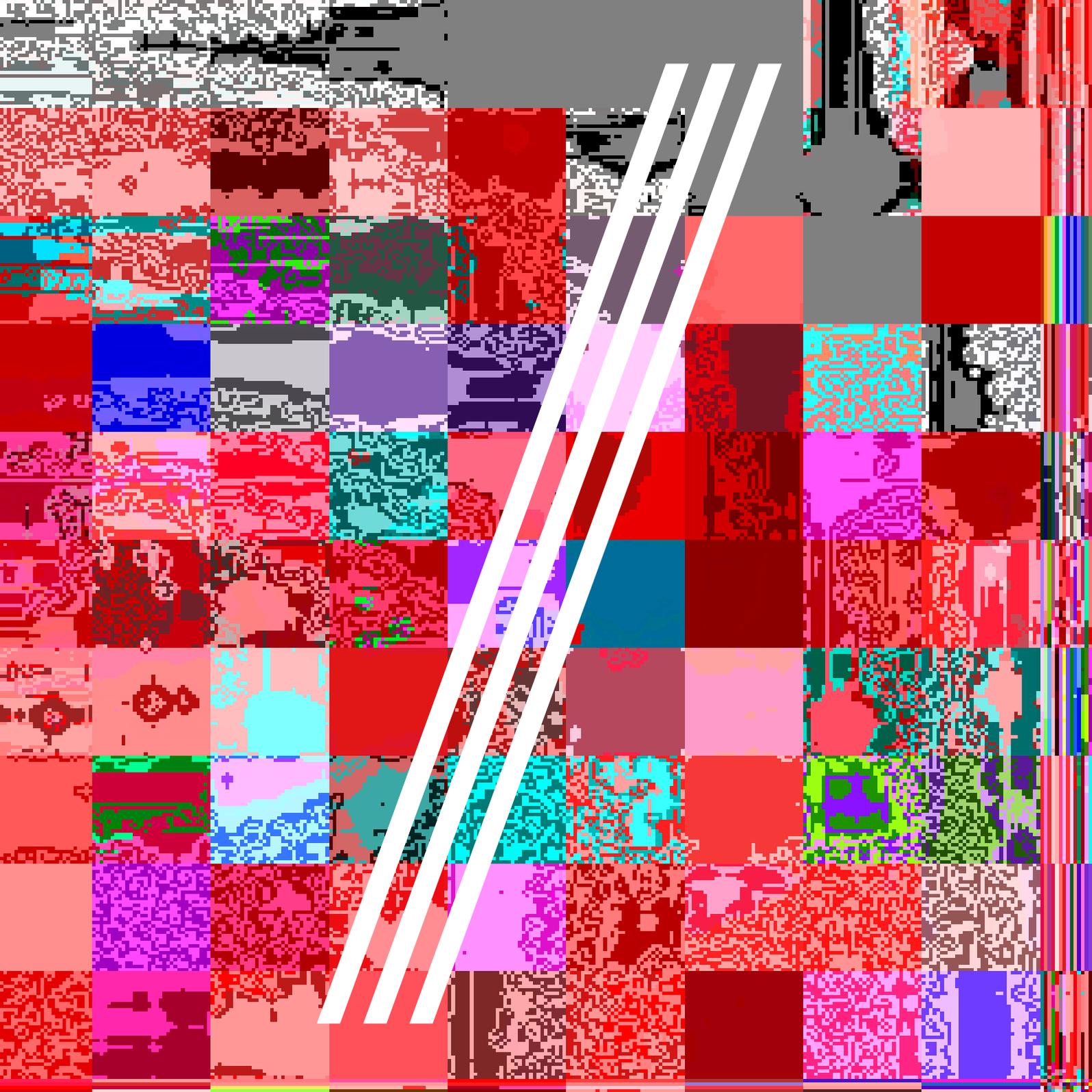
28 ___ ADICIONAIS

30 ___ BIBLIOGRAFIA

INDICE

1	/
6	0 GLITCH
10	PURE-GLITCH/GLITCH-ALIKE
12	GLITCH-SIM
13	TÉCNICAS
17	AUDACITY
25	PROCESSING
24	SACANAGEM
28	ADICIONAIS
30	BIBLIOGRAFIA





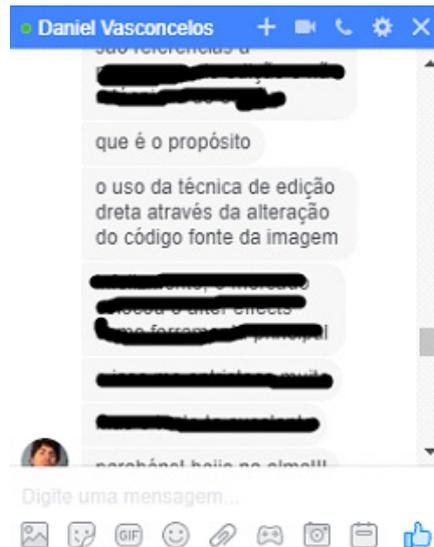


glitch art é um tipo de arte digital que trabalha com o corrompimento de arquivos. Existem diversas técnicas para a produção. Elas variam desde transformações analógicas, colocando imãs em cima de TV's, por exemplo, e em programas muito usuais como Notepad++, hex editor, bloco de notas, Photoshop, Audacity, Processing, Avidemux, e outros.

A glitch art pede um pouco de desapego estético e da precisão de resultados por ser “aleatório” _entre aspas porque é aleatório para nós humanos, para a máquina os processos são absurdamente específicos e qualquer detalhe é processado_. Claro, se você estudar a fundo é possível tentar prever o que vem pela frente. Por exemplo: onde altera as cores, onde duplica, o que pixela, os padrões no claro, escuro, preto, branco, vermelho, azul, verde... Aí é possível querer e conseguir alcançar um padrão estético específico, mas nunca vai ser exatamente o que se imagina. Esse é o caso da técnica GLIC, que é o código de um programa de edição de imagem com fins glitch escrito por Tomasz Sulej.

////para fazer exatamente o que quer tem que simular////

Glitch art é a manipulação mais densa dos arquivos, mexe com as menores partes, o código fonte; este é seu propósito.



Na arte, o glitch foi catalogado pela primeira vez pela Anna Scot no seu trabalho *Anti-fractal*

(2001). Mas o termo ganhou visibilidade de verdade com o trabalho *Aesthetics of failure* (2002), de Kim Cascone, de acordo com Gazana (2013):

“A primeira vez que a palavra glitch apareceu para referir-se às obras visuais é creditada a Ant Scott, com a criação da expressão Glitch art, em 2001, em um artigo intitulado “Anti-fractal” (FERNANDES, 2010, p.22). Apesar disso, o termo ficou mais conhecido, primeiramente, no mundo da música eletrônica, por causa do ensaio de Kim Cascone, chamado “The aesthetics of failure”, de 2002, onde o autor faz a primeira formalização da estética do erro. “Cascone discute o erro na música eletrônica, mas sua influência estendeu-se para a Arte Digital em geral.” (FERNANDES, 2010, p. 21)”.
GAZANA, C. et al. Glitch: estética contemporânea visual e sonora do erro. In: Cultura Visual, n. 19, julho/2013, Salvador: EDUFBA, p. 81-99.

A glitch art está dentro da categoria New Media.

“New Media is a 21st Century catchall term used to define all that is related to the internet and the interplay between technology, images and sound. In fact, the definition of new media changes daily, and will continue to do so. New media evolves and morphs continuously. What it will be tomorrow is virtually unpredictable for most of us, but we do know that it will continue to evolve in fast and furious ways.”

.....
“Most technologies described as “new media” are digital, often having characteristics of being manipulated, networkable, dense, compressible, and interactive.[1] Some examples may be the Internet, websites, computer multimedia, computer games, CD-ROMS, and DVDs. New media does not include television programs, feature films, magazines, books, or paper-based publications – unless they contain technologies that enable digital interactivity.”

(SOCHA, Bailey and EBER-SCHMID, Barbara. 2014. Online.)

Existem inúmeras técnicas, em inúmeros programas com fins de glitch. Alguns blogs _que estão na Bibliografia deste trabalho_ discutem questões de propriedades de arquivos, processos, tipos, conexões, etc. Além das técnicas.

Esta pesquisa tem como conclusão um vídeo com estética glitch e trilha sonora de ruídos. Nele foi usado técnicas somente a partir dos programas Audacity e Avidemux.

THE
LIFE
OF
SAMUEL JOHNSON
BY
JAMES BOSWELL
IN TWO VOLUMES.
THE SECOND VOLUME.
LONDON: PRINTED BY A. MILLAR, IN THE STRAND, 1791.

THE
MUSIC
OF
THE
MIDDLE
AGES
BY
JOHN
W. CANTON
REVISED
EDITION
OXFORD
UNIVERSITY PRESS
1996

OGITCHI



A palavra glitch significa erro, bug, falha inesperada de aparelhos tecnológicos.

gliCH/
informal
noun

1.
a sudden, usually temporary malfunction or irregularity of equipment.
“a draft version was lost in a computer glitch”

verb
US

1.
suffer a sudden malfunction or irregularity.
“her job involves troubleshooting when systems glitch”

O glitch vai além da estética, é técnica pura. É a investigação, a produção, entender o mesmo arquivo em diferentes programas e linguagens. Conseguir reconhecer a imagem quando está em formato de códigos.

Uma das maneiras de manipular um arquivo de foto ou vídeo é através de programas de edição de texto, mas obviamente uma imagem não abre num programa que funciona apenas com caracteres de letras, números e símbolos. Então, quando abrimos imagens nesses programas, aparece isso:

_____a imagem em .jpg



PURE-GLITCH/GLITCH-ALIKE

Moradi (2004) separou o glitch em duas categorias: a da falha técnica e a proposital, o pure-glitch e o glitch-alike respectivamente.

“O Pure glitch “[...] É o resultado de um mau funcionamento ou erro.” (MORADI, 2004, p. 9, tradução nossa). Aqui, o Glitch puro, o Glitch propriamente dito, é o resultado de um erro não intencional e acidental. O Glitch-alike, segundo Moradi (2004, p. 10, tradução nossa), “são conjuntos de artefatos digitais que se assemelham aos aspectos visuais de erros reais encontrados em seu habitat original”. O Glitch-alike é, portanto, uma representação planejada e intencional manipulada pelo artista.”

GAZANA, C. et al. Glitch: estética contemporânea visual e sonora do erro. In: Cultura Visual, n. 19, julho/2013, Salvador: EDUFBA, p. 81-99.

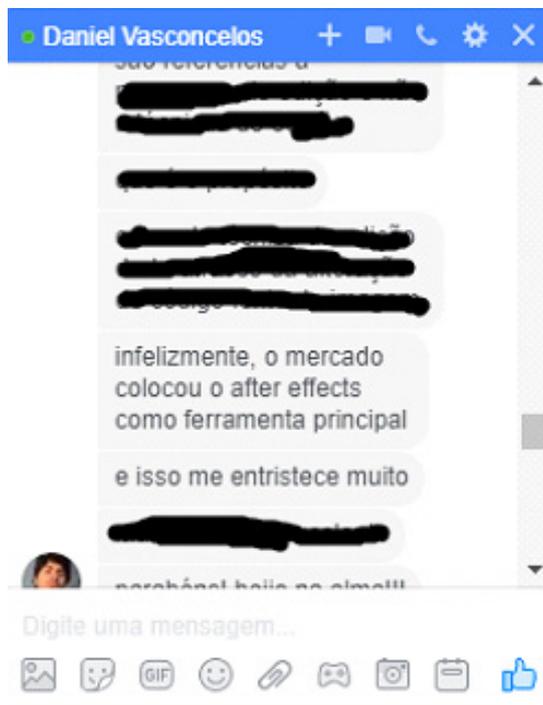
O *pure-glitch* *_glitch-puro_* é quando a imagem se altera graças às falhas elétricas: como quando chove e a imagem da TV pixela. É como se fosse uns respingos de tinta que, agradando ou não o artista, caíram na tela num momento de distração do pintor com o pincel na mão. O *glitch-alike* seria o próprio pintor programar onde jogar aqueles respingos de tinta, deixando com a mesma estética visual de uma distração.

O *glitch-alike* faz sentido, porque se foi provocado de alguma maneira, mesmo que através do corrompimento do arquivo, não é puro. Mas até que ponto vai o *glitch-alike* e onde começa uma nova definição?

Pode ser que seja um pensamento conservador ou separatista, mas usar técnicas de corrompimento, editar as imagens de forma não tradicional usando softwares que não tem como finalidade a manipulação de imagem e/ou vídeo, baixando e testando diversos tipos de técnicas, não merece estar no mesmo patamar dos glitches que são reproduzidos através de predefinições em aplicativos de edição de imagem, por exemplo.

O que mais existe hoje em dia, considerando que essa estética tecnológica e futurista está em alta, a do *glitch* principalmente, são aplicativos e plugins para softwares como AfterEffects que reproduzem *glitch*; simulam o efeito. O que considero problemático destas técnicas é justamente a reprodução da estética. A graça do *glitch* é nunca ver algo igual.

A estética *glitch* é literalmente copiada e planejada, apropriada de maneira falsa _diria artificial, mas qualquer arquivo é artificial por ter sido produzido pelo homem e não pela natureza.



A área da publicidade está usando muito a estética do *glitch*. Quando as pessoas pensam em *glitch*, logo vem a ideia da imagem que trava, se repete e depois volta ao normal, como seriam os *pure-glitch*. Mas o mercado pop, publicitário, edita cortando o vídeo em programas como Premiere, para simular aquele efeito, mas isso não gera nenhum tipo de pixelamento ou falha no arquivo, não trabalha com o propósito da manipulação do arquivo em sí. É uma falha limpa. Foi manipulada usando meios e técnicas para a simulação da estética.

Por isso, esse tipo de uso do glitch teria que estar em outra categoria e não na *glitch-alike*, já que esta inclui os processos de alteração no código fonte. Estaria na categoria da simulação, portanto o



TÉCNICAS

Existem inúmeras maneiras de produzir glitch art a partir do corrompimento. Alguns artistas ainda hoje utilizam cabos e fios, fazem intervenções com imãs, seria um glitch analógico, onde se interfere nas imagens a partir da parte elétrica dos aparelhos onde estão sendo reproduzidos os arquivos. Neste estudo eu não fiz nenhum tipo de teste ou sequer entrei em contato com essa técnica, meu objetivo é manipular os arquivos de maneira digital. Os programas usados no estudo foram: processing 2.2.1, Audacity, Notepad++, avidemux 2.6, photoshop, paint e Hex Editor.

Cada programa tem sua característica e técnica pra mexer e, por exemplo, o paint e o photoshop especificamente foram usados exclusivamente para a conversão de arquivos .jpg para .bmp.

Todos os arquivos, se formos comparar com os corpos orgânicos, têm seus átomos, suas menores partes, coisas que a gente não vê que as constituem. A manipulação acontece nessas micro propriedade.

Dá pra entender melhor o conceito das menores partes do arquivo usando de exemplo o efeito datamosh.



(Print de gif. SPURR-CHEN, Way.
Online.)

A sensação que dá é que a imagem derrete. O que realmente acontece é uma movimentação mapeada de pixels. Se fosse possível colocar um vídeo neste arquivo, veríamos a cantora Miley Cyrus, intacta e aos poucos a tela vai derretendo seguindo a direção dos movimentos dela.

Para esse efeito acontecer, o programa faz uma leitura dos frames do vídeo, mas existe mais de um. O tipo de frame mais usual é o I-frame, que significa *image frame*: várias fotos em sequência. Uma super câmera lenta consegue 1200 frames por segundo__ 1200 fotos por segundo__ uma câmera normal varia de 30 a 60.

Este efeito só é possível porque ao invés de usar o I-frame, usa-se o P-frame do vídeo.

O P-frame é a memória do movimento, mapeamento dos pixels, então se no vídeo tem alguém virando a cabeça num fundo branco uniforme, o fundo permanece o mesmo, só a cabeça da pessoa terá deformação por que só aqueles pixels mudaram de lugar. Existe também o B-frame, que é igual ao P-frame, só que ele mapeia dois movimentos do pixel.

“To understand I-Frame destruction (and the next section on P-frame duplication), we need to briefly cover how video compression works. Video files layer multiple images in sequence in order to make the appearance of motion. However, storing every single image in a video takes up a lot of space! So these images are compressed as “frames”, with three types - I-frames, P-frames, and B-frames.

I-frames are “inter” frames, also called keyframes, which contain the entirety of the frame’s image data. You can think of them as regular images.

P-frames are predictive frames. They don’t actually contain much image data themselves but represent the abstract motion of pixels within chunks of the frame between frames. Essentially you can think of them as tracking where exactly pixels move.

B-frames are bidirectional predictive frames. They’re basically like P frames, except they’re a little more efficient. We don’t use them for glitch as much and you can pretty much ignore them.”

(Spurr-Chen, 2015. Online)

O mais curioso do glitch é isso: revelar lados da imagem que não vemos. Trocar a roupagem de um arquivo diversas vezes. Usar e exaltar, ou não, propriedades escondidas. Um bom exemplo corriqueiro disso é quando uma foto está muito escura e ao aumentar o brilho ou diminuir o contraste, vemos mais detalhes.

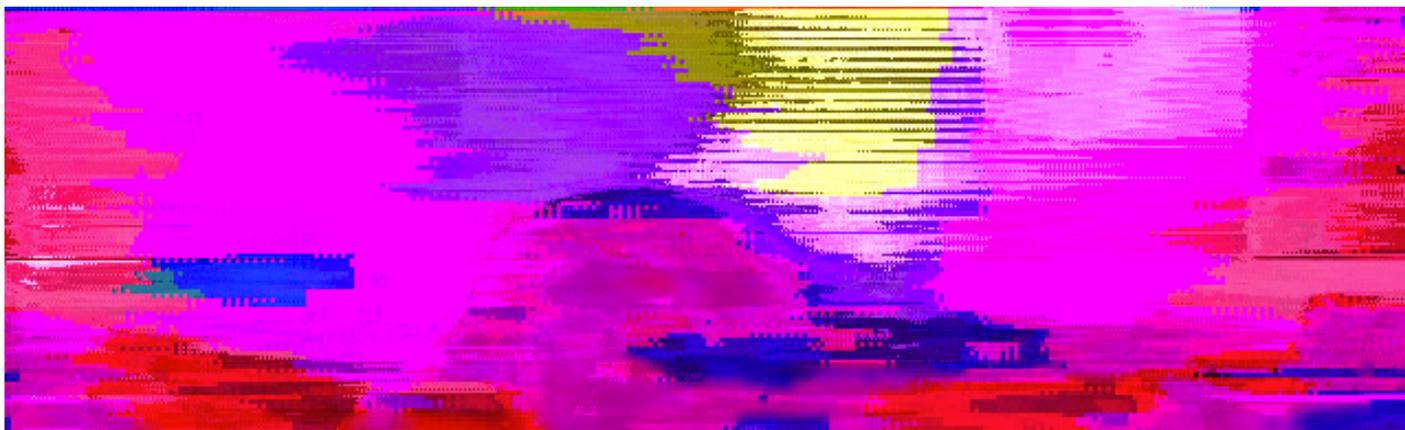
isso não é glitch, mas mostra que vemos pouco do que realmente está presente ali na nossa tela

Para o corrompimento em alguns programas como Audacity, que é um programa de edição de áudio, usamos os arquivos nos seus formatos RAW (cru). Se temos uma foto em formato RAW, como .bmp, podemos abrir em um programa de áudio porque o mesmo será capaz de ler o arquivo em forma de ondas.

Imagine que uma foto seja a palavra “pão”. Nós, aqui no Brasil, falamos pão, mas a mesma coisa no Estados Unidos, por exemplo, é chamado de “bread. É a mesma coisa. Nos dois casos são pães, mas em línguas diferentes.

Abrir uma foto em um programa de edição de áudio nada mais é do que traduzir o arquivo para outra linguagem. O programa vai abrir a foto e ler em formato de ondas, igualzinho a uma música.

Basicamente é como se fizéssemos o processo contrário do pão. Des-assar: separar a farinha, água e o sal e transformamos, com os mesmos ingredientes, em um pão com outra forma. Se começou com um pão francês, depois de desconstruir e modificar a forma dos mesmos elementos, virou uma baguete. Mesmos ingredientes, diferentes formatos e proporções.



AUDACITY - TUTORIAL

Imagem 1: Imagem original

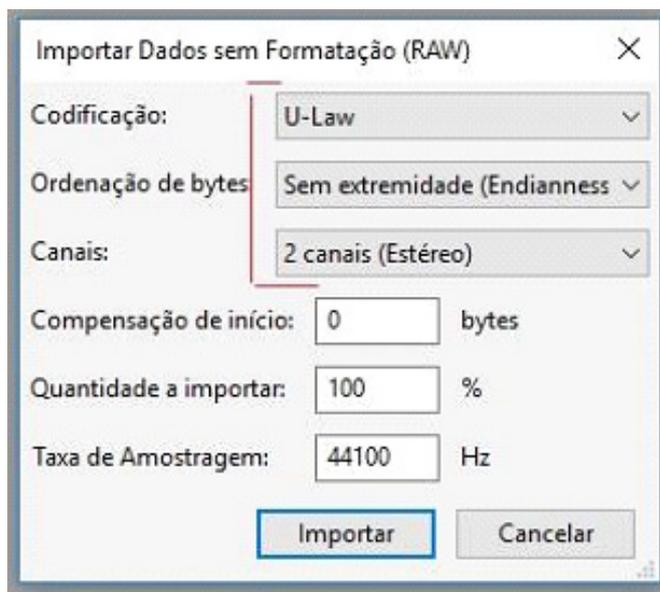


O primeiro passo para editar uma imagem no Audacity é alterar o formato para .bmp - isso pode ser feito no Paint ou Photoshop ou qualquer outro programa de edição de imagem - e depois abrir no editor de áudio.

[passos para importar imagem em formato raw no programa Audacity]

Arquivo > importar > áudio > sem formatação (RAW) > seleciona a imagem

Em seguida esta caixa de configurações irá abrir e as opções tem que estar exatamente como na imagem:



obs.: Só é possível editar imagens com a codificação A-law ou U-law, as outras configurações [Ordenação de bytes] & [Canais] pode ser qualquer opção. Eu prefiro a estética da mesa de trabalho que irá aparecer com as configurações acima`

Imagem 2: Imagem importada no programa de áudio.

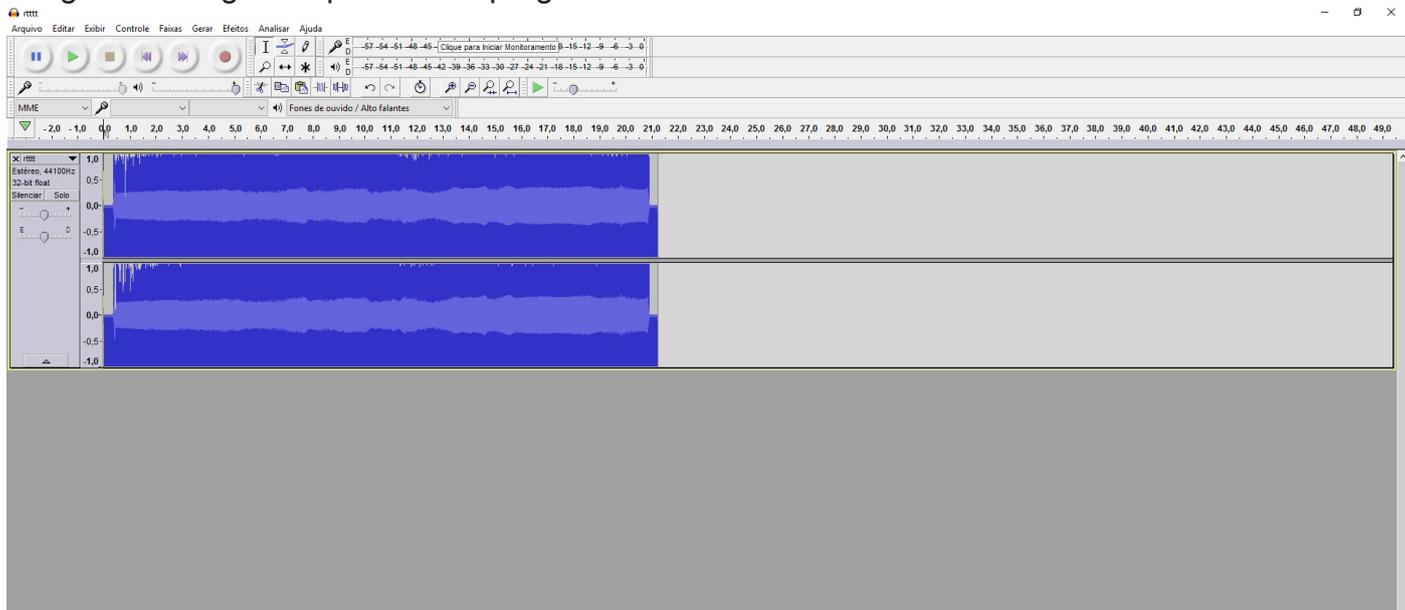


Imagem 3: seleção da parte a ser editada.

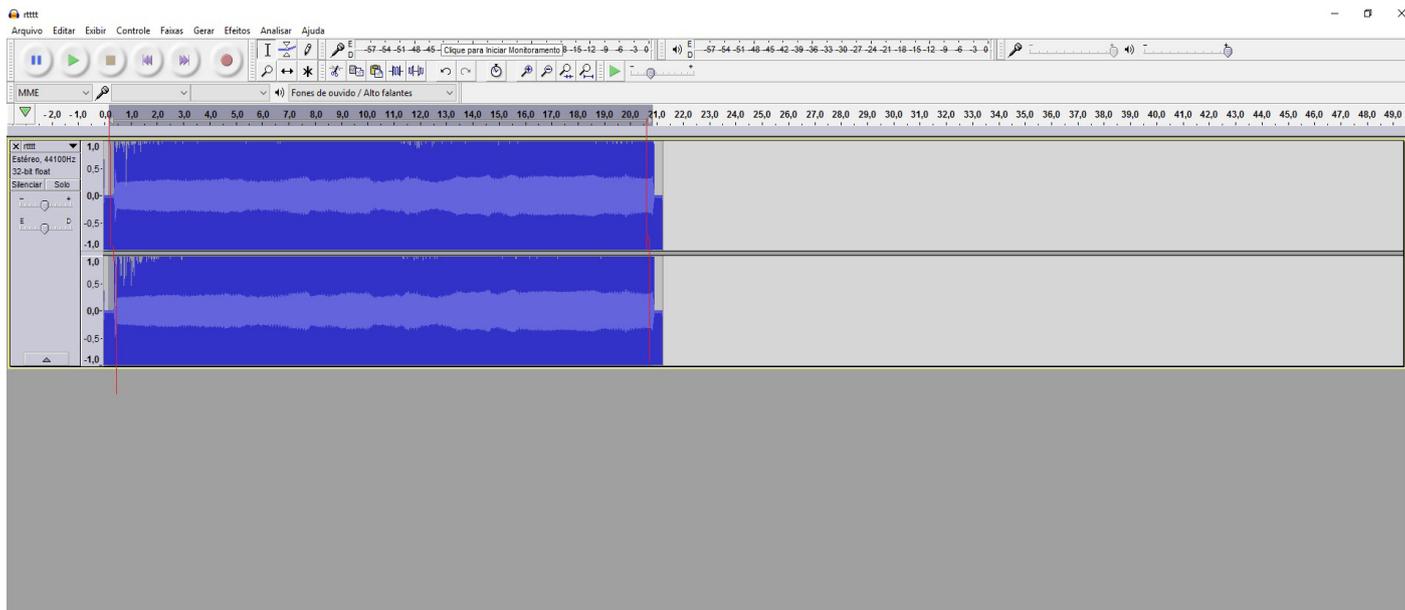
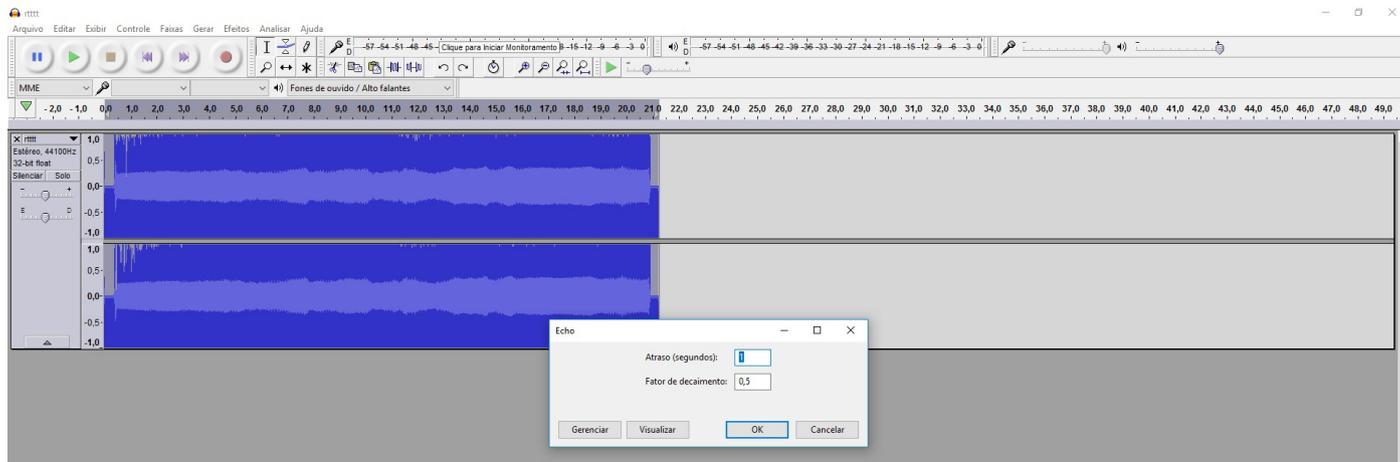
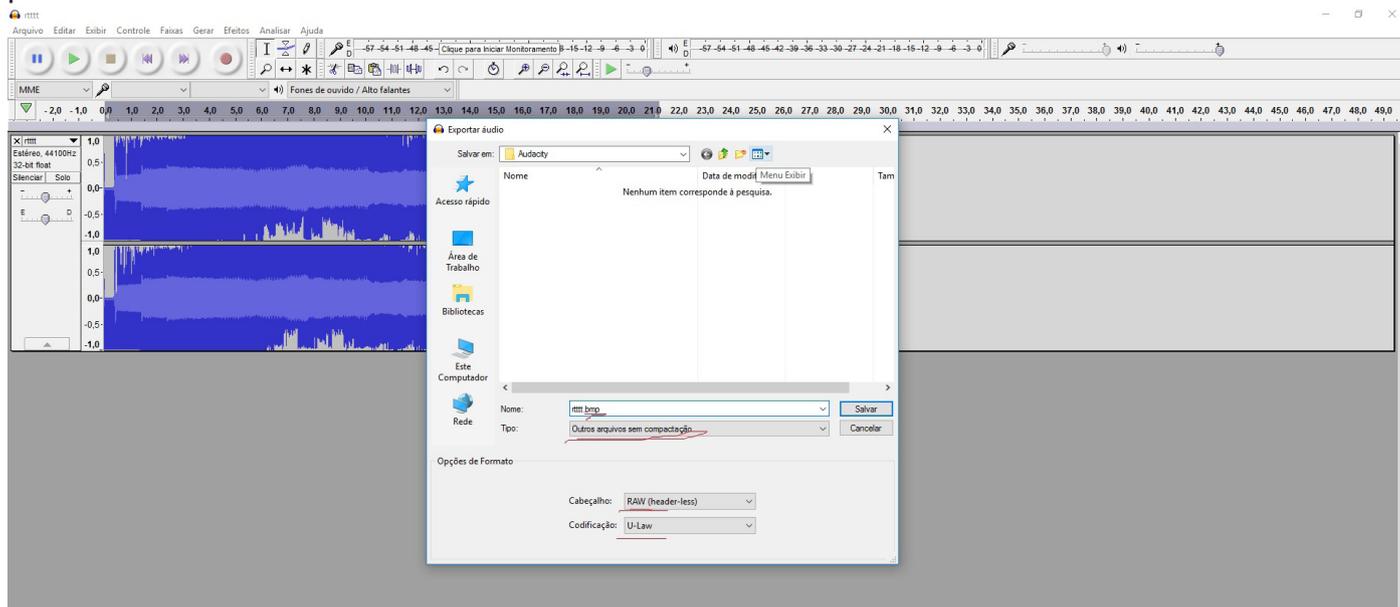


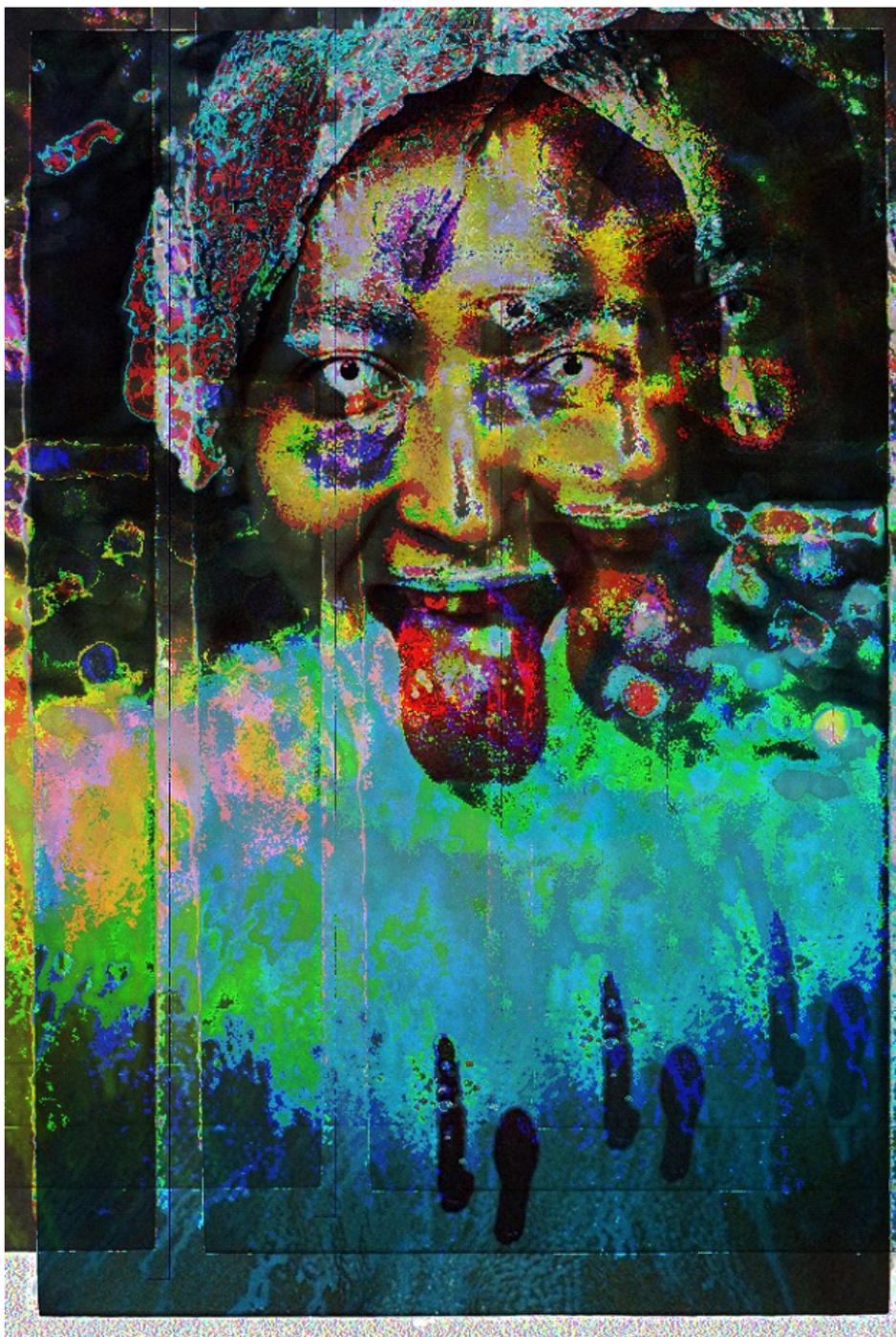
Imagem 4: seleção do efeito “eco” e propriedades.



_____ exportando a imagem

Imagem 5: efeito já aplicado. Como a foto que você está editando foi importando como “áudio sem formatação”, na hora de exportar você tem que especificar com qual tipo de arquivo quer trabalhar além das configurações específicas ressaltadas; por isso adiciona-se “.bmp”, no caso de imagem, para que consiga exportar direito. É como se fosse a tradução para alguém de fora do país.

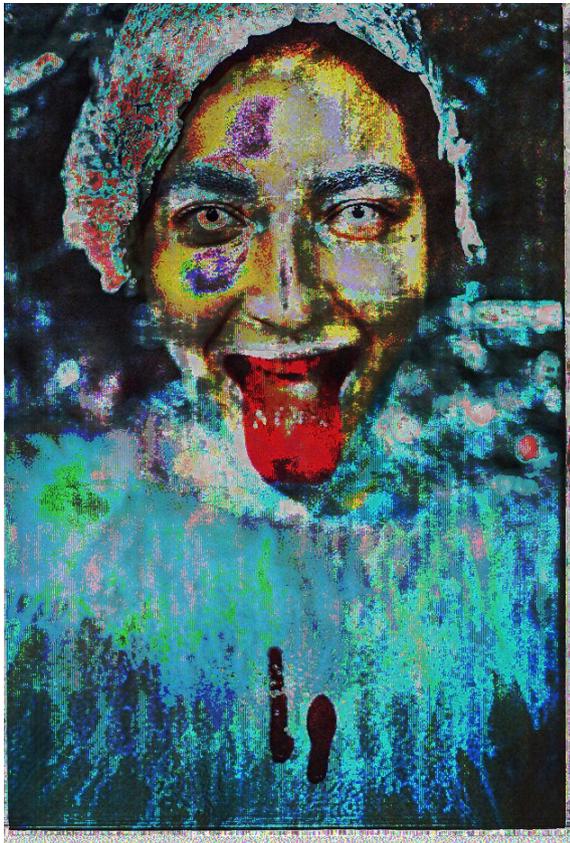




É possível prever mais ou menos o que acontece em uma imagem se você tem familiaridade com o que os efeitos do programa faz nos áudios.

 explicando

Eco é quando, ao gritar, por exemplo, ouvimos nosso grito mais uma vez, mais uma vez, mais uma vez, até ficar tão fraco que não ouvimos mais. É a repetição daquele elemento. Do mesmo jeito que o eco age no som, repetindo o que foi falado várias vezes, ele repete alguns elementos da imagem também. Não é possível prever muito bem o que o programa vai alterar na imagem por não conseguirmos ver todos os elementos do arquivo, não sabermos se o efeito “eco” lê a luz, as cores, os relevos, as texturas da imagem. Vários efeitos podem ser aplicados nas imagens editadas no Audacity. A seguir alguns exemplos:



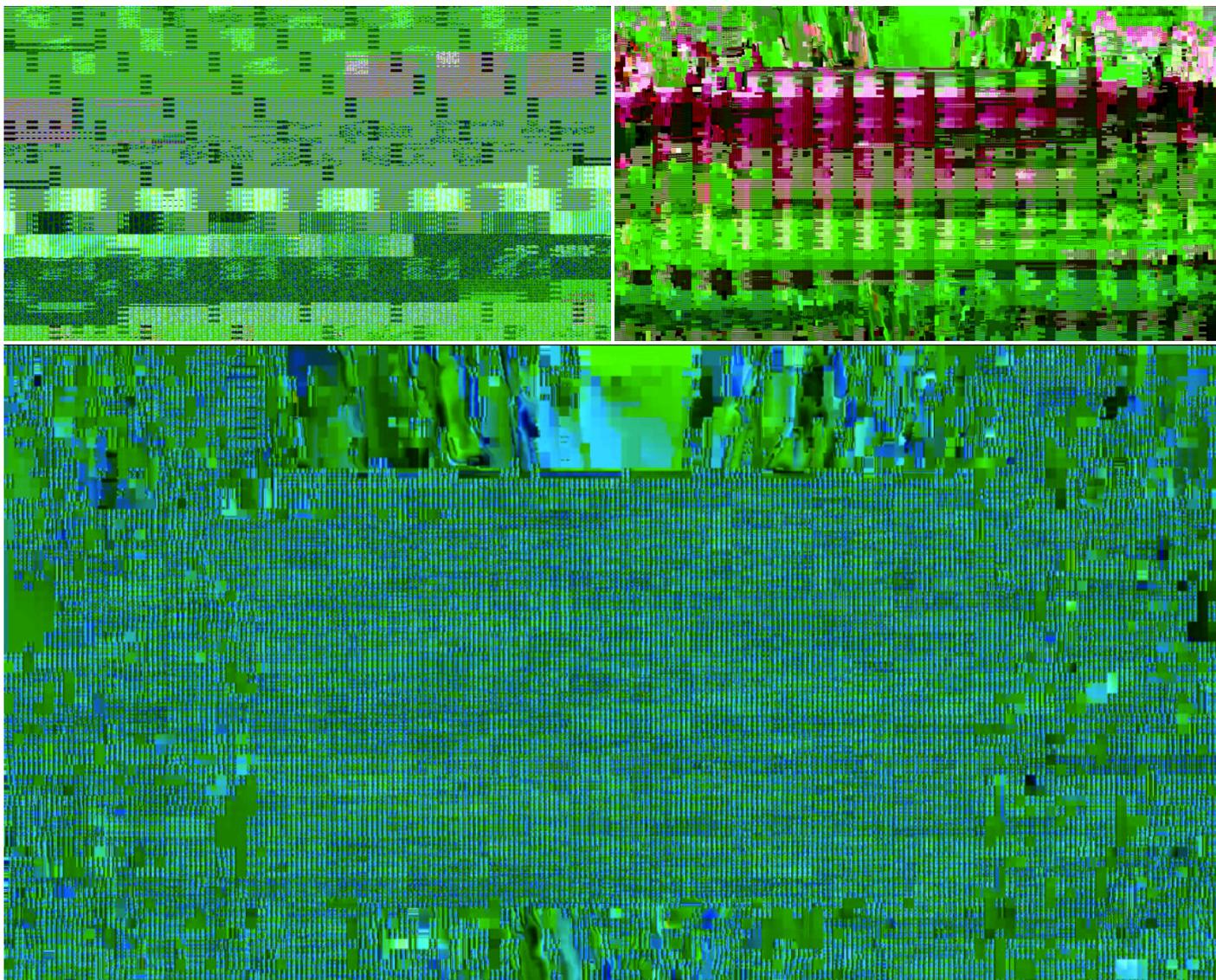
reverb



whawha

Apesar dos resultados glitch ser aleatórios, o processo é específico, mas depois de decorado o caminho passa a ser intuitivo.

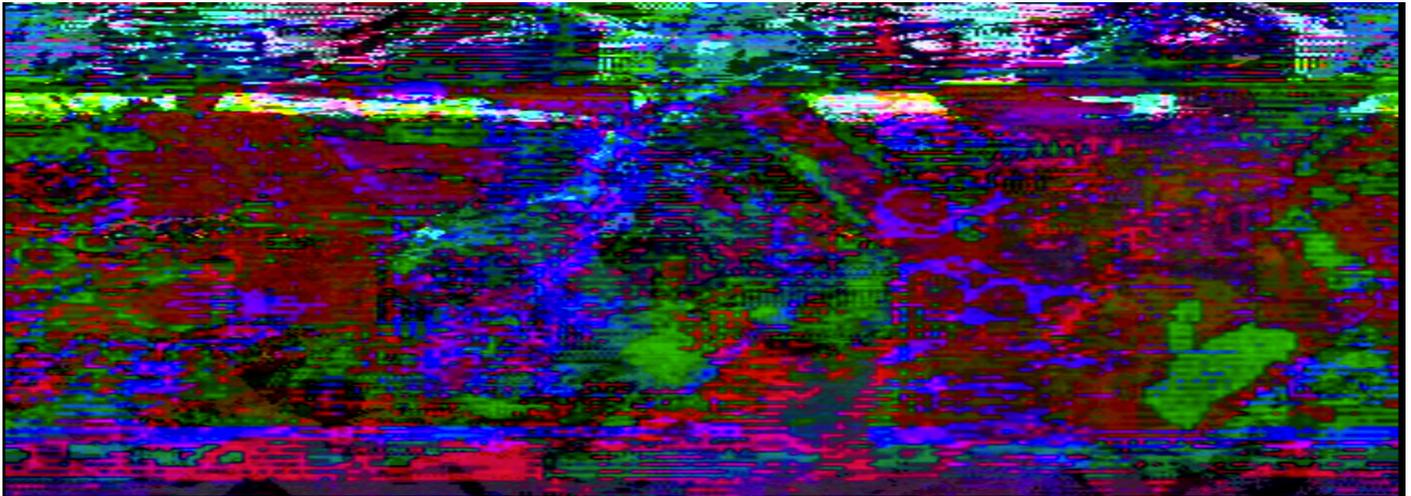
Para trabalhar com vídeos no Audacity o processo é muito parecido. As diferenças são: o formato do vídeo tem que ser .avi se quiser adicionar efeitos diversos e, ao invés de U-law, selecionar a opção A-law na codificação da imagem. _com vídeos não funciona outro tipo de codificação_. Ao exportar, segue raciocínio da foto, colocar .avi depois do nome que quiser dar pro arquivo.



É possível editar vídeos em formato .mp4 no Audacity, mas qualquer manipulação corrompe totalmente o arquivo, então é simplesmente importar e exportar que já dá um resultado interessante.

Outro programa utilizado na pesquisa foi o Avidemux. Este programa não tenta corrigir os arquivos danificados ao reproduzir, então ele é perfeito para assistir os danos causados pelo glitch. Ele também é o programa mais fácil para converter vídeos em .avi.

I. Print de vídeo editado no Audacity, depois da conversão no Avidemux.



PROCESSING

Outro programa que foi foco da pesquisa é o processing. A versão que usei foi a 2.2.1.

“Processing is a flexible software sketchbook and a language for learning how to code within the context of the visual arts.”
FRY, Ben and REAS, Casey. Online.

O Processing é um software de programação onde trabalha-se através dos códigos para fins artísticos. Já tinha entrado em contato com o programa anteriormente, mas não conhecia o glitch. Como é um programa de códigos, é possível abrir um terceiro software dentro do programa, por exemplo.

Existe um site chamado Glitch Hub onde você pode baixar diversos softwares e pré-definições de animação criados por programadores tanto para o estudo estético como tecnológico.

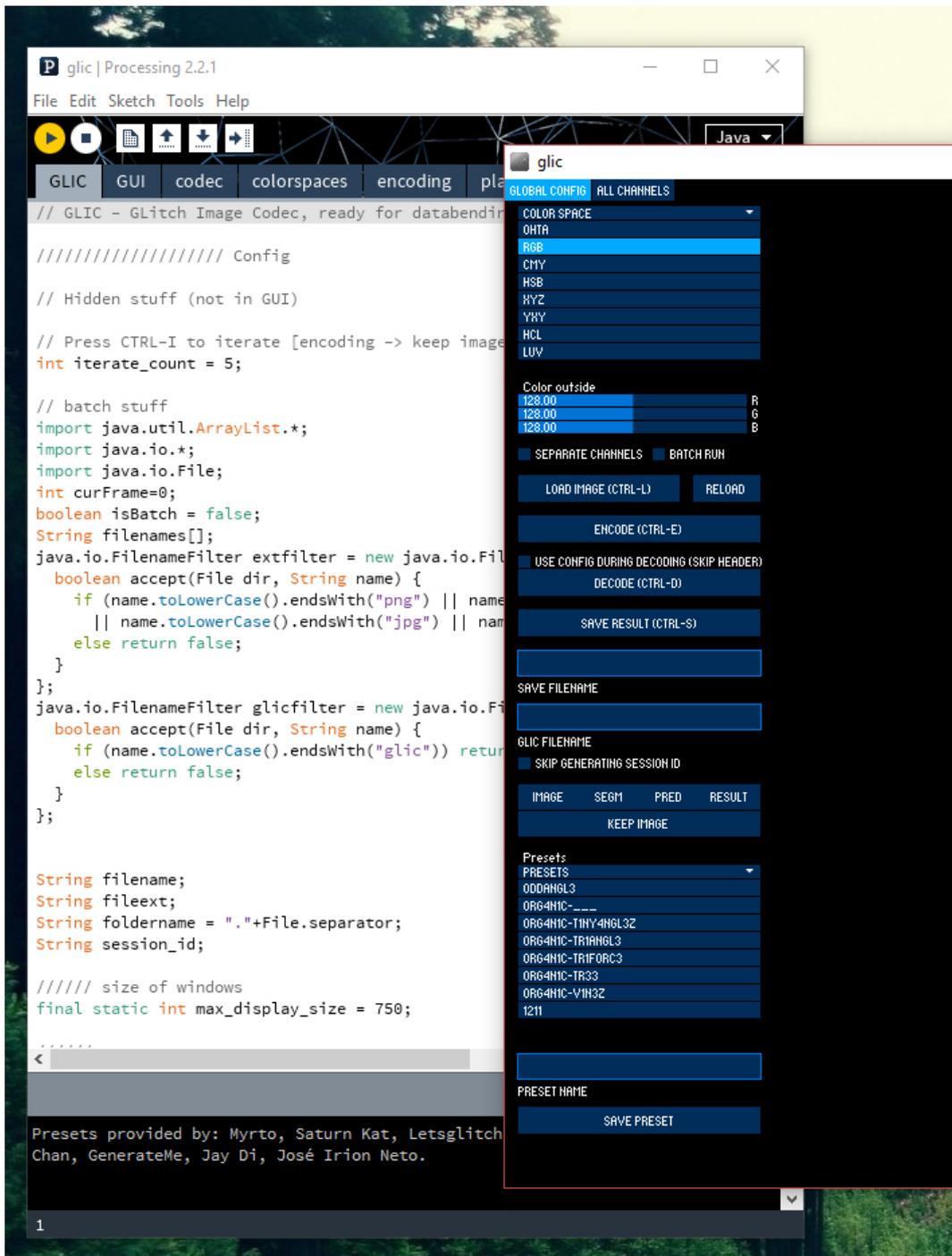
“GitHub is home to over 20 million developers working together to host and review code, manage projects, and build software together”
(Online)

Nos estudos com o Processing eu usei um código e um código somente: GLIC. É um código que se cola no campo do Processing e abre um segundo software de edição de *glitch*.

Conheci esse código através do artista Daniel Vasconcelos _citado anteriormente através de prints_ que me colocou em um grupo do Facebook que inclusive tinha o nome do código, e lá tem uma coletânea de imagens editadas a partir desse código por outros artistas.

Fui atrás da pessoa que criou e achei facilmente: Tomasz Sulej. O post fixo do grupo era dele. O post fixo do grupo era o código. Foi só juntar dois neurônios e mandar uma mensagem. Nessa mensagem eu pedi para ele me contar quem é ele, qual sua relação com *glitch* e como e por que criou o GLIC. Ele leu, topou responder as perguntas e achou razoável a maneira como iria citá-lo _como citei o Daniel também: através de prints_ e pediu um tempo.

Passado esse tempo....



“Ok, de volta à vida. A história por trás do GLIC é simples: eu estava curioso em saber como os efeitos de databending apareciam em imagens/vídeos. Porque são como são? Por que são diferentes? Como revelá-los de maneira mais controlada. Isso foi cerca de dois anos atrás, quando comecei a pesquisar. O principal objetivo dos meus esforços era encontrar todos os passos, algoritmos, métodos de compressão de imagem/vídeo. Coletá-los, achar as partes em comum e reconstruir no meu próprio código. Como mencionei, domorou dois anos de estudo, leitura de códigos, papéis, livros, tentando modificar os codecs já existentes. Eu estava escrevendo umas partes preliminares do código para tentar alguns passos isolados do processo todo de compressão/decompressão.

E, na verdade, percebi que só 3 elementos eram cruciais: decomposição para blocos, algoritmos de previsão e transformação.

Eu apresentei os resultados finais ano passado durante a Fu:ar 2016 em Zagreb, e depois escrevi a versão final do GLIC para todos.

GLIC é codificação de imagem onde é possível escolher os parâmetros dos passos, escolher diversos algoritmos para todos os passos e revelar uma grande variedade de artifatos “fora da caixa”. Parece que eu criei algo novo porque é muito difícil imitar artifatos que já são conhecidos (é difícil conseguir algo similar ao databend de jpg, .png, webp no GLIC).

E é essa a história.

O que mudou? Para usuários comuns: é outro esperimento de codec com algumas coisas únicas. Para pesquisadores: é um aglomerado de conhecimento sobre como codes geralmente funcionam e quais passos fazem o que.

SEG 09:58

Ok! Back to living 🙌 The history behind GLIC is quite simple. I was curious how image/video databending artifacts appeared. Why they look as they look? Why are they different? How to reveal them in more controlled way. It was about two years ago when I started to research. The main goal of my efforts were to find all steps, algorithms, methods of image/video compression. Collect them, find common parts and reconstruct in my own code.

As I mentioned it took about 2 years of reading code, papers, books, trying to modify existing codecs. I was writing some preliminary parts of code to try some steps isolated from the whole compression/decompression process.

And actually it came out that only 3 elements are crucial: decomposition into blocks, prediction algorithms and transforms.

I presented final results last year during Furbar 2016 in Zagreb, and later I've written final GLIC code for all 🙌

GLIC is image codec where you can parametrize steps, choose several algorithms for every step and reveal very wide range of artifacts out of the box

Looks like I created something new, because it's really hard to mimic already known artifacts (it's hard to get something similar to databent.jpg, .png, webp in glic)

So this is the story 🙌

What it changed? For typical users: it's another codec to experiment with and get some unique stuff. For researchers - it collects the knowledge about how generally codecs work and what steps are doing what. 🙌 1

and finally, who am I?

Actually glitch is my hobby. I need to keep my mind working. I have math and computer science education but for years I don't do this for money. On daily basis I work for IT company as a manager.

I live in Poland, Warsaw

That's all!

You sing very nice. I like Brazilian stuff, mostly older bossa (Baden Powell, Jobim, Bonfá, Moraes) but also listened to Ava Rocha or Lara Renno.

LuManzin sounds great.

Tomaz Sulej
Online há 1h

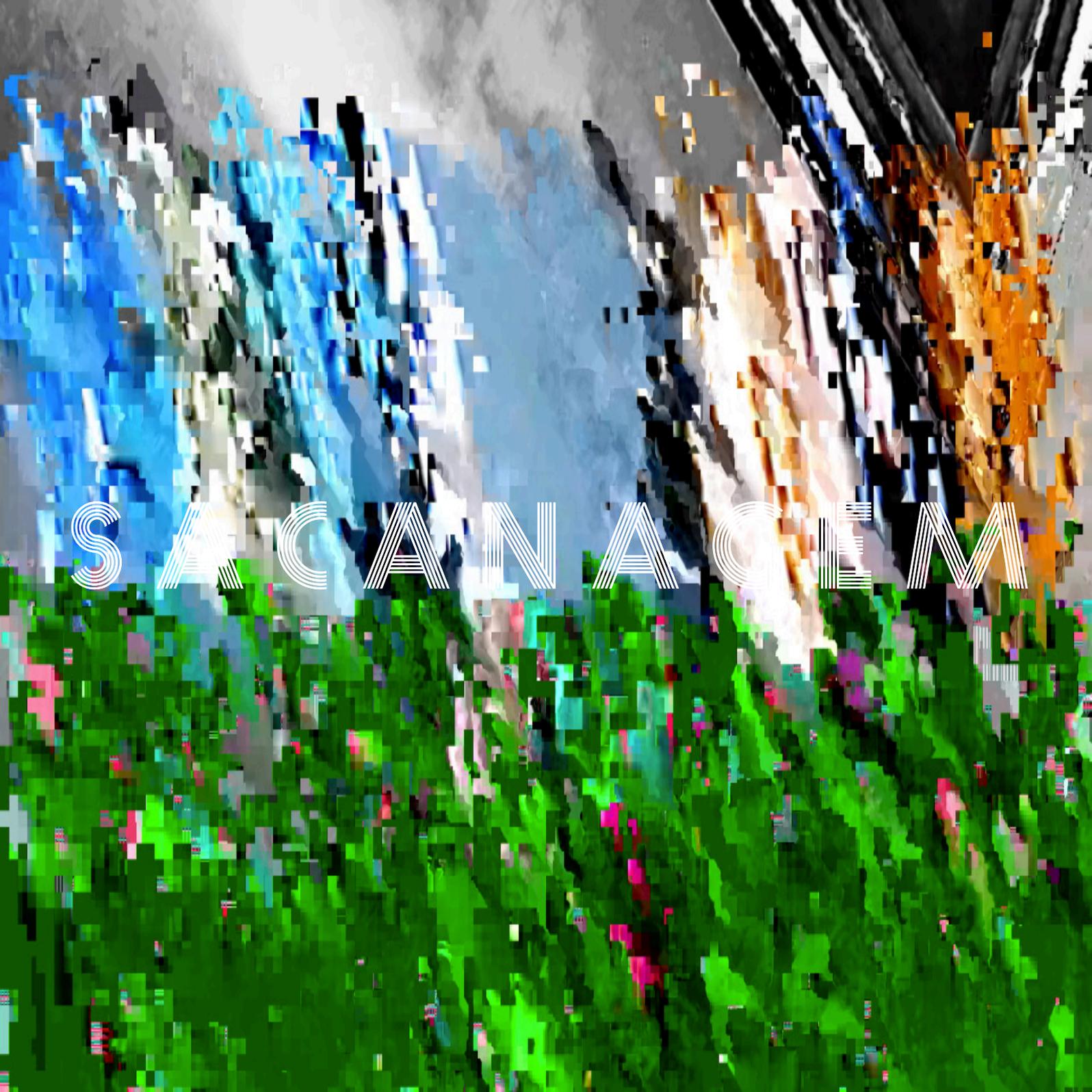
Opções

- Procurar na conversa
- Editar apêditos
- Alterar cor
- Alterar emoji
- Notificações

Perfil do Facebook
<https://facebook.com/tsulej1>

E, finalmente, quem eu sou? Na verdade glitch é meu hobb. Eu preciso manter minha cabeça funcionando. Eu tenho diplomas de matemática e Ciência da Computação, mas há anos não trabalho com isso. No meu dia-a-dia trabalho como gerente de uma empresa de T.I. Eu moro em Warsaw, na Polônia. É isso!”

- Tomasz Sulej, criador do GLIC.



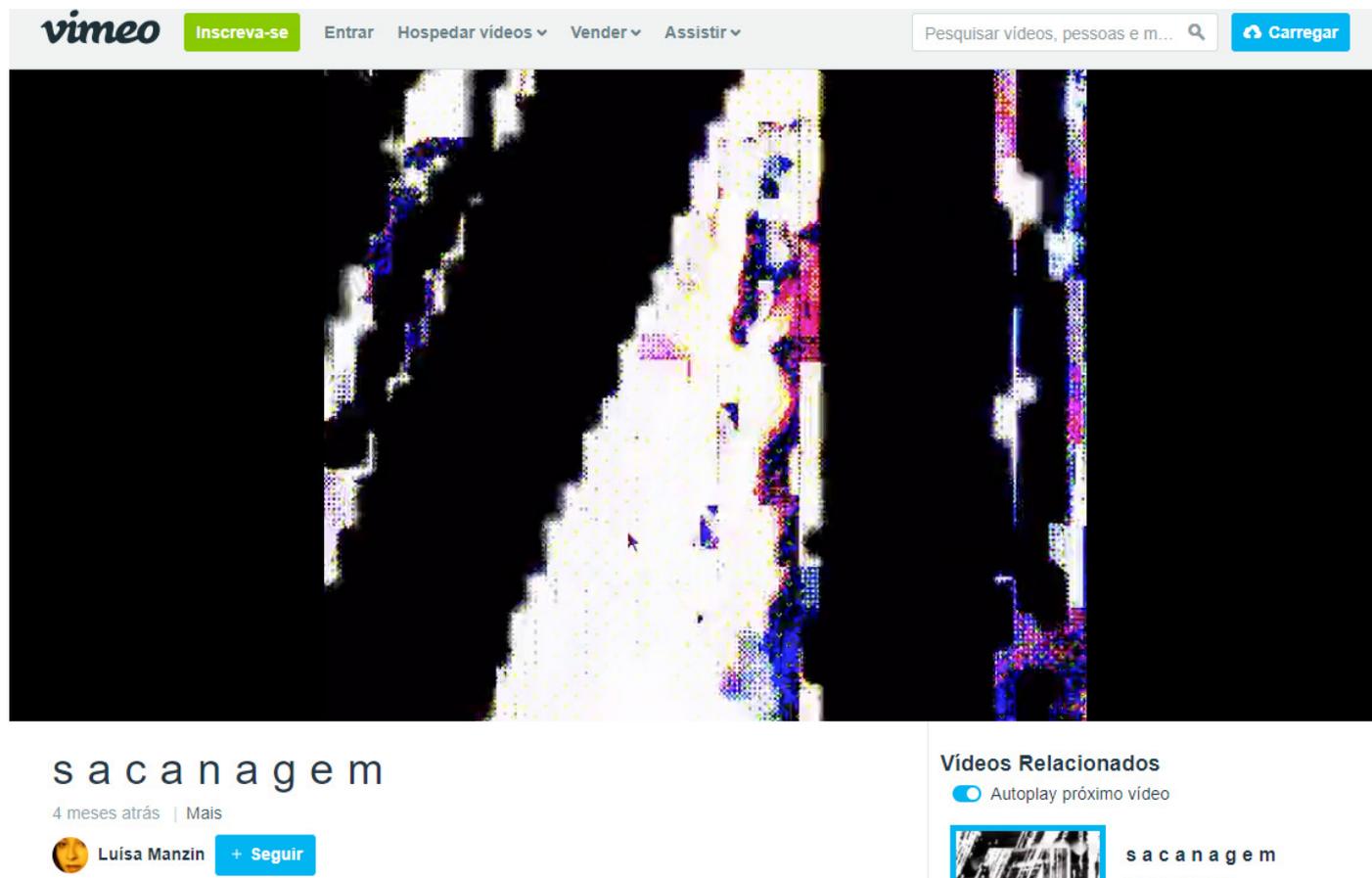
SACRAMENTO

O produto deste estudo é o vídeo `_s a c a n a g e m_` com estética Glitch com trilha sonora de ruído. Foi feito com imagens de uma viagem que eu fiz para praia com duas amigas minhas e a trilha é composta por ruídos. Trabalho com imagens estáticas principalmente, frames de vídeos anteriores `_mas não publicados_` e vídeos propriamente ditos.

O vídeo é totalmente abstrato e nele foram usadas diversas técnicas, predominantemente as do Audacity. O objetivo com ele é causar qualquer desconforto, seja pelas imagens muito coloridas, sejam os cortes rápidos, seja o ruído incessável ou o silêncio do começo.

este vídeo foi produzido em Julho de 2017 durante uma viagem para Natal - RN. E passou, junto de duas imagens, na curadoria da revista eletrônica Utsanga.it, de pesquisa e crítica de linguagem

O vídeo está disponível no Vimeo.



The screenshot shows the Vimeo interface. At the top, there's a navigation bar with the Vimeo logo, a green 'Inscreva-se' button, and links for 'Entrar', 'Hospedar vídeos', 'Vender', and 'Assistir'. A search bar contains the text 'Pesquisar vídeos, pessoas e m...' and a blue 'Carregar' button. The main video player area displays a highly distorted, glitched frame with vertical bands of black, white, and vibrant colors like blue, purple, and red. Below the video, the channel name 'sacanamagem' is displayed in a lowercase, spaced-out font. Underneath the name, it says '4 meses atrás | Mais'. To the left of the channel name is a profile picture of a woman and the name 'Luísa Manzin' next to a blue '+ Seguir' button. On the right side, there's a section titled 'Vídeos Relacionados' with a toggle switch for 'Autoplay próximo vídeo' which is currently turned on. Below this, a small thumbnail of another video is visible with the channel name 'sacanamagem' overlaid.

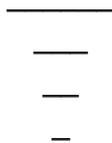
The background is a complex, abstract composition of overlapping, curved shapes in shades of red, white, and black. The shapes are pixelated and layered, creating a sense of depth and movement. The word "ADICIONAIS" is centered horizontally across the middle of the image.

ADICIONAIS

No meio do processo do TCC minha carreira artística como cantora e produtora executiva começa a emergir.

Em setembro do mesmo ano, 2017, ministrei meu segundo Workshop com tema “Introdução à Glitch Art e técnicas de edição”, nos programas Audacity e Notepad++, na Semana de Mídias da PUC-SP.

o primeiro workshop foi em 2015 sobre criação de conteúdo visual no Colégio Sion



A *glitch art*, principalmente por ser um tipo de arte que trabalha com a estética digital está em alta e acredito que esse seja um movimento exponencial. O uso e a apropriação de qualquer tipo de glitch _puro, alike ou sim_ está só no começo.

Depois que eu entrei em contato com a estética da *glitch art* percebo que essa nomenclatura e significado pode se expandir para questões inclusivas sociais e pessoais. Quando temos, por exemplo, um ato falho, que é quando nós colocamos no nosso discurso elementos que não fazem parte dele, mas estavam na nossa cabeça naquele momento, isso é um *glitch*. Quando descemos uma escada e estamos desatentos e nosso corpo está preparado para descer mais um degrau, mas não tem e levamos um susto, isso é *glitch*. Quando temos um *dejavù*, isso é *glitch*. Se *pure-glitch* é quando a parte elétrica falha, quando nossa parte elétrica interna, que são as conexões neuronais, falham, nós temos um *glitch-puramente-orgânico*.

/tudo como extensão de tudo, já dizia McLuhan em outras palavras e outro contexto, aqui eu só abranjo.

Glitch também desconstrói a imagem, traz elementos que estão escondidos nos arquivos. Então ele traz uma pergunta que é: somos aquilo que vemos, realmente?

O *glitch* expandiu, de todas as formas, coisas que já estavam em expansão na minha percepção de mundo porque agora eu tenho uma desculpa para achar o não-comum bonito e quem sabe poder me gabar por aí por conhecer técnicas tão simples, mas tão “secretas”.

O *glitch* nunca se repete. Você pode, em alguns casos, salvar pré-definições em aplicativos que você só acha se entra num fórum escondido na internet e pega o código e baixa um software para abri-lo, mas qual a chance de ficar igual ou alterar da mesma forma arquivos diferentes sendo que eles são... diferentes?

Talvez seja meio ambicioso dizer que glitch e a vida, os sentimentos, as experiências tenham alguma relação, mas não vejo uma forma de não relacionar.

Concluindo, o glitch além de ser lindo _as vezes até emocionante_ me trouxe uma calma-caótica-estética e ele aparece na minha vida exatamente num momento de caos.

Mas é um caos bonito.

ass: lulu

BIBLIORAMA

CHAIRLIFT. Evident Utensil - Video-clipe. Disponível em: <<https://vimeo.com/3139412>>. Acesso em: 5 Out. 2016

DOWNEY, Jonas. Glitch Art: Ninth Letter. Disponível em: <[http:// HYPERLINK “http://ninthletter.com/”ninthletter.com](http://HYPERLINK \)> e <<https://jonas.do/assets/essays/glitch-art-jonasdowney.pdf>>.2012. University of Illinois.

DR.JMART. Analog. Disponível em: <<https://www.youtube.com/watch?v=3oY5rRbDzMA>>. Acesso em: 5 Out. 2016
<http://danieltemkin.com/Tutorials>

GAZANA, C. et al. Glitch: estética contemporânea visual e sonora do erro. Disponível em: <http://anpap.org.br/anais/2015/comites/cpa/cleber_gazana.pdf>. Acesso em: 18 Abr. 2017

Glitch Irion, disponível em: <<http://glitch-irion.blogspot.com.br/>>. Acessado em 05/05/2017.

Grupo “Glitch Artists Collective” do Facebook. Disponível em: <<https://www.facebook.com/glitchartistscollective/>>. Acesso em: 5 Set. 2016

Grupo “Glitch Artists Collective: Tools Time” do Facebook. Disponível em: <<https://www.facebook.com/groups/GACToolTime/>>. Acesso em: 23 Set. 2016

IAN, Cheng. bbraattss. Online video clip , 2013. Acesso em: 23 Mar 2017

KVLEWIS. Is Glitch Art actually glitch?. Disponível em: <<http://samplereality.com/davidson/dig101/2014/10/19/is-glitch-art-actually-glitch/>>. Acesso em: 24 Mar. 2017

MACHADO, Arlindo, Arte e Mídia. Disponível em: <<https://revistas.pucsp.br/index.php/galaxia/article/viewFile/1285/785>>. São Paulo: Jahar, 2007.

MCGEE, Ryan Michael et al; Voice of Sisyphus: an image sonification multimedia installation. Disponível em: <https://www.mat.ucsb.edu/Publications/McGee_ICAD_2012.pdf> 22 Jun. 2012. Atlanta, EUA.

MORADI, Iman. Glitch aesthetics. Huddersfield: TUH, 2004. Dissertação de mestrado, The University of Huddersfield. Disponível em: <<http://www.ocu-lasm.org/glitch/download>>. Acesso em: 28 Abr 2017.

New Media, disponível em: <<http://www.newmedia.org/what-is-new-media.html>>. Acessado em 05/05/2017

RUSSOLO, Luigi. The Art of Noises. Disponível em: <<http://www.ubu.com/papers/russolo.html>>. Acesso em: 28 Abr. 2017.

UNKOWN. Frames, Fields, Pictures (I, P, B). <<http://www.bretl.com/mpeghtml/pixtypes.HTM>>. Acesso em 13/09/2017.

WAY SPURR-CHEN, [TUTORIAL] Make Video Glitch Art: Houw to Datamosh in Plain English. <<http://forum.glitchet.com/t/tutorial-make-video-glitch-art-how-to-datamosh-in-plain-english/36>> 11 Ago 2015. Acessado em: 13/09/2017.

