

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA OSIJEK**

Obrada slike i računalni vid

LITHOPHANE GENERATOR

Projektni zadatak

Nositelj: Izv. prof. dr. sc. Irena Galić

Student: Marin Stević, DRD

SADRŽAJ:

1.	UVOD	3
2.	PREGLED PODRUČJA I PROBLEMATIKE	4
3.	OPIS ZADATKA S DOBIVENIM REZULTATIMA	5
3.1.	Programsko rješenje	5
3.2.	Analiza rezultata	7
4.	ZAKLJUČAK	9
5.	LITERATURA	10
6.	UPUTE ZA KORIŠTENJE	11

1. UVOD

Lithophane [1] predstavlja vrstu umjetničkog djela graviranog ili oblikovanog u vrlo tankom prozirnrom porculanu koje je vidljivo samo pod pozadinskim osvjetljenjem. Riječ "*lithophane*" potječe od grčkog "*litho*", što je od "*lithos*" što znači kamen ili stijena, a "*phainein*" što znači "izazvati pojavu" ili "izazvati pojavljivanje naglo". Iz ovoga proizlazi značenje "svjetlosti u kamenu" ili "pojaviti se u kamenu", jer se trodimenzionalna slika pojavljuje iznenada kada je osvijetljena stražnjim izvorom svjetla.



Slika 1.1. Primjer porculanskog *lithophane-a*

Primjer *lithophane-a* na slici 1.1. je napravljen ručno ali razvojem i popularizacijom 3D printera ljudi su ubrzo shvatili da se izrada *lithophane-a* može automatizirati. 3D printeri kakve danas poznajemo su uređaji upravljani računalom koji na određenim XY koordinatama ostavljaju određenu količinu plastike i time mogu stvarati razne 3D oblike. Pošto *lithophane* predstavlja dosta jednostavan trodimenzionalni objekt možemo zaključiti da ga možemo napraviti pomoću 3D printera.

2. PREGLED PODRUČJA I PROBLEMATIKE

Projekt se sastoji od dva problema:

- generiranja *lithophane-a* iz slike
- printanja *lithophane-a* pomoću 3D printera

Programi za generiranje *lithophane-a* iz slika postoje od pojave prvih komercijalnih 3D printera. Ti programi su trenutno pre komplicirani, imaju previše nepotrebnih dodataka i zahtijevaju podešavanje previše postavki koje početnik ne može razumjeti. Stoga je jedan od ciljeva ovog projekta napraviti program jednostavnim.

Printanje *lithophane-a* više nije problem jer standardizacijom i masovnom komercijalizacijom 3D printera i najlošiji 3D printer ima mogućnost printanja *lithophane-a* visoke kvalitete.

3. OPIS ZADATKA S DOBIVENIM REZULTATIMA

Cilj seminarskog rada je implementirati jednostavno rješenje za generiranje *lithophane* 3D modela iz predane slike pomoću Python programskog jezika uz korištenje OpenCV [2] i Numpy [3] biblioteka. Za spremanje 3D modela korišten je STL [4] datotečni oblik koji ima jednostavnu sintaksu i nije potrebno koristiti nikakve dodatne alate za njegovo stvaranje i obradu.

3.1. Programsko rješenje

Programsko rješenje implementirano je u programskom jeziku Python. Proces generiranja *lithophane-a* se može podijeliti u sljedeće korake:

- učitavanje slike
- unos parametara
- kvantizacija debljine modela
- generiranje 3D modela

Za učitavanje slike korištena je OpenCV biblioteka. Slike se učitavaju iz direktorija *images* u crno-bijelom formatu gdje je svaki piksel predstavljen razinom svjetlosti. Najčešće se koristi 8-bitna rezolucija za svaki piksel što daje 256 različitih razina svjetlosti. Vrijednosti svakog piksela su potom spremljene u Numpy *array* što olakšava njihovo uređivanje. Prva stvar što se treba uraditi na slici je da se vrijednost za svaki piksel invertira, odnosno da vrijednost 0 predstavlja maksimalnu razinu svjetla (bijeli piksel), a 255 minimalnu (crni piksel). Ovaj postupak olakšava kasnije kvantiziranje debljine jer ovako su mjesta najmanje debljine, maksimalnog propusta svjetla, predstavljena s 0, a mjesta najveće debljine, minimalnog propusta svjetla, su predstavljena s 255.

Poslije učitavanja slike trebaju se unijeti glavni parametri za izradu 3D modela koji su: minimalna i maksimalna debljina. Nakon toga trebamo kvantizirati debljinu *lithophane-a* na onoliko vrijednosti koliko ima razina svjetlosti predana slika. Ako za primjer uzmemo sliku koja ima 256 razina svjetlosti, minimalnu debljinu postavimo na 0.6mm i maksimalnu na 3mm dobiti ćemo da svaka sljedeća razina svjetlosti dodaje približno 0.00941mm na osnovnu debljinu od 0.6mm. Na taj način promjene debljine dobija se privid različitih vrijednosti razina svjetlosti.

Kada smo izračunali debljinu za svaki piksel ulazne slike, u posebnu datoteku s nastavkom *.stl* zapisujemo sve točke na sljedeći način:

- **v x y z**

Gdje *v* predstavlja posebnu oznaku u *.stl* sintaksi koja označava da se radi o točki u 3D prostoru i poslije toga predajemo njene koordinate *x*, *y*, *z*. Kada smo unijeli sve točke trebamo ih spojiti sa stranicama na sljedeći način:

- **f** *t*₁ *t*₂ *t*₃ ... *t*_{*n*}

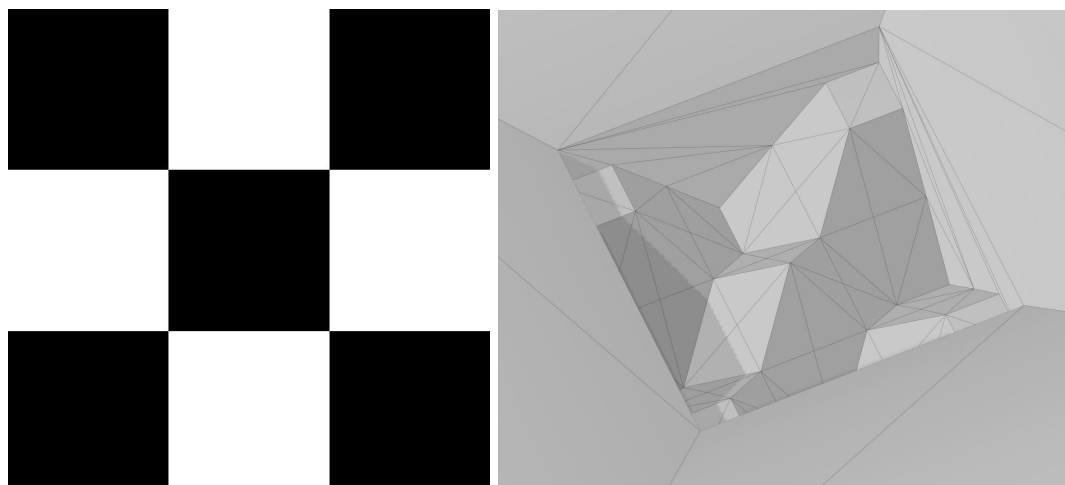
Pomoću *f* navodimo da se ovdje radi o *facet-u* odnosno stranici 3D modela i još predajemo točke kojima je ta stranica ograničena. Na slici 3.1. nalazi se primjer navođenja nekoliko točaka i stranica.

```
v 10.0 10.0 6.5
v 0 0 4.0
v 5.0 0 4.0
v 0 5.0 4.0
v 5.0 5.0 4.0
v 0 0 0.6
v 5.0 0 0.6
v 0 5.0 0.6
v 5.0 5.0 0.6
f 50 52 53 51
f 56 52 50 54
f 57 53 52 56
f 55 51 53 57
f 54 50 51 55
f 60 56 54 58
f 61 57 56 60
f 59 55 57 61
f 58 54 55 59
```

Slika 3.1. primjer zapisa iz *.stl* datoteke

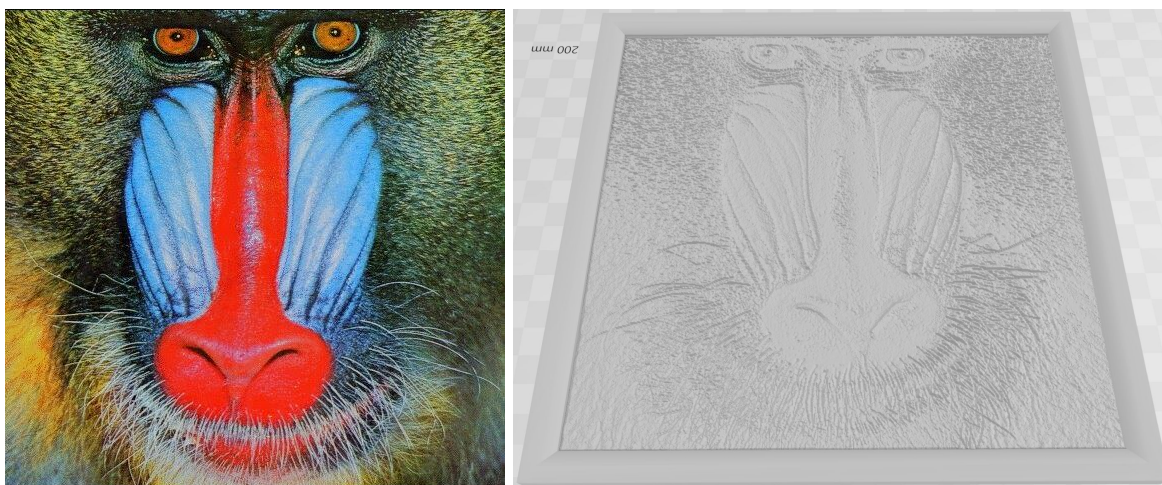
3.2. Analiza rezultata

Na slici 3.2. ispod je prikazan primjer testne slike i rezultirajućeg 3D modela. Ova slika je dobar primjer zato što pokazuje generiranje 3D modela iz slike koja sadrži krajnje vrijednosti odnosno samo crne ili bijele piksele.



Slika 3.2. primjer ulazne slike i rezultirajućeg 3D modela

Iz primjera sa slike 3.2. može se vidjeti i način zapisivanja točaka gdje su rubne vrijednosti jednake unutarnjima i gdje se vidi računanje točaka između glavnih vrijednosti piksela zbog povećanja preciznosti koja pomaže pri 3D printanju.



Slika 3.3. primjer ulazne slike i rezultirajućeg 3D modela



Slika 3.4. primjer isprintanog *lithophane-a*

Na slici 3.3. se vidi primjer ulazne slike i rezultirajućeg 3D modela koji je složeniji nego primjer sa slike 3.2. Također ovaj model se može vidjeti isprintan na slici 3.4. gdje se vidi model prije i poslije dodavanja pozadinskog osvjetljenja.

4. ZAKLJUČAK

Izvršavanjem zadatka ovoga projekta dobiven je program koji može iz predane slike izgenerirati 3D model koji realiziran pomoću 3D printera daje *lithophane*.

Iako ovaj program nije na razini postojećih rješenja po brzini ili izgledu sučelja. Njegove prednosti su jednostavno sučelje, korištenje Python programskog jezika i direktno generiranje 3D modela tako da se računa svaka točka i strana zasebno što daje mogućnost za proširivanjem osnovne funkcionalnosti programa.

Proširivost se može ostvariti tako da se dodaju funkcije za promjenu same slike ili načina generiranja 3D modela. Primjer toga može biti kvantiziranje ulazne slike čime se ne gubi kvaliteta slike ali ako gledamo crno-bijelu sliku u 3D prostoru tako da količina svjetla prikazuje visinu smanjujemo broj različitih debljina materijala i samim time smanjujemo broj naglih pokreta (*jitter*) 3D printera jer su to vrijednosti unutar 1mm.

5. LITERATURA

- [1] <https://en.wikipedia.org/wiki/Lithophane>
- [2] <https://opencv.org/>
- [3] <https://numpy.org/>
- [4] [https://en.wikipedia.org/wiki/STL_\(file_format\)](https://en.wikipedia.org/wiki/STL_(file_format))

6. UPUTE ZA KORIŠTENJE

Izvorni kod programskog rješenja, dostupan je na sljedećoj poveznici:

<https://github.com/MarinStevic/OSiRV-Projekt>

Za pokretanje programa korisnik mora imati instaliran Python3 i potrebne biblioteke (OpenCV i Numpy). Sve slike koje korisnik hoće koristiti trebaju se staviti u *images* direktorij.

Pokretanje programa se vrši sljedećom naredbom:

- *python generate.py*

```
Please enter image name: airplane.bmp
Minimal thickness (mm, default=0.6): 0.6
Maximal thickness (mm, default=3.0): 3.0
Choose sizing option:
  1. Pixel size
  2. Object size
Choose: 2
Choose:
  1. Height
  2. Width
Choose: 2
Enter height (mm, default=200): 200
Chosen parameters:
Image: airplane.bmp
Thickness: 0.6-3.0mm
Pixel size: 0.3515625mm
Press enter to continue...
```

Slika 6.1. primjer unosa parametara

Nakon toga korisnik unosi tražene parametre, ako korisnik ne unese vrijednost koristiti će se standardna vrijednost navedena u zagradi. Generirani 3D model će se nalaziti u *litho* direktoriju.

Napomena:

Ako se dobiveni model ne može učitati u neki od programa (*slicer-a*) za korištenje 3D printera. Korisnik treba *.stl* datoteku otvoriti u nekom drugom programu za obradu 3D modela i ponovno spremiti.