Databending

Vižintin, Petra

Undergraduate thesis / Završni rad

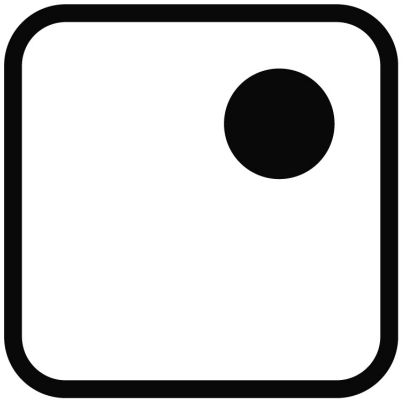
2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Graphic Arts / Sveučilište u Zagrebu, Grafički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:216:667365

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-17**



Repository / Repozitorij:

Faculty of Graphic Arts Repository

SVEUČILIŠTE U ZAGREBU
GRAFIČKI FAKULTET

ZAVRŠNI RAD

Petra Vižintin

Sveučilište u Zagrebu
Grafički fakultet

Smjer: tehničko-tehnološki

ZAVRŠNI RAD

DATABENDING

Mentor:
dr.sc. Maja Strgar Kurečić

Student:
Petra Vižintin

Zagreb, 2018.

Rješenje o odobrenju teme završnog rada

SAŽETAK

U ovom će se završnom radu objasniti kako se slikovna datoteka (digitalna fotografija) može uređivati u programima koji nisu namjenjeni za uređivanje slikovnih datoteka, a to su teksteditori, audioeditori, hexeditori te Blender kao predstavnik programa za 3D animaciju. Ovakvim se postupcima dobiva glitch kao rezultat nepodudaranja formata i kôda za koji je program namjenjen te formata i kôda datoteke. Objasnit će se na koji se način može "prevariti" program da ipak otvori slikovnu datoteku, iako je, recimo, namjenjen uređivanju auditivnih datoteka. Objasnit će se i razlozi dobivanja glitcha, što je to što u programu izazove poremećaj, kao i razotkriti računalni sloj slikovnih datoteka, koji se na prvi pogled uopće ne doživljava jer je korisnik fokusiran na kompoziciju i objekte prikazanih na fotografiji.

KLJUČNE RIJEČI: databending (slikovnih datoteka), kôd, bitmapa, glitch, korupcija

SADRŽAJ

1 . UVOD 1
2. TEORIJSKI DIO 2
2.1 Digitalna umjetnost i novi mediji 2
2.2 Interpretacija i umjetnost greške 4
2.3 Autori i njihova djela 5
2.4 Osnovni pojmovi 11
2.4.1 Rezolucija 11
2.4.2 Interpolacija 12
2.4.3 Bit 12
2.4.4 Kompresija 12
2.4.5 Kanali boja 13
2.5 Databending 13
2.5.1 Wordpad 14
2.5.2 HEX 15
2.5.3 Sonifikacija 17
2.5.4 Blender 18
3. EKSPERIMENTALNI DIO 19
3.1 Teksteditori 19
3.1.1 *Wordpad* 19
3.1.2 Notepad++ 21
3.2 Hexeditori/Notepad++ 22
3.2.1 Hexeditor 22
3.2.2 Notepad++ 24
3.3 Audacity (Sonifikacija) 24
3.4 Blender OBJ Glitch 30
4. RASPRAVA I ZAKLJUČAK 33
5. LITERATURA 37

1 . UVOD

Glitch se može definirati kao kratkotrajna pogreška u sustavu, neočekivano zastajkivanje programa koje najčešće nestaje samo od sebe i koje se nije moglo predvidjeti, a *databending* kao proces manipulacije jedne datoteke u programu za koji ona nije namjenjena pa se tim činom izazivaju i dobivaju neočekivane pogreške. Takvi radovi najčešće spadaju pod *Glitch Art* ili *glitch* umjetnost, umjetnost greške. Ni *glitch*, ni *databending* nemaju svoje inačice u hrvatskom jeziku, a nije niti pretjerano poznat kao "grana" umjetnosti. Pa iako spada pod neku vrstu umjetnosti, *databending* bi se mogao koristiti i kao dodatan alat koji bi služio dizajneru, ali i kao zanimljiviji način učenja pozadinske logike i opcija određenih programa, iako je ovo potonje subjektivnije prirode. *Glitch* vraća u prvi plan ono što je stavljeno i skriveno u pozadini.

Tema ovog završnog rada izabrana je kako bi se produbilo autorovo znanje, ali i zainteresiranost za istu. Mada, na engleskom tržištu već postoji (barem jedna) knjiga koja se tiče korištenja *databendinga* u dizajerske svrhe, autorova zainteresiranost dolazi i sa umjetničke strane.

U prvoj rečenici definiran je pojam *glitcha*, što bi trebalo otvoriti pitanje, mogu li se rezultati *databendinga*, namjernim i određenim postupcima priređivanja datoteke i njenim "uređivanjem", smatrati *glitchem*? Puristički je odgovor, naravno, ne. No, definira li se *glitch* kao "poremećaj u protoku informacija"[1], mijenja li se tada situacija i odgovor. Kako god bilo, ovaj se završni rad tiče *databendinga*, odnosno procesa, a ne *glitcha*, odnosno konkretnog rezultata – budući da za isti postoje druge tehnike dobivanja, kao što je manipulacija slikove datoteke u *Photoshopu* ili konkretnim programima/aplikacijama koje imaju *glitch* filtere koji se primjenjuju na sliku.

[1] https://www.kulturpunkt.hr/content/umjetnost-greske

2. TEORIJSKI DIO

2.1 Digitalna umjetnost i novi mediji

"Danas se nalazimo u situaciji sličnoj onoj iz četrnaestog i devetnaestog stoljeća, kada su tiskarska preša i fotografija izvršile revolucionarni utjecaj na razvoj moderne umjetnosti i kulture, tj. da se danas nalazimo usred revolucije novih medija – pomak cjelokupne kulture ka računalno posredovanim oblicima proizvodnje, distribucije i komunikacije."[2]

Digitalna umjetnost spada pod novomedijsku umjetnost, odnosno umjetnost koja koristi digitalne tehnologije i prakse. Takvu umjetnost još nazivaju računalna ili multimedijska umjetnost. Digitalna umjetnost transformirala je i u sebe ukomponirala neke klasične ("analogne") umjetnosti kao što su skulptura, slikarstvo ili ilustracija/crtež. Kako bi se shvatilo na što se cilja kada se kaže "novomedijska umjetnost" potrebno je definirati što su novi mediji.

Fotografija koja se nalazi na računalu, CD-u ili je na računalu obrađena, smatra se novim medijem, ta ista fotografija isprintana na papiru – ne, a takvih bi se primjera dalo nabrojati pa ipak granicu između novih i stari medija negdje treba povući. Potrebno je definirati principe kojima će se opisati što se smatra novim medijem. Postoje pet principa kojima se može definirati prirodna estetika digitalnog, a to su: numerički prikaz, modularnost, automatizacija, varijabilnost i transkodiranje [1].

1. Numerički prikaz

   Sve medijske datoteke, one stvorene na računalu ili one konvertirane iz svoji analognih oblika sačinjene su od digitalnih kôdova, tj. imaju svoj numerički prikaz (binarni). Može se zaključiti da se svi novi mediji mogu opisati matematički i mogu se obrađivati algoritmima.
   Originalni oblik novih medija je numerički. Digitalizacijom analogni mediji postaju novomedijski.

[2] Lev Manovič, Jezik novih medija, Multimedia Clio, 2015., str.61 [1]

2. Modularnost

"Fraktalna struktura novih medija" [1]; fraktali na različitim razinama zadržavaju istu strukturu pa tako i medijske datoteke predstavljene kao svoji diskretni dijelovi (pikseli, poligoni, ...) imaju svoju modularnu strukuru. Pikseli, iako čine pr. digitalnu fotografiju, zadržavaju svoj identitet i svoju neovisnost. Drugi primjer modularnosti jest taj da datoteke zadržavaju svoju neovisnost, pr. ako se slika ubaci u *Word* dokument. Svim se elementima može pristupiti neovisno o drugima.

3. Automatizacija

Numerički prikaz i modularnost omogućuju automatizaciju operacija što dopušta da se bar djelomično isključi ljudska namjera (*Photoshop* može automatski podesiti kontrast datoteke koja se uređuje).

4. Varijabilnost

Nijedna novomedijska datoteka nije zamišljena tako da je se nakon stvaranja i obrade više neće nadograđivati ili izmjenjivati, sve novomedijske datoteke mogu postojati u bezbrojnim verzijama. Varijabilnost ne bi trebala biti stran pojam pošto se danas neke od aplikacija kao što su Instagram ili Facebook na mobitelima nadograđaju gotovo svakodnevno. "Sada je moguće razdvojiti razine "sadržaja" i sučelja. Moguće je sa istim podacima stvoriti niz različitih sučelja. Objekt novih medija može se odrediti kao jedan ili više sučelja sa multimedijalnom bazom podataka (Freud-Lissitsky Navigator)."[1]

5. Transkodiranje

Digitalizacijom pretvaramo medije u računalne podatke, tu nastaje dualna priroda novomedijske datoteke. Na primjeru digitalne fotografije, na razini prikaza, odnosno onog što vidimo - na ljudskoj razini, na njoj najvjerojatnije prepoznajemo objekte, motive, određeno značenje, i sl... ali ona je računalni dokument pa na toj razini, "ulazi u dijalog sa drugim računalnim datotekama" [1] gdje se govori o veličini datoteke, vrsti kompresije, pikselima, vrijednostima boja, formatima...

Novi mediji funkcionaraju na dvije razine, onoj ljudskoj i onoj računalnoj, samo nalikuju na svoje analogne parove, dok su u suštini zaista nova vrsta medija i nemaju svog povijesnog prethodnika.

2.2 Interpretacija i umjetnost greške

Digitalizacijom objekti postaju izračunljivi, a računalo postaje generator za stvaranje medija. Digitalna greška je "stanje dodavanja, izuzimanja i/ili izmjene gradbenih elemenata u digitalnoj komunikaciji"[3], ako se takva greška namjerno izazove, u sustavima u kojima je izazavana javlja se nepredviljivost, a kao posljedica javlja se *glitch*. Ovaj je proces poznat kao *databending* - veliki broj tehnika kojima se namjerno unosi greška u digitalni kôd datoteke, a rezultat je datoteka s novim estetskim svojstvima. *Databending* je tehnika koja se koristi u *glitch* umjetnosti, umjetnosti koja greške ne interpretira kao nešto loše, već poželjno te se veseli rezultatima koje će ona izazvati. Nasumičnošću unošenja greške u sustav otkriva se računalni sloj medijskih datoteka. *Glitch* je spoj avangardnih praksi i računala, svoje korijenje vuče iz pravaca kao što su kubizam i dadaizam. "*Glitch* kao kritika umjetno stvorenoj nestašici u suvremenom digitalnom sustavu može stvoriti prostor za proaktivnu praksu vizualne umjetnosti."[4]

[3] G/D [IV book], autor: udruga Format C, https://formatc.hr/glitch_databend#I
[4] G/D [IV book], autor: udruga Format C, https://formatc.hr/glitch_databend#I

Circuit bending analogni je pandan *databendingu*, pojam *databendinga* sastoji se od dvije riječi *data* (podaci) i *bending* ("savijanje"), aludira na manipulaciju kôda i dovođenja istog do granice "pucanja".

Ono na što umjetnici greške žele upozoriti jest da se vrijednost podataka krije u njihovom sadržaju, ali i u ljudskoj sposobnosti interpretacije istih, osvještenje o rukovanju podacima i informacijama.

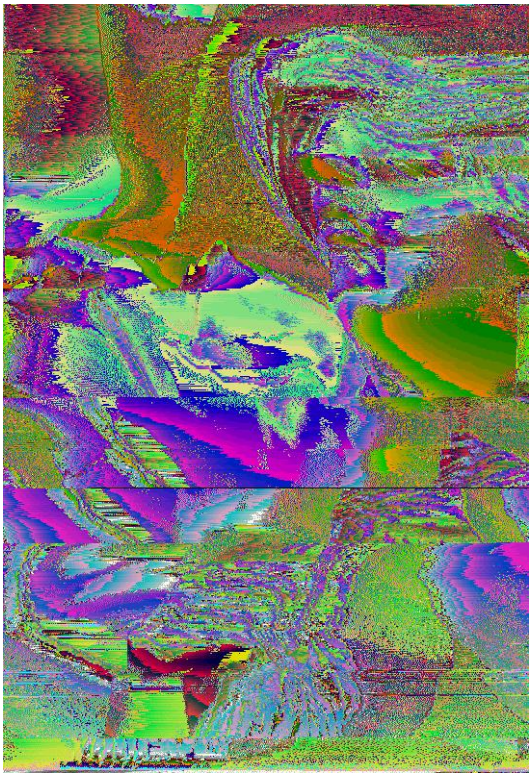
2.3 Autori i njihova djela

Rosa Menkman

Izvor: https://beyondresolution.info

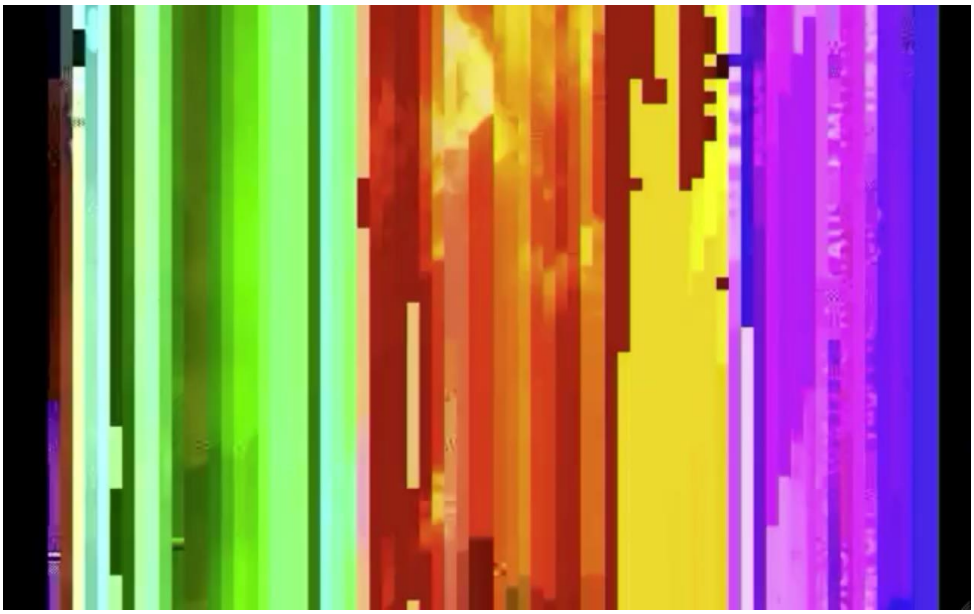
Izvor: https://beyondresolution.info

Domenico Barra

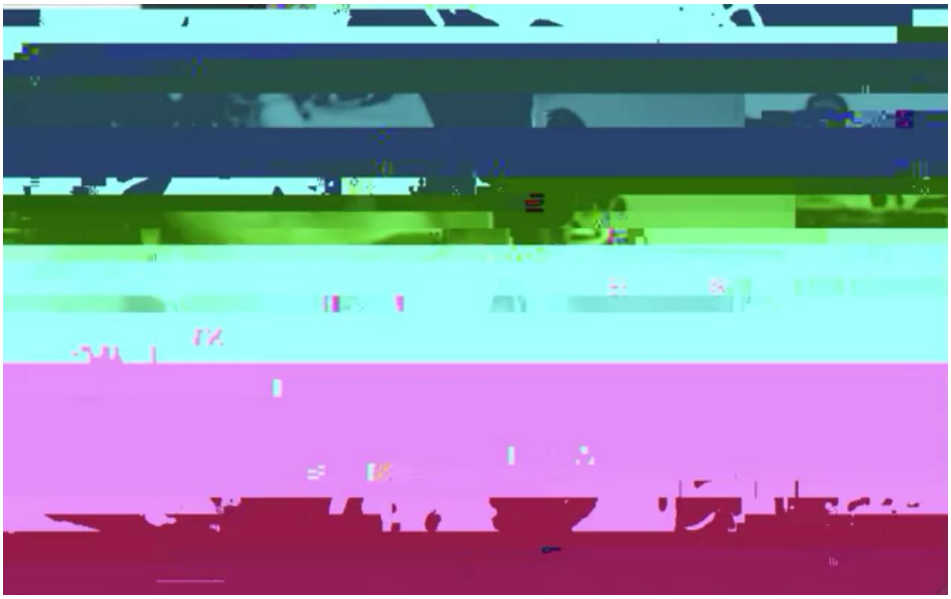


Izvor: http://dadaclub.online/tag/databending/

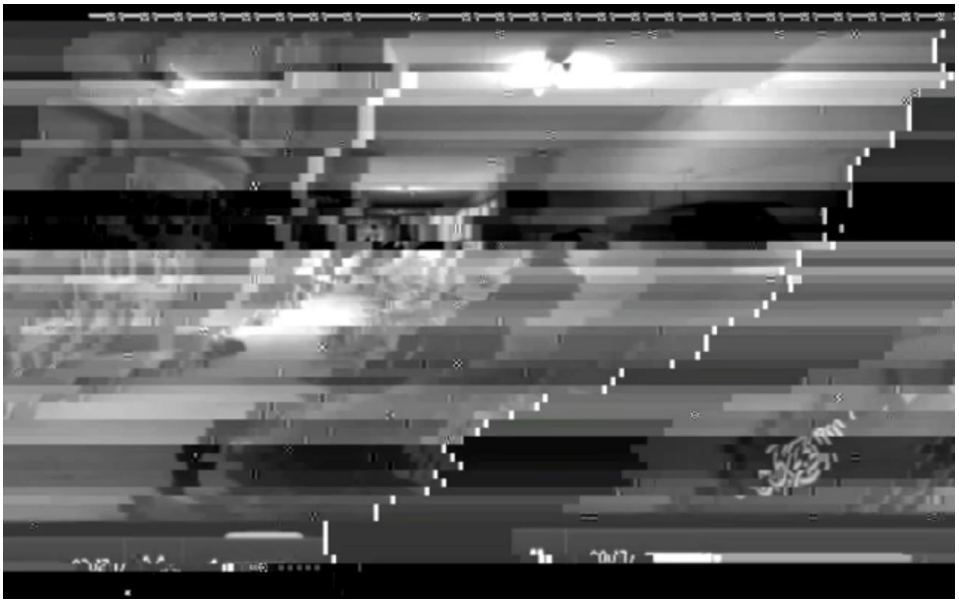
Alex Myers



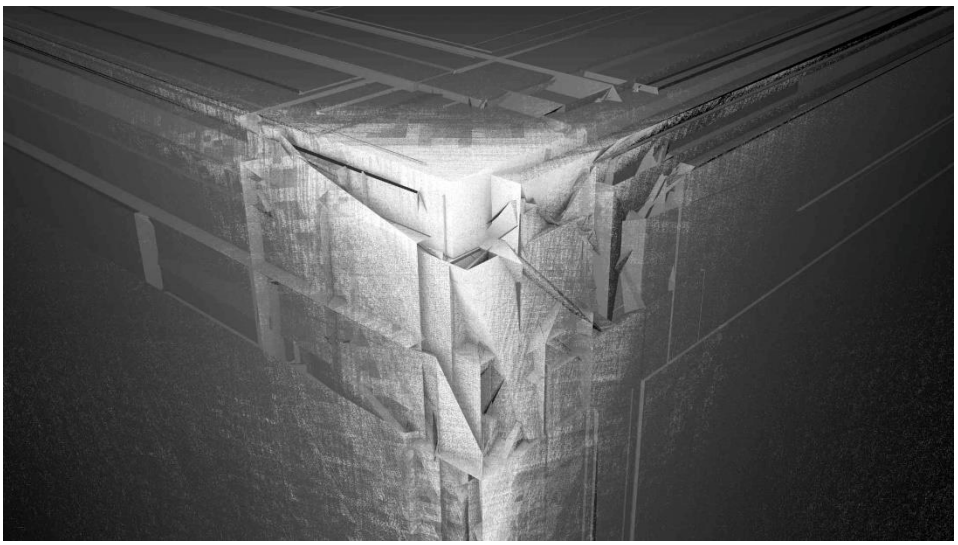
Izvor: http://alexmyers.info/



Izvor: http://alexmyers.info/



Izvor: http://alexmyers.info/



Izvor: http://alexmyers.info/

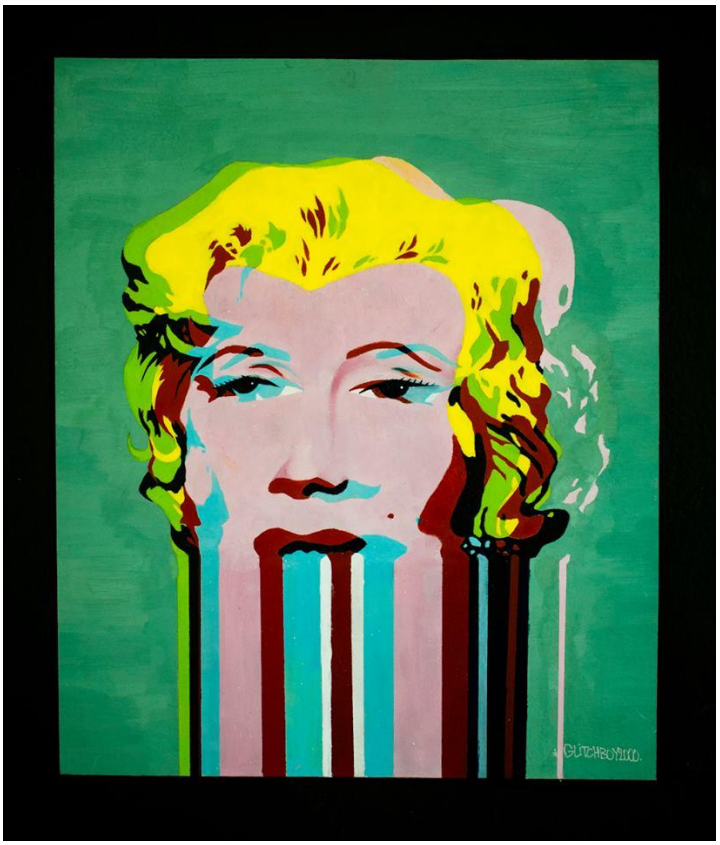
Izvor: http://alexmyers.info/

Glitchboy2000

Zagrebački umjetnik koji *glitch* slika na platnu.

Izvor: https://vizkultura.hr/greske-na-platnu-glitchboy/

Izvor: https://vizkultura.hr/greske-na-platnu-glitchboy/



Izvor: https://vizkultura.hr/greske-na-platnu-glitchboy/

Korey Bilmez

Izvor: https://deskgram.org/ubikobi.art?next_id=1702423101351986075_5556268511

Izvor: https://deskgram.org/ubikobi.art?next_id=1702423101351986075_5556268511

2.4 Osnovni pojmovi

2.4.1 Rezolucija

Rezolucija ili razlučivost može se definirati kao mogućnost kojom se razlučuju ili raspoznavaju najsitniji detalji, a opisuje se kvaliteta slike. U klasičnim tehnikama, odnosno na filmu, to su bile linije koje su se mogle razlučiti na dužinski milimetar filma, dok u digitalnim tehnikama ovisi o mediju. Medij za koji se zanima ovaj završni je digitalna fotografija, što znači da je rezolucija broj piksela po horizontali (x) i po vertikali (y), pr. 600x400. Što je veći broj piksela, a piksele definiramo upravo kao

najsitnije detalje digitalne fotografije ili slike, to je dig. fotografija kvalitetnija i manje rasterizirana. Što je kvaliteta veća, veća je i memorija koju zauzima dig. fotografija. Piksel definiraju boja i intenzitet, a pojam poznat kao dubina boje piksela (eng. *color depth*) označava dinamički raspon boje, odnosno broj ukupnih nijansi boje, ono bitno za ovaj završni – broj bita po svakom pikselu (BPP – *bits per pixel*), a tako opisana datoteka naziva se bitmapa (.bmp). Veličina rasterske slike je rezultat broja piksela, količine informacija o boji i kompresije.

2.4.2 Interpolacija

Interpolacija se tiče izračunavanja novih vrijednosti piksela na temelju već postojećih. Digitalna fotografija se može povećavati i smanjivati – mijenjati broj piksela na fotografiji tj. može se mapirati jedan piksel na čitavu grupu ili mapirati grupu piksela u jedan. Sustav pokušava izračunati vrijednost piksela uzimajući u obzir vrijednosti okolnih piksela, a to se radi pomoću različitih algoritama. Kada se slika povećava, stvaraju se rupe koje je potrebno popuniti, a kada se smanjuje, nastaje višak piksela te tako nastaje potreba za interpoliranjem.

2.4.3 Bit

Intenzitet piksela, kao što je već napisano, je broj bita po svakom pikselu. Bit je binaran, što znači da može imati samo dvije vrijednosti – 0 ili 1. Kod monokromatskih slika svaki piksel ima jedan bit, a taj piksel će biti 0 ili 1, a kvaliteta će ovisiti samo o rezoluciji. Kod sivih tonova svaki piksel će imati 8 bita (1 bajt) pa će moći poprimiti vrijednosti od 0 do 255 ($2^8 = 256$). U računalnoj grafici koristi se RGB sustav boja. RGB sustav boja sastavljen je od 3 osnovne boje – crvene, zelene i plave, miješanjem tih boja, mogu se dobiti sve ostale koje pripadaju vidljivom spektru. Kod slika u boji svaki će piksel biti spremljen kao 24 bit-a (3 bajta) – 8 bit-a za svaku boju. Kod takvih je slika moguće (2^{24}) 16 777 216 kombinacija boja.

2.4.4 Kompresija

Tokom pohrane datoteke, postupak smanjivanja memorije koju će ista zauzimati naziva se kompresija. Postoje dva osnovna tipa kompresije – bez gubitaka (eng. *losless*) i sa gubicima (eng. *lossy*). Vrsta kompresije utjecat će na osjetljivost datoteke prilikom *databendinga*. Kompresija sa gubicima prisutna je kod formata kao što je JPEG, a takva

kompresija izbacuje piksele, koje ljudsko oko nije u mogućnosti percipirati, pomoću različitih algoritama. Kompresija bez gubitaka sažima podatke datoteke bez gubitaka informacija, takav je proces reverzibilan i jednakovrijedan izvornom. Neki od formata jesu PNG i GIF. Tokom *databendinga*, koriste se nekompresirane datoteke kako bi iste mogle "bolje" podnositi korupciju kôda. Moguće je koristiti kompresirane datoteke, ali se mora znati da će one biti podložnije korupciji te će, u konačnici, češće postajati nečitljive.

2.4.5 Kanali boja

Slike u boji sastoje se od tri kanala – crvene, zelene i plave boje. Jedan kanal nosi podatke o količini crvene, drugi zelene i treći plave boje (crveni, zeleni i plavi kanal). Kod spremanja slikovne datoteke iz nekog programa za uređivanje slika (formati RAW i TIF) može se odrediti kako će se spremiti informacije o boji. Kod RAW formata nude se opcije *interleaved* i *non-interleaved*, dok se kod TIF formata nude opcije *interleaved* i *per channel*. *Interleaved* znači da će se informacije o boji spremati za svaki pojedini piksel RGB redoslijedom – podaci su zapisani redom za svaki piksel, dok *non-interleaved* odnosno *per channel* znači da će se podaci o boji za sve piksele spremati redom RGB, tj. prvo će se spremati podaci o crvenoj boji za sve piksele, podaci o zelenoj boji za sve piksele pa na kraju podaci o plavoj boji za sve piksele – podaci će biti grupirani po kanalima. Taj redoslijed utječe na rezultate *databendinga*. Preporučuje se korištenje *non-interleaved*/*per color* opcije, pošto se njome dobivaju izraženije boje korupcijom kôda.

2.5 Databending

Jedna od najvažnijih stvari kod *databendinga* jest da se nikada ne bi smjelo "uređivati" ili koristiti izvornu datoteku. Taj je koncept poznat i kod uređivanja fotografija i slika u programima kao što je Photoshop. Bitna razlika je ta da se u Photoshopu izvorna uređena fotografija može spremiti pod novim imenom korištenjem „*Save As*“ opcije te nema nikakvih reperkusija, dok se kod *databendinga* nikada ne bi smjelo uređivati izvornu datoteku jer se ona može uništiti pošto se takvo uređivanje direktno tiče korupcije kôda slikovne datoteke. Dakle, nikada ne uređivati izvornu datoteku te uvijek napraviti kopiju datoteke i koristiti kopiju tokom *databendinga*.

U slučaju kad se koriste tehnike u teksteditorima, hekseditorima ili audioeditorima datoteku je potrebno prirediti kao bitmapu ili sirovi (eng. *raw*) format. Što se tiče Blendera, moguće je koristit .jpg jer se učitava kao tekstura. No, preporuča se ekperimentiranje sa formatima, zbog neočekivanosti i nepredvidljivosti rezultata.

2.5.1 Wordpad

Wordpad je tekstualni uređivač koji dolazi sa *Windows-ima*, besplatan je i jednostavniji od *Microsoft Word-a*. *Databending* u *Wordpad-u* je jedan od, ako ne i, najjednostavniji *glitch* efekt koji se može napraviti. Sastoji se, jednostavno, od učitavanja priređene slikovne datoteke u programu, a *glitch* se dobiva kada *Wordpad* pokušava interpretirati slikovnu datoteku kao tekstualnu.

Efekt se dobiva između ostalog zato jer Wordpad tokom učitavanja datoteke mora znati gdje će prelomiti redak, a to se isto događa tokom učitavanja slikovne datoteke. Prijelomi redaka (eng. *line breaks*) su bajtovi u kôdu koji teksteditoru govore gdje će se prelomiti redak i krenuti u novi. Ti bajtovi predstavljaju tzv. *Line Feed*, a kôd koji ga reprezentira je 0x0A u heksadecimalnom sustavu, i *"carriage return"* (poluga za povlačenje valjka u novi red) kojeg reprezentira heksadecimalni kod 0x0D. Takvo nazivlje dolazi još od pisaćih mašina kada se papir provlačio kroz valjke, a red u kojem se počinjalo pisati bio je *"line feed"* te se trebala povući poluga/ručka da bi mašina otišla u novi red. U principu, ista se stvar događa u kôdu - 0x0A govori teksteditoru gdje završava redak, 0x0D govori da se prebaci na početak novog reda.

Slikovne datoteke ne sadrže *"line feed-ove"* ni *"carriage return-se"*, ali Wordpad ne prepoznaje da se radi o slikovnoj datoteci, već ju "doživljava" i interpretira kao tekstualnu datoteku pa traži 0x0D i 0x0A kako bi znao gdje prelomiti redove, ti bajtovi kôda u slikovnoj datoteci znače nešto posve drugo. Svaki put kad naiđe na ijedno od tih bajtova bez svog para, dodaje ili 0x0D ili 0x0A, ovisno na što je naišao. Pošto, kao što je već napisano, ti bajtovi u kôdu slike znače nešto sasvim drugo, kada Worpad misli da ispravlja datoteku, zapravo ju koruptira i tu se stvara Wordpad efekt.

Wordpad efekt je jako destrukivan efekt, što se tiče rezultata slike i promjene kôda, tako da u ovom slučaju nije pretjerano reći da će se više slikovnih datoteka uništiti, nego što će ih uspjeti *databend-ati*, važno je uvijek koristiti kopiju izvorne datoteke.

Databending sam po sebi zahtjeva dosta strpljenja pošto se tiče direktne korupcije kôda (slikovne) datoteke.

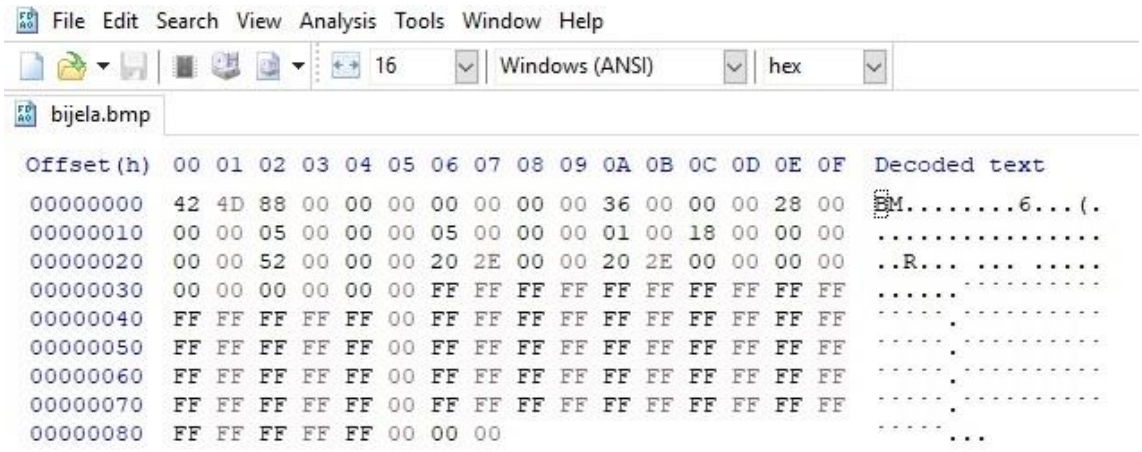
2.5.2 HEX

Heksadecimalni brojevni sustav je sustav s bazom broja 16 pa se i sastoji od 16 znamenki, a to su brojevi od 0 do 9 i slova engleske abecede od A do F, gdje A predstavlja brojku 10, a F 15 u dekadskom sustavu. Heksadecimalna notacija univerzalni je prikaz podataka u računalima. Pokazao se kao puno praktičniji sustav od binarnog jer je brojevna baza potencija broja 2 pa svaka znamenka u heksadecimalnom sustavu zamjenjuje četiri znamenke binarnog sustava (2^4=16). U dekadskom sustavu dvije znamenke predstavljaju četiri bita, a četiri bita, jedan bajt. Svaki bajt može imati vrijednosti od 00 do FF, odnosno u dekadskom sustavu od 0 do 255. Dakle, bilo kakvi računalni podaci zapravo su polje (eng. array) heks bajtova.

Hexeditori ili uređivači dopuštaju nam prikaz i uređivanje izvornog heksadecimalnog kôda digitalne fotografije/slike.

Heksadecimalni prikaz boja dobro je poznat dizajnerima i programerima i vjerojatno svima između. Heksadecimalni prikaz boje izgleda ovako: #RRGGBB. Svaki piksel ima šest znakova, tri seta po dvije znamenke, gdje prvi set predstavlja crvenu, drugi zelenu i treći plavu boju, a minimalne, odnosno maksimalne vrijednosti kreću se od 00 – FF (0-255). Dakle, kada bi se htio dobiti crveni piksel, heksadecimalna notacija bila bi #FF0000 – postavilo bi se crveni set na maksimalnu vrijednost, a ostale na minimalnu. Bijela boja bila bi #FFFFFF, a crna #000000 (RGB je aditivni sustav miješanja boja).

Svaka bitmapa sastoji se od dva dijela – *file headera* (zaglavlje datoteke) i *image data* (podataka slike). U zaglavlju se nalaze osnovni podaci o datoteci kao što su vrsta datoteke, njene dimenzije i slično, a u drugom dijelu pikseli. Ovo će se najlakše shvatiti ako se u Photoshopu (ili GIMP-u) izradi bitmapa (.bmp) datoteka veličine 5 x 5 piksela. Takva se datoteka otvori u nekom hekseditoru kao što je primjerice HxD.



Lijevi stupac predstavlja broj redova, sredina je heksadecimalni kôd, a desno je tekstualni ASCII kôd. Prvi red heksadecimalnog kôda je:

42 4D 88 00 00 00 00 00 00 00 36 00 00 00 28 00

Ovo su informacije o datoteci, 42 4D nalaze se na početku svake bitmape.

88 predstavlja veličinu datoteke, pretvorena u dekadski sustav datoteka zauzima 136 bajta. Prvi se red zove *BITMAPFILEHEADER*.

Drugi, treći i dio četvrtog reda kôda jesu:

00 00 05 00 00 00 05 00 00 00 01 00 18 00 00 00

00 00 52 00 00 00 20 2E 00 00 20 2E 00 00 00 00

00 00 00 00 00 00

05 00 je širina koja je namještena na 5 piksela, a nakon toga je visina koja je isto namještena na 5 piksela. 01 00 18 00 standardne *header* informacije koje bitmapa mora sadržavati. Ovaj dio datoteke sadrži informacije kao što su veličina, rezolucija, palete i slično. Taj dio datoteke zove se *BITMAPINFOHEADER*.

Prilikom uređivanja heksadecimalnog kôda tokom *databendinga* *BITMAPFILEHEADER* i *BITMAPINFOHEADER* se ne smiju uređivati, osim ako korisnik točno zna koji podatak mijenja, jer će u suprotnom uništiti datoteku i ona više neće biti čitljiva. Datoteke formata .bmp se čitaju s desna na lijevo, od dole prema gore, a boje se čitaju B-G-R redoslijedom, također obrnuto od uobičajenog.

Da se bitmapa pisala ručno u hekseditoru nakon *file header-a* i *info header-a,* prvi red koji bi se pisao zapravio bi se nalazio pri dnu slikovne datoteke kada bi se ona otvorila u nekom *image view-eru*.

Sam *databending* u hekseditoru zahtjeva otvaranje slikovne datoteke, prethodno spremljene kao bitmape te uređivanje kôda po vlastitom ukusu. Može se direktno mijenjati heksadecimalni kôd, a može se uređivati i tekstualni ASCII kôd, važno je znati da je jedan ASCII znak promjena od dvije znamenke u heksadecimalnom kôdu. Također se preporuča korištenje opcije Find&Replace ili ručne zamjene znakova umjesto jednostavnog brisanja kôda.

2.5.3 Sonifikacija

Sonifikacija je, u kontekstu *databendinga* slikovne datoteke, proces pretvaranja slike u zvuk u programu koji služi uređivanju auditivnih medija. Slikovna se datoteka učitava kao sirovi podaci koje će program interpretirati kao zvuk te se na takav zvuk aplicira logika zvučnih efekata koje nudi program. Osim što se sonifikacija koristi od strane vizualnih digitalnih umjetnika kao tehnika uređivanja slikovne datoteke, eksperimentalni glazbeni umjetnici učitavaju slikovne datoteke kako bi dobili zvuk i od toga rade pjesme.

U poglavlju koje se ticalo heksadecimalnog kôda i njegovog uređivanja, objašnjeno je zašto se ne smije uređivati zaglavlje kôda. Ta se logika proteže kroz sve tehnike i ni u jednoj nije dopušteno uređivati zaglavlje, osim ako se radi o *RAW* formatu koji nema zaglavlja, jer u protivnom datoteka neće biti čitljiva.

Nakon što se slikovna datoteka importira u program, jedna od najbitnijih postavaka na koje treba paziti je *enconding. Encoding* se odnosi na promjenu digitalnog auditivnog medija iz jednog formata u drugi, obično iz nekompresiranih u kompresirane. Opcije koje su nam bitne tokom *databendinga* jesu samo dvije – A-law i U-law. Kad se slikovni medij importira u program trebalo bi izabrati između jedne od tih dviju opcija, zato jer će jedino te dvije opcije učitati datoteku bez dodavanja ikakvih informacija u zaglavlje datoteke i jedine će datoteku učitati isključivo kao čiste podatke. Kod *channels* (kanala) možemo staviti – 1 (mono). S ostalim opcijama poželjno je igrati se pošto će na neki način utjecati na krajnji rezultat zajedno s efektima primjenjenim u programu.

2.5.4 Blender

Blender je *open source* program za 3D računalnu grafiku. Postupak koji je objašnjen u eksperimentalnom dijelu, zapravo je transkripcija videa[5], a njen je autor Christopher Lynch. U tom se postupku koristi format .obj kojeg je razvila tvrtka *Wavefront Technologies*. Taj je format također *open source* pa je tako adoptiran od strane drugih tvrki koje se bave razvojem 3D računalne grafike. OBJ format predstavlja 3D geometriju, odnosno koordinate zakrivljenosti linija i ploha poligona koji su napravljeni u Blenderu. *Databending* se u ovoj tehnici događa onda kada se taj format otvori u tekstualnom editoru te kada se koordinate poligona počnu zamjenjivati i mijenjati. Slikovna se datoteka zapravo učitava kao tekstura, a opcijama u Blenderu pretvara se u 3D presliku, a ono što se "uređuje" ili koruptira u tekstualnom editoru zapravo su njene UV koordinate. Blender je jako osjetljiv na taj format pa je u slučaju pogreške neće učitati. Brojeve je sigurno mijenjati iz manjih u veće, a linije se smiju dodavati, smije im se mijenjati redoslijed, ali se nikako ne bi smjele brisati. Kada se takva datoteka ponovno učita u Blender, također ponovno će se učitati ista tekstura slikovna datoteka, ali će Blender sada primjeniti nove koordinate i tako će se postići *glitch*.

[5] https://www.youtube.com/watch?v=_YbvXgls1Yw&t=23s

3. EKSPERIMENTALNI DIO

Cilj rada je izrada serije slika, reinterpretiranih digitalnih fotografija, koje će biti rezultat obrađenih i objašnjenih tehnika *databendinga*. Unošenjem i izazivanjem greške u kôdu slikovnog medija pružiti će se uvid u estetiku neočekivanog i umjetnost greške.

Metodologija *databendinga* bazira na pripremanju adekvatnog formata slikovnog medija, njegova korupcija u programu te ponovnog spremanja reinterpretiranog djela pod novim imenom. Ponekad se korupcija postiže samim otvaranjem datoteke u programu, dok se kod nekih drugih tehnika na nju primjenjuju različiti efekti, izmjenjuje se heksadecimalni kôd medija ili se *glitch* postiže remećenjem koordinata 3D objekta.

3.1 Teksteditori

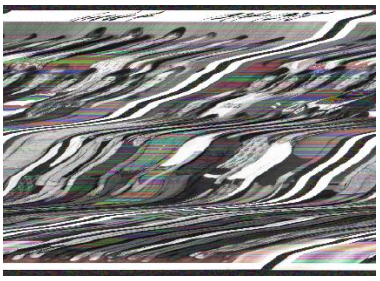
3.1.1 *Wordpad*

Wordpad je teksteditor koji dolazi sa gotovo svim verzijama *Windowsima*, kako bi oni koji nemaju *Microsoft Office* mogli otvarati i uređivati tekst datoteke. Složeniji je od *Microsoft Notepad-a*, ali jednostavniji od *Microsoft Word-a*. *Wordpad* se počinje „čudno“ ponašati kada pokušava interpretirati sliku kao tekst pa zahvljajući tome dobivamo Wordpad efekt. Taj je efekt otkrio Benjamin Berg, internetu poznatiji kao stAllio!, a spada u najjednostavniji *glitch* efekt. Sve što je potrebno uraditi je spremiti sliku kao .bmp u Paint-u, otvoriti je u Wordpad-u i ponovno spremiti. Wordpad će sam odraditi posao.

Izvor: Charlie's Angels (1976)

Izvor: autor

Izvor: flickr.com (Brecht Bug)



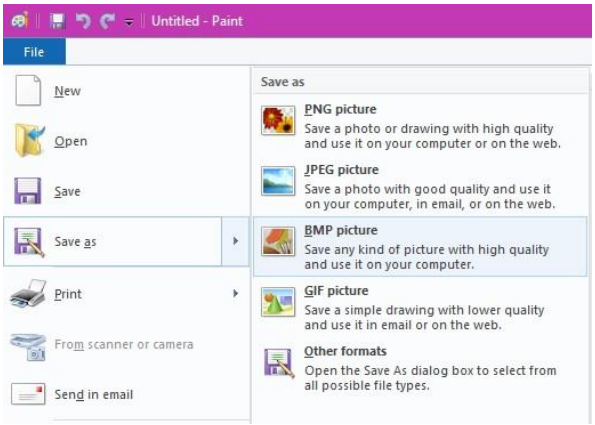
Izvor: autor

Izvor: pinterest.com (CNN)

Izvor: autor

Postupak:

1. Otvoriti sliku/fotografiju u Paint-u i spremiti ju kao bitmapu (.bmp), u padajućem izborniku izabrati 24-bitnu bitmapu

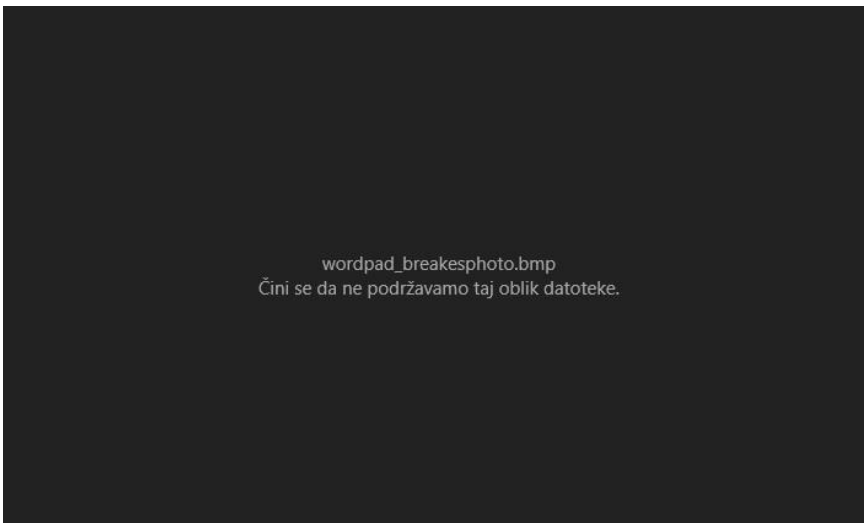


Izvor: autor

2. Tako spremljenu sliku otvoriti u Wordpad-u (desni klik na fotografiju, „**Open with**“ i pronaći Wordpad ili otvoriti Wordpad, ako ga se ne nalazi, stisnuti **Win + R** i napisati „**write**“ te kliklnuti „**Enter**“ pa bi se Wordpad trebao otvoriti. U izborniku „**File**“ kliknuti „**Open**“ te u donjem desnom padajućem izborniku odabrati „**All Documents (*.*)**“ i kliknuti na spremljenu bitmapu.

3. Kada Wordpad završi sa interpretacijom slike kao teksta, dovoljno je ponovno spremiti datoteku. Kliknuti na ikonicu diskete u gornjem lijevom kutu ili u padajućem izborniku „**File**“ kliknuti „**Save**“
4. Otvoriti tako spremljenu datoteku ponovno u Paint-u i spremiti ju kao .jpg

Slike se mogu konvertirati u bilo kojem drugom softveru za uređivanje slika, ali pošto se ovdje koristi Wordpad, može se koristiti i Paint, pošto i taj program dolazi sa gotovim svim Windowsima. Wordpad-ova najveća prednost, ali i mana je njegova nepredvidivost – može se zablokirati, može uništiti datoteku pa ju se tako može i izgubiti, a može i odraditi sve bez „greške“ pa se kao rezultat dobije *instant glitch*.

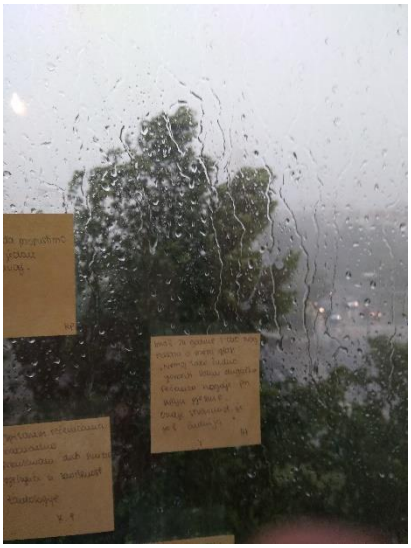


Izvor: autor

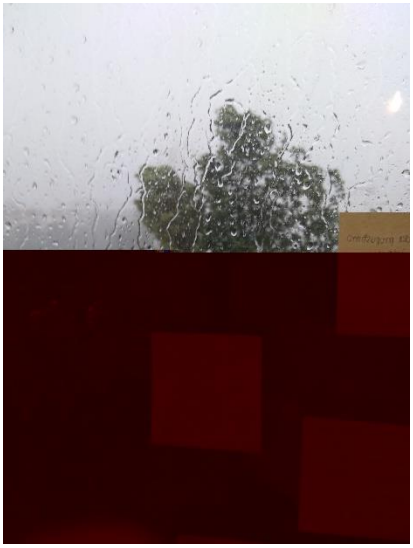
3.1.2 Notepad++

Notepad++ možemo koristiti na različite načine kod *databendinga,* jedan od njih je da ga koristimo kao teksteditor. Datoteka spremljena kao bitmapa otvara se u programu, za razliku od Wordpad-a, Notepad++ neće koruptirati datoteku samim njezinim otvaranjem u programu, već korisnik sam mora intervenirati i mijenjati kôd. Nikako se ne preporuča brisanje kôda, već samo zamjena nekoliko znakova, ali se svakako preporuča eksperimentiranje granicama korupcije kôda.

Postupak je sličan onom od Wordpad-a. U softveru za uređivanje fotografija, a može i u Paint-u datoteku spremamo kao bitmapu te ju otvaramo u Notepad++-u, tamo zamjenjujemo znakove ASCII kôda, spremamo datoteku i u jednom od softvera za uređivanje slika bitmapu spremamo kao .jpg.



Izvor: autor



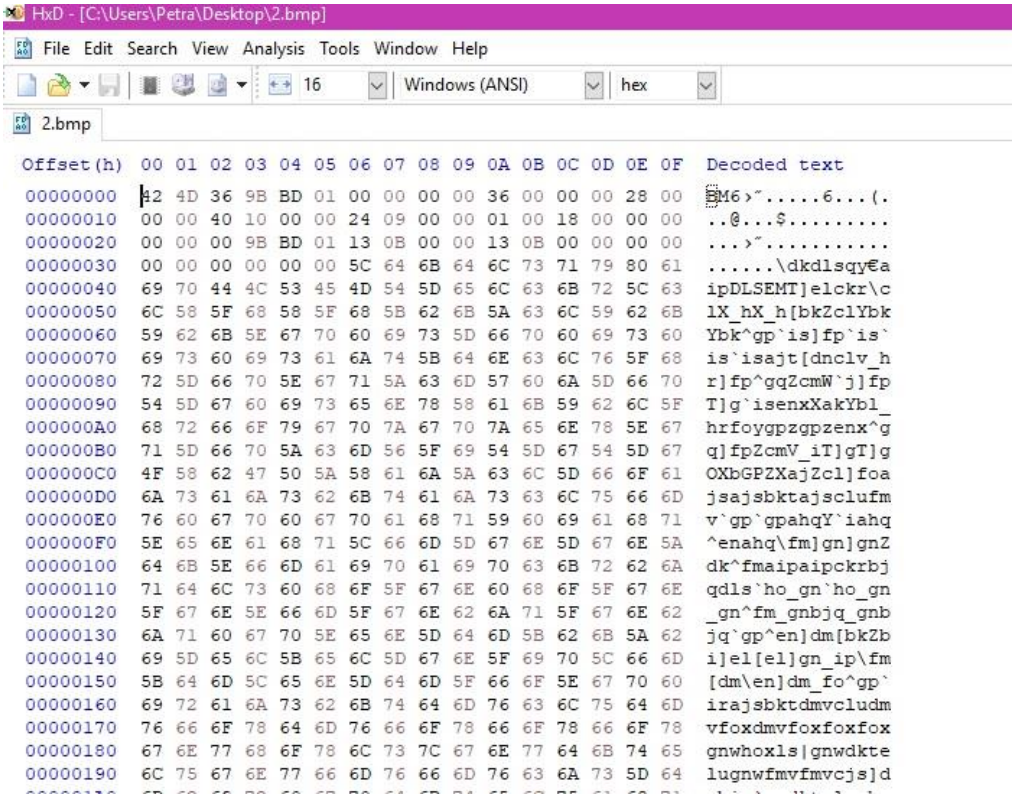
Izvor: autor

3.2 Hexeditori/Notepad++

3.2.1 Hexeditor

Hexeditori ne dolaze zajedno sa Windowsima (moguća je prenamijena Notepad++-a), dakle, moraju se skinuti sa interneta, ali zato postoje besplatne verzije i one koje se plaćaju. Funkcije koje pružaju nisu jednake, ali principi na kojima funkcioniraju jesu. Tako svaki hexeditor ima barem dva načina prikaza datoteke: heksadecimalni i tekstualni odnosno ASCII kôd.
Databending u hexeditoru je direktna korupcija izvornog kôda te omogućuje veću fleksibilnost u radu jer nije destruktivna kao što su Wordpad i Notepad++ koristeći se tekstualnim (ASCII) znakovima. U hexeditoru dvije brojke predstavljaju jedan ASCII znak.

Potrebno je izbjegavati zaglavlje kôda, pošto se tamo nalaze informacije o samoj datoteci i njenom prikazu. Formati kao što su JPEG ili PNG, također se mogu izmjenjivati, ali su zbog svoje kompresije podataka "ranjiviji" na direktno uređivanje njihovog kôda. Ovisno o formatu/kompresiji datoteke u hexeditoru dobit će se različiti rezultati iako su elementi od kojih je sastavljen njihov kôd slični.



Slikovna datoteka u hexeditoru – HxD

Izvor: autor

Postupak

1. Priprema slikovne datoteke kao bitmape
2. Otvaranje bitmape u hexeditor programu kao što je HxD
3. Izmjena kôda – mogu se koristiti opcije kao što su *copy/paste*, *find/replace*, može se izmijenjivati i ASCII kôd
4. Spremanje datoteke

Nakon svake uspješne izmjene kôda, datoteka bi se trebala spremiti pod novim imenom (*Save As*) jer nije pitanje hoće li datoteka postati nečitljiva, već kada će to biti.

Izvor: autor

Izvor: autor

Izvor: autor

Rezultati dobiveni tokom jedne manipulacije

3.2.2 Notepad++

Notepad++ se može koristiti i kao hexeditor, ali je potrebno instalirati dodatak. U tom slučaju, nakon otvaranja programa, odlazi se na izbornik *"plugins"* (dodaci), u padajućem izborniku izabrati *plugin manager* te zatim *Show Plugin Manager*, potražiti i označiti kućicu pored koje piše *HEX-Editor* te kliknuti *Install*. Tada se program treba restartirati. Nakon ponovnog otvaranja u padajućem izborniku *Plugins* trebala bi se nalaziti nova kategorija *HEX-Editor*. Slikovna se datoteka otvori u programu, odlazi se mišem na novu kategoriju *HEX-Editor* te nakon novog padajućeg izbornika klikne se na *"View in HEX*" ili se jednostavno, nakon otvaranja datoteke u Notepad++-u stisne **Ctrl+Alt+Shift+H**. Nakon toga postupak je isti kao i kod hex-programa.

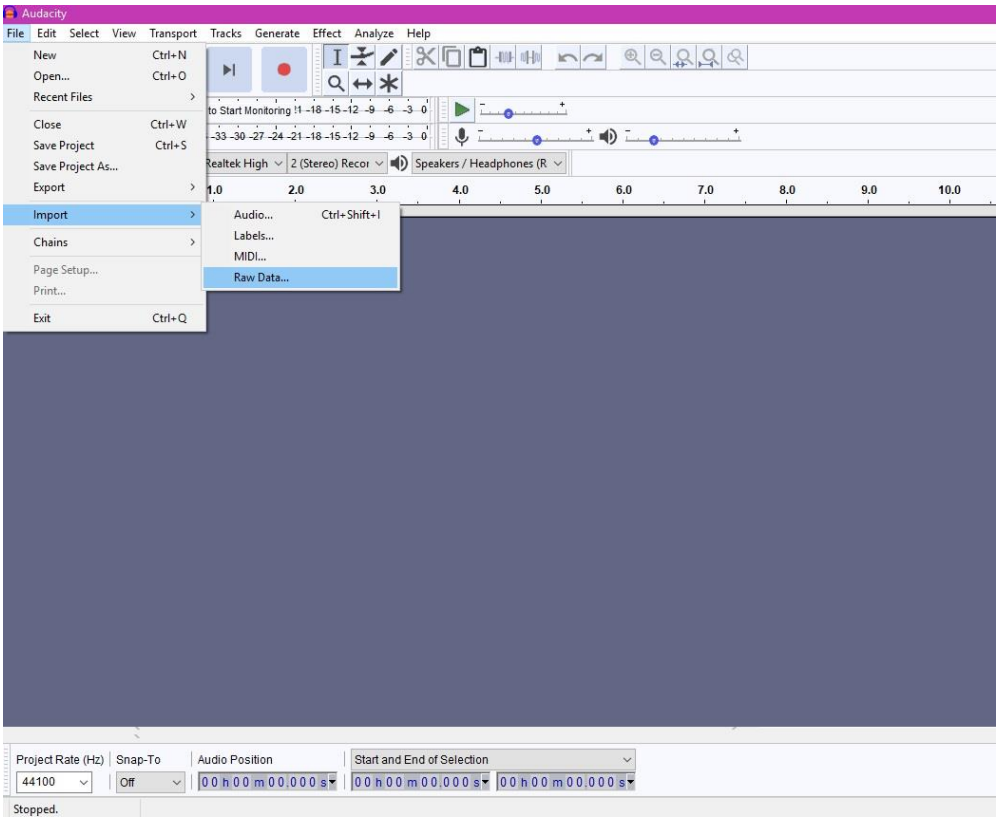
3.3 Audacity (Sonifikacija)

Audacity je program za obradu i snimanje zvuka, otvorenog kôda/izvora, besplatan i dostupan na svim operativnim sustavima. Sonifikacija je proces u kojem se slikovna datoteka u programu interpretira i obrađuje kao auditivni medij. Specifična estetika koja nastaje takvom obradom slikovne datoteke najčešći je oblik *glitcha*. Slikovna datoteka se može obrađivati primjenjivanjem audio efekata na zvučnu traku, stapanjem jedne datoteke s drugom ili *codecima*. Kao i kod ostalih tehnika, najbolje je

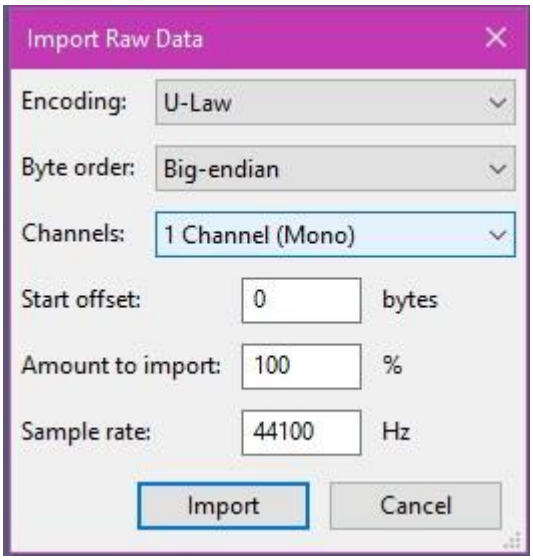
koristiti nekompresirane datoteke kao što su BMP ili TIFF. Kompresirane se datoteke također mogu obrađivati, ali su osjetljivije i lako postaju nečitljive. Slikovni format koji je najlakše i najsigurnije obrađivati je *RAW* slikovni format jer on ne sadržava zaglavlje (*header*) na koji je inače potrebno paziti.

Postupak:

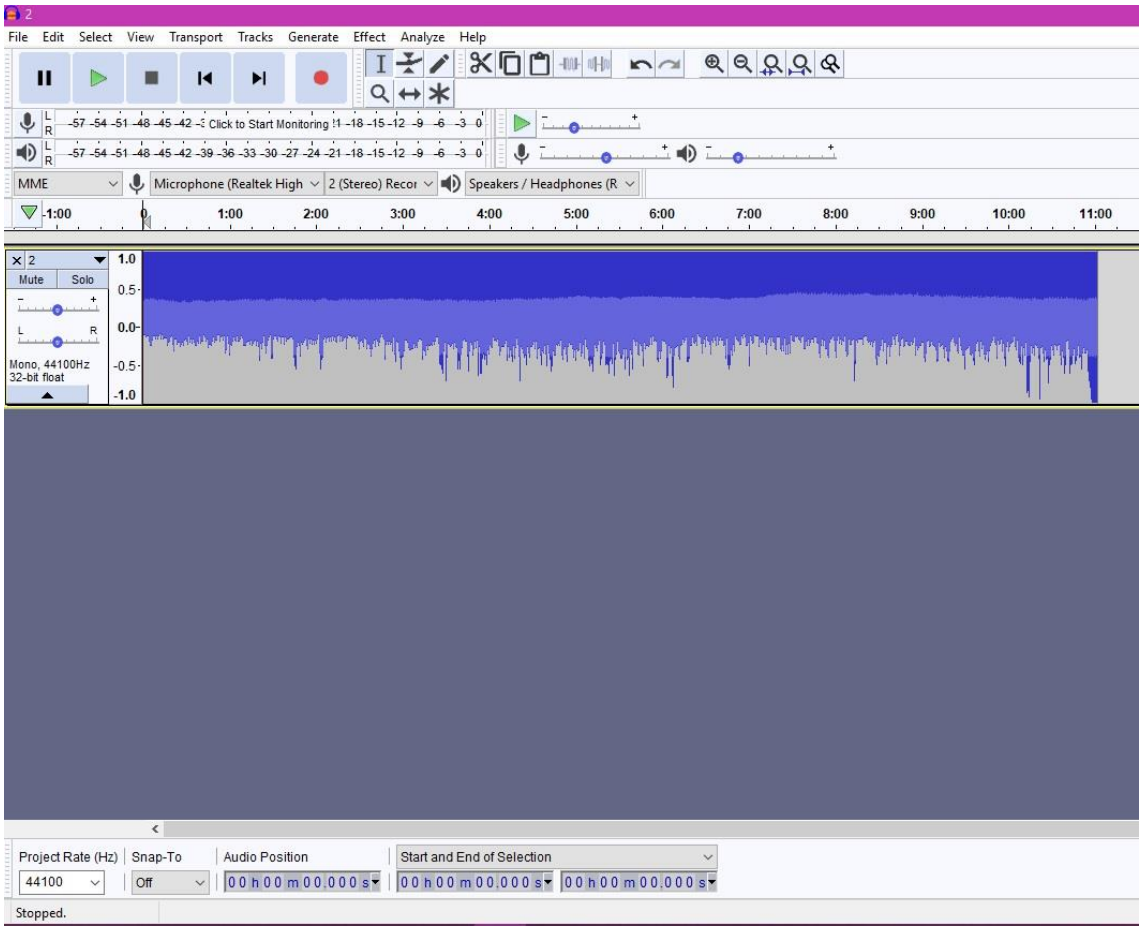
1. File → Import → RAW Data
   Encoding: A-law ili U-law
   Channels: 1 (mono)
   S ostalim opcijama se može eksperimentirati i pratiti kakav utjecaj imaju na krajnji rezultat.
2. Primjena različitih efekata na dio datoteke ili na cijelu; izostaviti prvih par sekundi ako se radi o formatima koji nisu RAW (može se koristiti bilo koji efekt osim oni koji utječu na trajanje datoteke – Paulstrecth ili Timeshift)
3. File → Export
   Format koji se odabire je 'other uncompressed files' iz padajućeg izbornika ipod imena datoteke. Bitno je (p)ostaviti encoding onakvim kakve su postavljene na početku, a Header: RAW (header-less). U imenu datoteke upotrijebiti onu ekstenziju, format, kao i kod ulazne datoteke



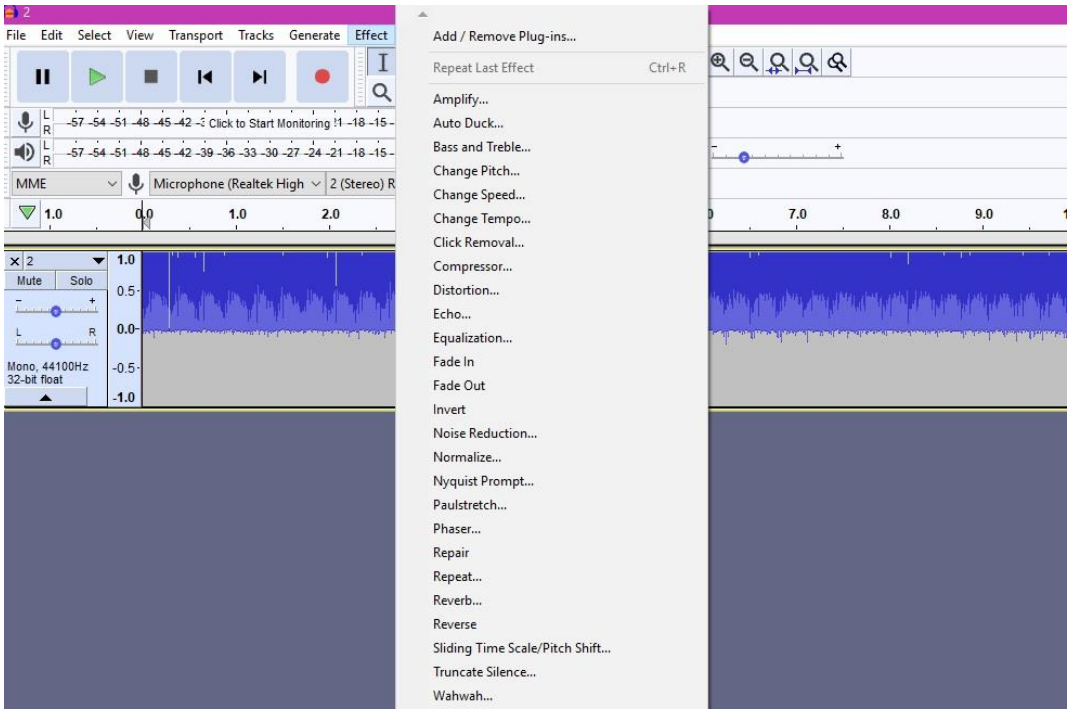
Izvor: autor



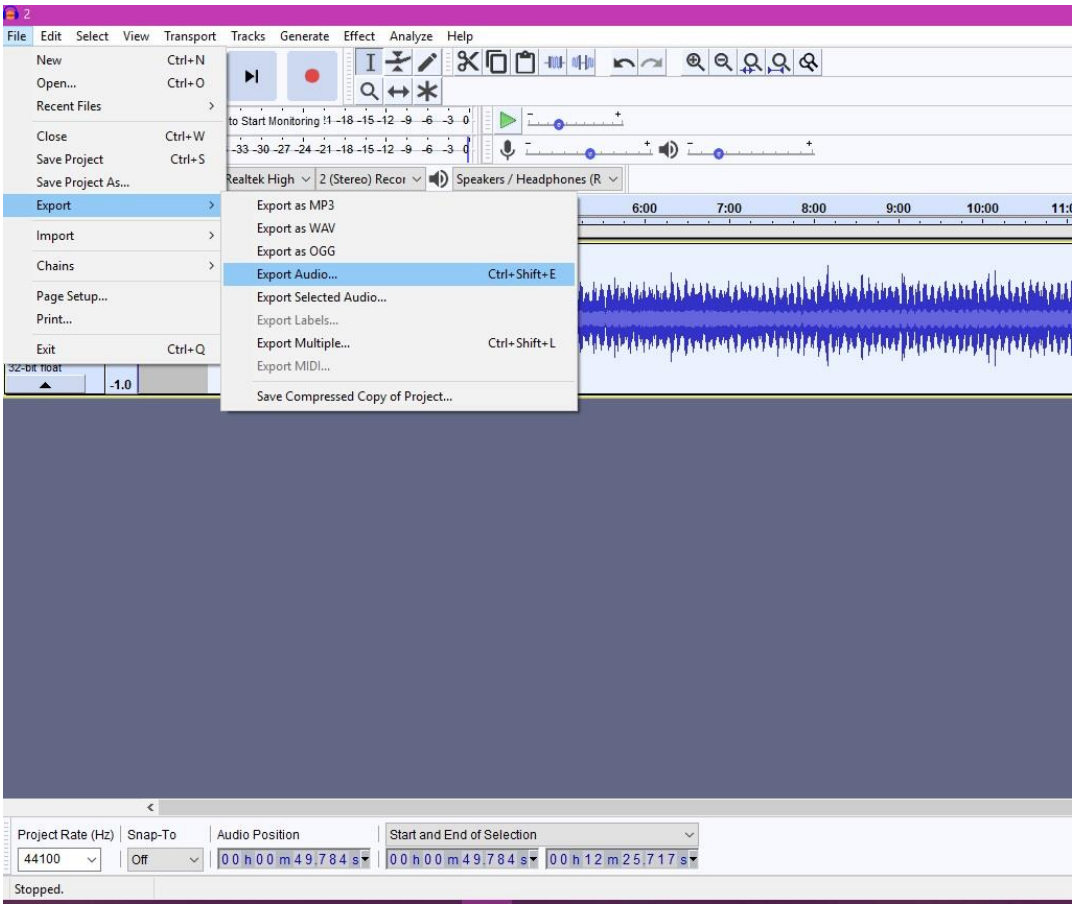
Izvor:autor



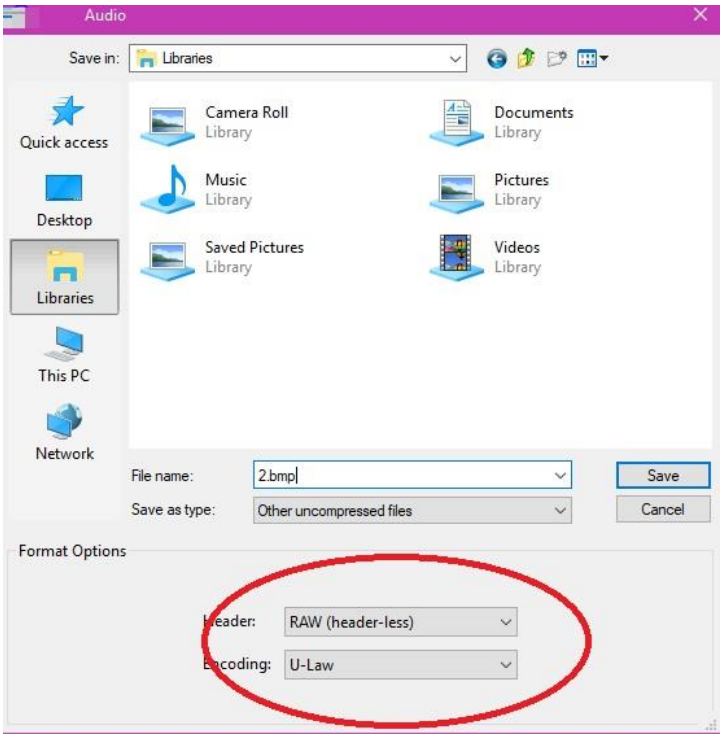
Izvor: autor



Izvor: autor

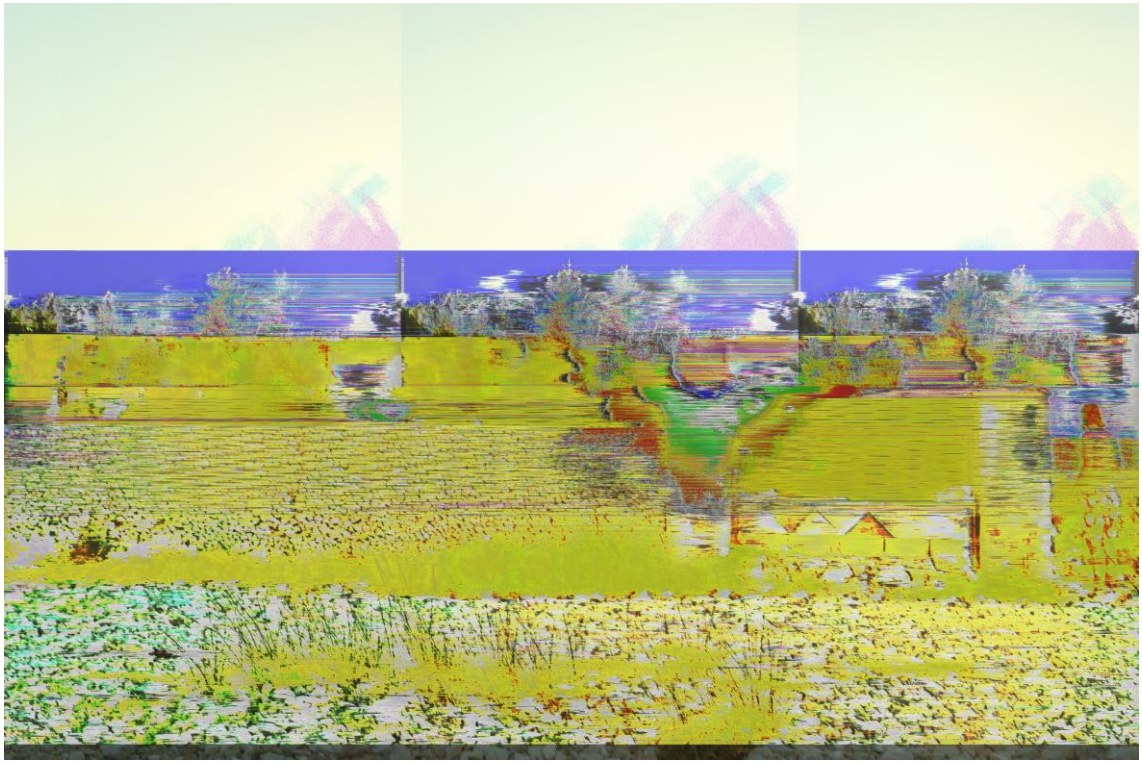


Izvor: autor



Izvor: autor

Izvor: autor



Izvor: autor

3.4 Blender OBJ Glitch

Blender je besplatan softver otvorenog kôda za 3D računalnu grafiku te kao Audacity, dostupan na svim operativnim sustavima. OBJ format u sebi sadrži koordinate i definicije geometrijskih objekata u tekstualnom obliku i jako je osjetljiv na formu, odnosno, Blender neće učitati datoteku ako se pogriješi tokom njenog uređivanja.

Postupak:

1. Selektirati i obrisati kocku (lijevi klik, *delete*)
2. Gornji *drop down* izbornik u alatnoj traci potrebno je prebaciti u *Cycles render*
3. *Numpad* 7 za *Top Perspective*
4. Prebacivanje u *Object mode* (donji lijevi padajući izbornik)
5. Dodamo plane: **shift + a** → *Mesh* → *Plane*
6. Selektiramo *Plane* i sa kraticom s, povećavamo plohu do zadnjeg reda *grida*
7. Vraćamo se iz *object moda* u *Edit mode* (tipka TAB)
8. **W** – *specials* → *subdivide* dok cijeli *plane* ne postane jednobojan, tj. crte postanu jako guste
9. **n**-kratica: *transform* → *Scale*; unosimo dimenzije *source* slike (pr. 1000 x 2000 px upisujemo 10 x 20)
10. u desnom izborniku *Material* (srebrna kugla) – *use nodes* – *new* – *source* = *emission*, *color* = *image texture*, *open* = odabiremo izvornu datoteku, *vector* = *texture coordinates* – *Generated*
11. prebacimo *method* na *rendered*, nalazi se pokraj *Object/Edit mode*, dolje lijevo – vizualna reprezentacija datoteke na plohi
12. *modifier* (francuski ključ) – *add modifier* → *Displace*, *texture* = samo stisnuti +, *texture coordinate* – UV
    Datoteka bi sada trebala izgledati kaotično. U slučaju, ovisno o verziji blendera, da je ostala ista, u izborniku *Texture* iz padajućeg izbornika potrebno je odabrati *Displace texture* (umjesto *Image texture*),a pod *Image* = *open* i otvoriti izvornu datoteku
13. *Modifier* izbornik = *st(renght)* 0.01
14. *Edit mode*

15. Kratica **u** (*unwrap*)
16. *File → export → Wavefront (OBJ)*
17. OBJ datoteka otvara se u teksteditoru, a sadržava popis sa koordinatama.

    V= *vertices*

    Vt= *texture coordinates*

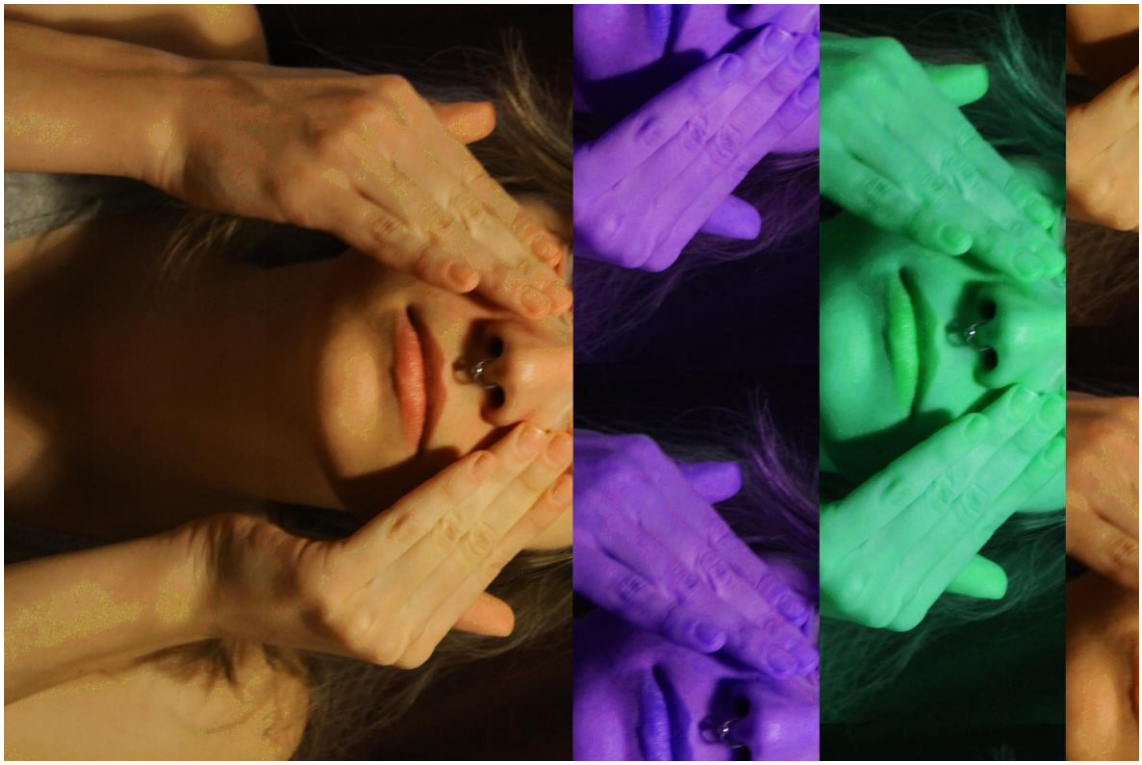
    F=*polygonal face element*

    Koordinate u datoteci se mogu seliti i modificirati, mogu se dodavati iz jedne skupine u drugu, kopirati i lijepiti, modificirati vrijednosti koordinata (manje u veće, obrnuto će možda uspjeti, možda ne). Linije se nikako ne smiju brisati, tako će se uništiti datoteka.

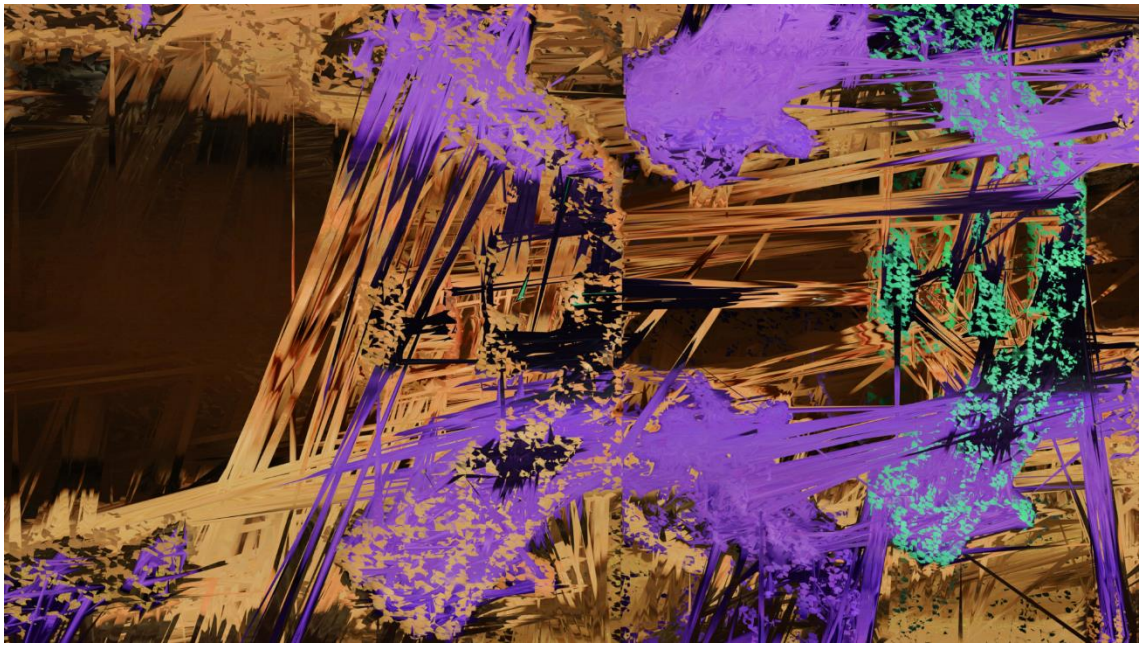
    Spremi se datoteka pod novim imenom.
18. *File → import → Wavefront (OBJ)* – novospremljena datoteka
19. U gornjem desnom kutu nalazi se izbornik sa svim elementima u Blenderu (camera, lamp, plane,...), odabire se novi *plane* koji se importiraju (*plane 1*)
20. *Material – use nodes – source = emission, color = image texture, open =* odabire se izvorna datoteka, *vector = texture coordinates – generated*
21. Renderiranje – numpad 7, odabire se *plane* – izbornik gore desno – ili lijevi klik na plohu, **Ctrl + Alt + 0** kako bi se centrirala kamera, vjerojatno je potrebno udaljiti se od objekta

    numpad 7 – *top prospective*, skrola se dovoljno daleko, držeći srednjim klik, pokretima miša namjestimo radnu površinu tako da s boka gledamo kameru i objekt, sa 0 mijenjamo pogled između kamere i korisničke perspektive
22. *Render* – ikona kamera – *resolution* – željena rezolucija, *slider* ispod na 100%, PNG, RGBA, *compression* 100%

    Stisne se F12
23. Dolje lijevo, *image → save image*



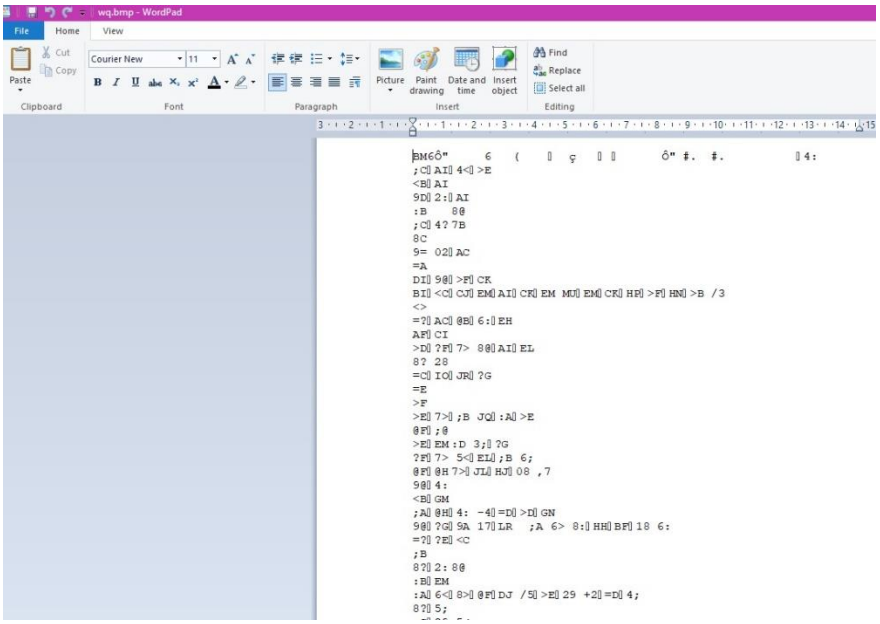
Izvor: autor



Izvor: autor

4. RASPRAVA I ZAKLJUČAK

Databending sam po sebi nije težak, ni onda kad se točno ne razumije što se radi i zašto se neki efekt/*bend*/*glitch* negdje dogodio. Iako bi se u *databending* moglo krenuti, a i većina kreće, samo radi dobivanja *glitcha*/rezultata, pozadina računalnog svijeta koja se potom otkriva gotovo je fascinantna, pogotovo za osobe koje ne mogu reći da se smatraju programerima . Slike uključene u ovo poglavlje dobivene su u Audacity-ju, Blenderu i HxD-u. Sa Wodpad-om 2018 naišlo se na nekoliko problema, a to su da isprva nije htio učitati datoteku, kasnije ju je učitao, ali ju nije htio spremiti pa kad bi ju spremio datoteka bi bila uništena. Rezultati, tj. slike u poglavlju koje se tiče Wordpad-a, dobivene su u nekim od prošlih verzija tog programa.



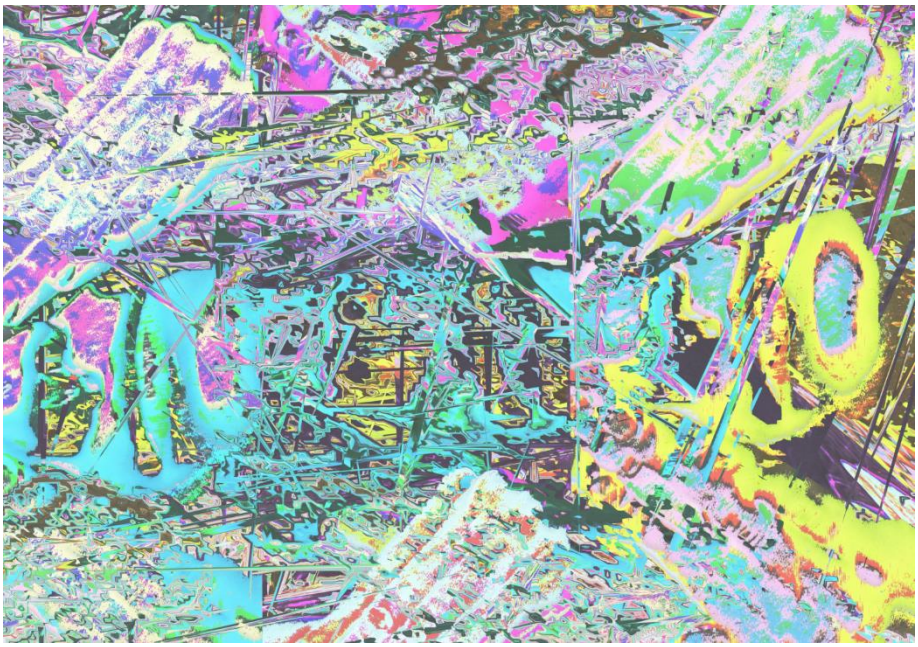
Izvor: autor

Također, ASCII kôd svake datoteke koja bi se uspjela otvoriti izgledao je slično prikazanom, što je dalo naslutiti da datoteka nakon spremanja više neće biti čitljiva, pošto su u svakom retku tek nekoliko znakova prije nego započne novi – kôd se koruptirao toliko samim otvaranjem datoteke da *bend* više nije bio moguć. Svi su ostali programi i pustupci funkcionirali.

Tokom *databendinga* nije pitanje hoće li datoteka postati ne čitljiva, već kada će to biti pa je za ovakvu tehniku najbitnije strpljenje, obavezno je kopiranje izvorne datoteke, a

preporučuje se i spremanje (*Save As*) datoteke nakon aplikacije nekog efekta (Audacity) ili mijenjanja kôda (HEX).

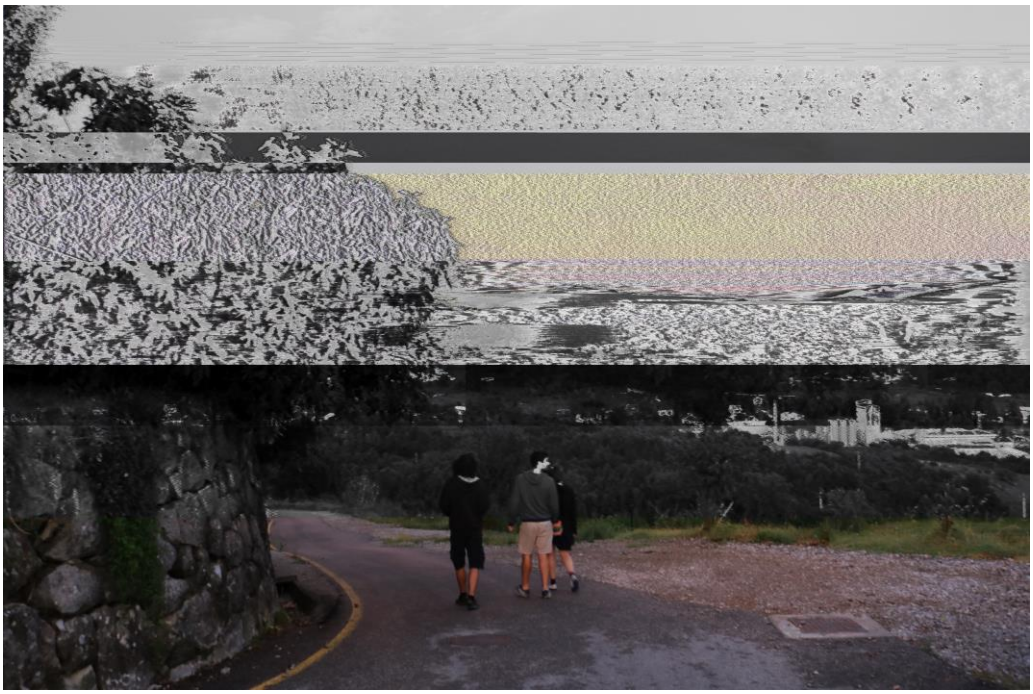
U *glitch* kulturi, *glitcheve* dobivene u programima kao što je Audacity, tj.namjernom izmjenom kôda, nazivaju "pripitomljenim *glitchevima*", a "divlji *glitchevi*" jesu oni stvarno slučajni, a takvi postaju sve zanimljiviji.



Izvor: autor

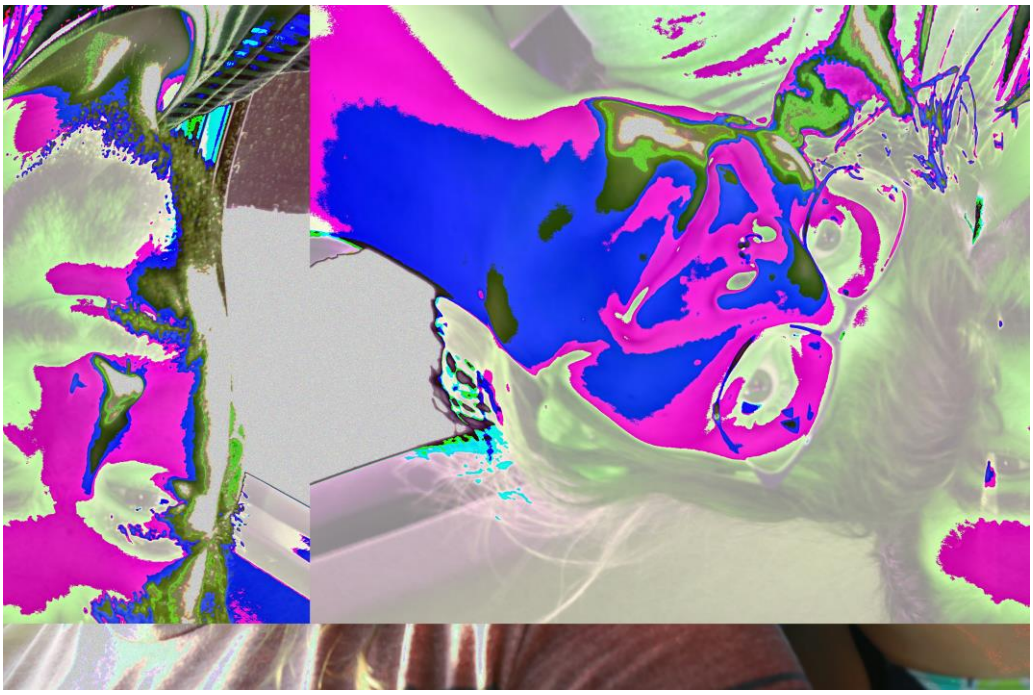
Izvor: autor

Izvor: autor



Izvor: autor

Izvor: autor



Izvor: autor

5. LITERATURA

[1.]Manovič L. (2015.), Jezik novih medija, Multimedia Clio, Beograd

[2.] https://formatc.hr/glitch_databend#I

[3.] https://hr.wikipedia.org/wiki/Digitalna_umjetnost

[4.] http://datamoshing.com/

[5.] https://en.wikipedia.org/wiki/Wavefront_.obj_file

[6.] https://hr.wikipedia.org/wiki/Razlu%C4%8Divost

[7.] http://datamoshing.com/2016/06/26/how-to-glitch-images-with-wordpad/

[8.] http://magazine.art21.org/2011/09/13/how-to-create-a-bitmap-image-file-by-hand-without-stencils/