



Факултет за информатички науки и компјутерско инженерство
ФИНКИ



ДИГИТАЛНО ПРОЦЕСИРАЊЕ НА СЛИКА

Тема:

Background subtraction



Изработија:
Стефан Лазаревски 226079
Ана Марија Краус 226135

Ментор:
проф. д-р Ивица Димитровски





Содржина

Вовед	Краток опис на целта и примената на апликацијата
Преглед на концепт	Набројување и опис на функционалностите
Технолошки компоненти	Опис на користените библиотеки и останати технологии
Метод 1	Автоматско одземање
Метод 2	Одземање со избор на квадрат
Метод 3	Одземање со избор на четка
Метод 4	Одземање на видео позадина во реално време
Метод 5	Додавање на позадина на одредена слика
Користена литература	Референци и документации





Вовед



Што е?

Нашата апликација е **Desktop GUI** алатка изградена со **Python** и **Tkinter** која овозможува одземање на позадина од статични слики и видео во реално време користејќи повеќе методи.



Инспирација

Инспирирана е од потребата за едноставни, лесни за користење алатки што овозможуваат напредна обработка на слики без скап професионален софтвер.



Цел

Целта е да се изолира предниот план со елиминирање на неважни визуелни информации, со што се подобрува видливоста и употребливоста на објектот.





Преглед на концепт

Апликацијата нуди 4 методи за одземање на позадина:

01 **Автоматско одземање**

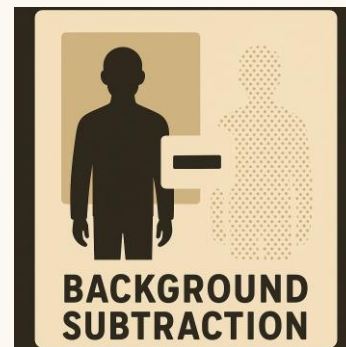
02 **Избор на квадрат**

03 **Избор на четка**

04 **Видео позадина во реално време**

Исто така нуди и функционалност за:

05 **Додавање на позадина на одредена слика**





01

Автоматско одземање

- ❖ Овој метод користи длабок невронски модел од `rembg` библиотеката базиран на U2-Net за автоматско препознавање и отстранување на позадината од сликата.
- ❖ Не бара никаква корисничка интеракција, што го прави идеален за брзи и квалитетни резултати на едноставни или добро осветлени фотографии.

```
output_image = remove(img_data)
```

Основната операција што ја
отстранува позадината

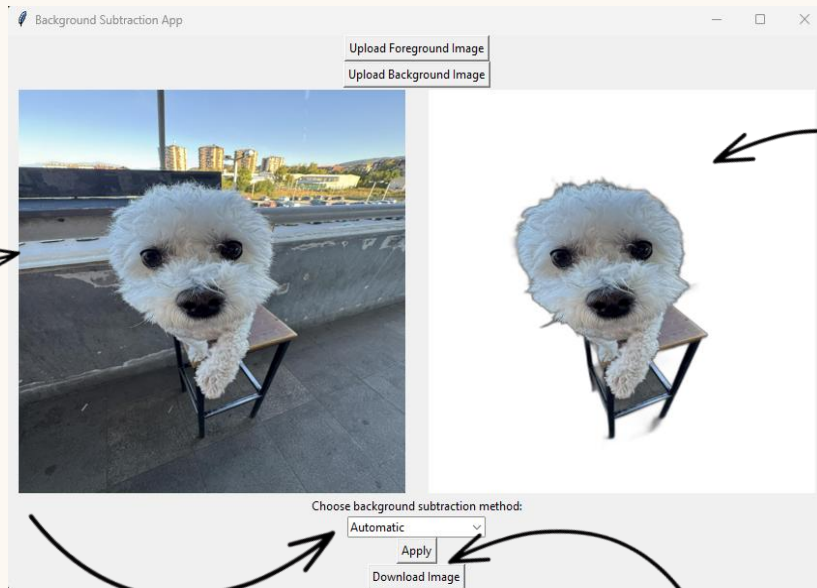




Пример:

1. Избор на слика

2. Избор на метод 1:
Automatic



3. Притискање Apply

4. Резултат





02

Избор на квадрат

- ❖ Корисникот рачно селектира правоаголен регион околу предметот на интерес, а grabCut алгоритмот врз основа на оваа селекција ја анализира текстурата и бојата за да ја одземи позадината.
- ❖ Овој метод е лесен за употреба и многу ефикасен кога објектот има јасна граница со позадината.

cv2.EVENT_LBUTTONDOWN
почетна точка на
правоаголникот

cv2.EVENT_MOUSEMOVE
визуелна селекција во
реално време

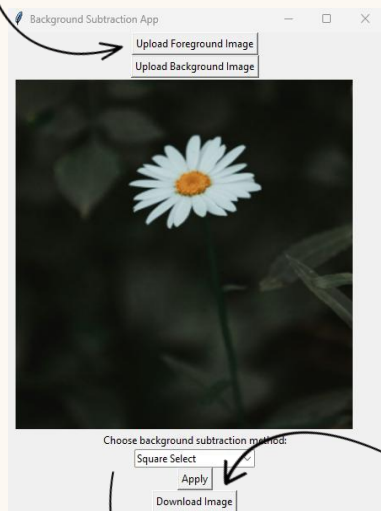
cv2.EVENT_LBUTTONUP
завршување на
селекцијата





Пример:

1. Избор на слика



2. Избор на метод 2:
Square Select

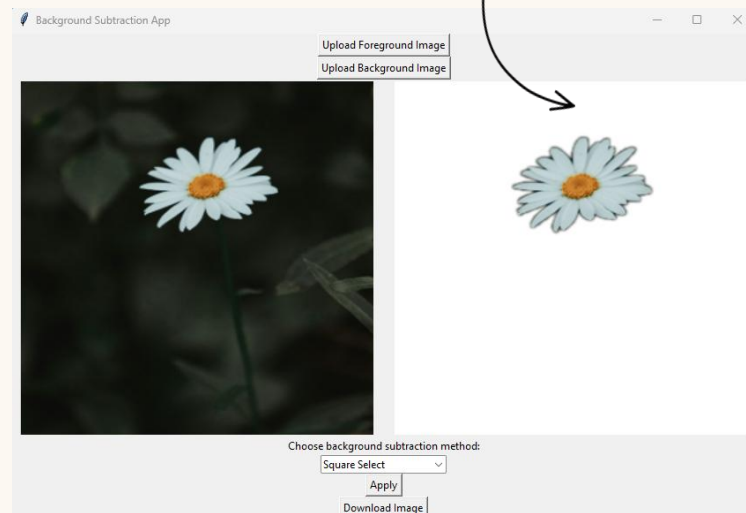
3. Притискање Apply



4. Исцртување на
правоаголникот околу
посакуваниот дел

5. Внес на „Q”

6. Резултат





03

Избор на четка

- ❖ Овој метод овозможува корисникот интерактивно да го обележи предметот со „цртање“ преку глумче, што овозможува селекција на сложени форми или деликатни рабови.

```
cv2.grabCut(img_mask, rect: None, bgdModel, fgdModel, iterCount: 10, cv2.GC_INIT_WITH_MASK)
```

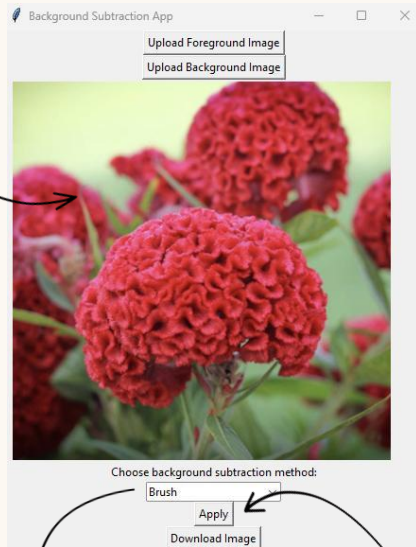
- ❖ Алгоритмот grabCut ја активира ова наредба за интелегентно одделување на предметот од позадината врз основа на маската што ја обележал корисникот. Алгоритмот користи статистички модели за да ја прошири селекцијата и прецизно ги детектира рабовите на предметот, дури и ако формата е сложена.





Пример:

1. Избор на
слика



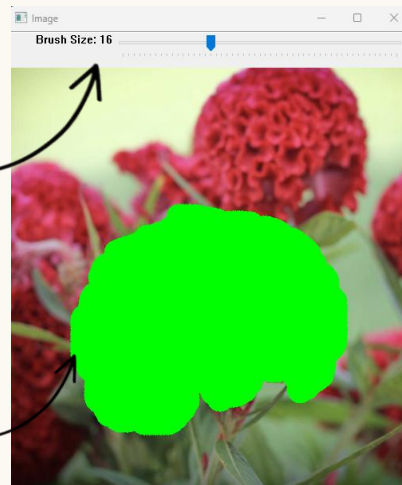
2. Избор на метод 2:
Square Select

3. Притискање Apply

4. Прилагодување на
големината на четката

```
cv2.createTrackbar('Brush Size', 'Image', brush_size, count, update_brush_size)
```

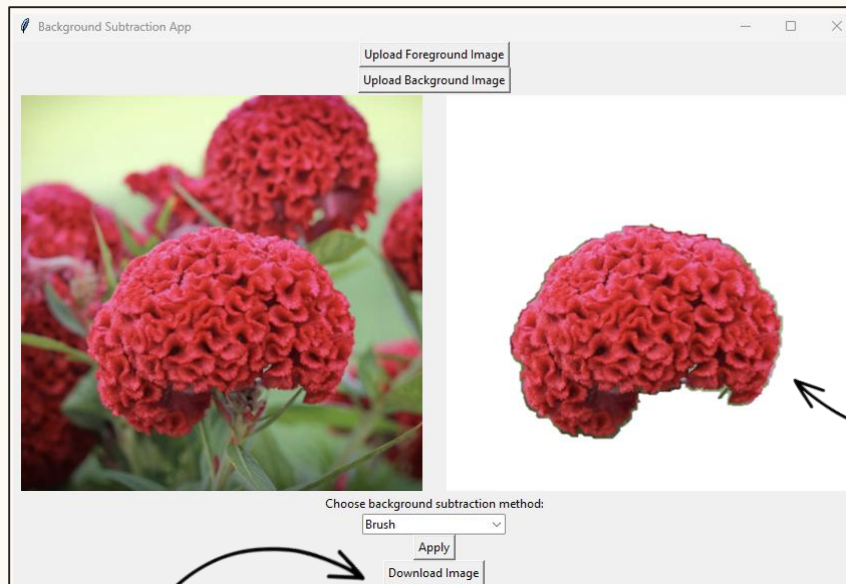
5. Исцртување на
посакуваниот дел





Пример:

6. Внес на буквата „Q“ означава крај на исцртувањето



7. Резултат



8. Зачувување на резултат





04

Видео позадина во реално време

- ❖ Преку активирање на камерата, овој метод користи Selfie Segmentation модел од MediaPipe за да врши реална сегментација на објектот во секоја видео рамка.
- ❖ Позадината се заменува во реално време користејќи вештачка интелигенција која го препознава човекот на видеото. Со секој нов кадар, MediaPipe ја генерира маската, а остатокот од сликата се заменува со нова позадина

```
output_image = np.where(condition[:, :, None], frame, resized_background)
```

Оваа линија врши пиксел-по-пиксел замена каде што луѓето се зачувуваат, а останатиот дел од сликата се заменува со нова позадина.





Пример:

1. Избор на слика за позадина

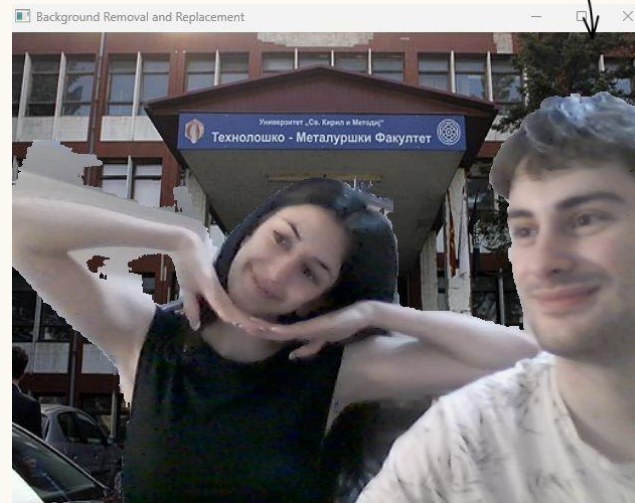
2. Избор на метод 4: Real-time видео



3. Притискање Apply

1. Избор на слика за позадина

4.Резултат: Приказ на real-time видео со посакуваната позадина



5. Видео поток трае се додека корисникот не внесе „Q“

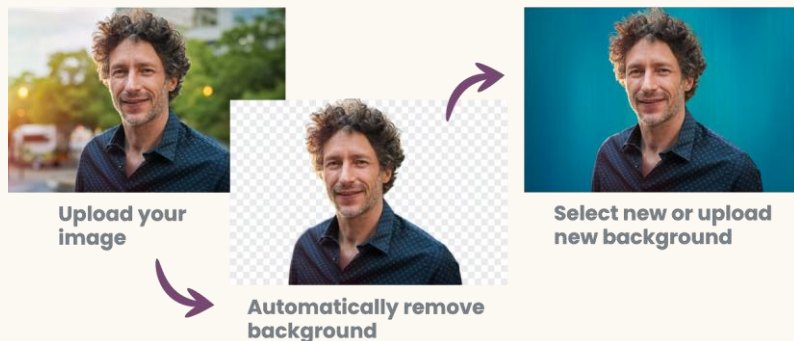




05

Додавање на позадина на одредена слика

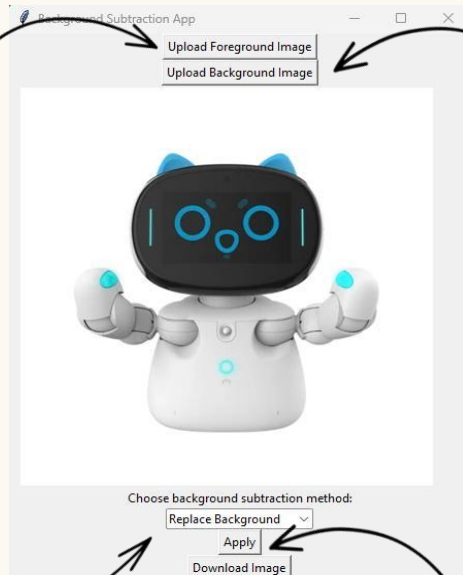
- ❖ Методот комбинира 2 слики: една слика од преден план со транспарентна позадина и уште една слика како позадина.
- ❖ Се користи алфа-блендирање за да се спојат двете слики на природен начин - заменувајќи ја алфа транспарентноста на предниот план со позадината (другата слика).



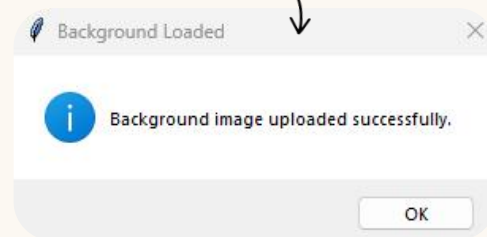


Пример:

1. Избор на слика со транспарентна позадина како **Foreground** слика



2. Избор на слика за позадина како **Background** слика



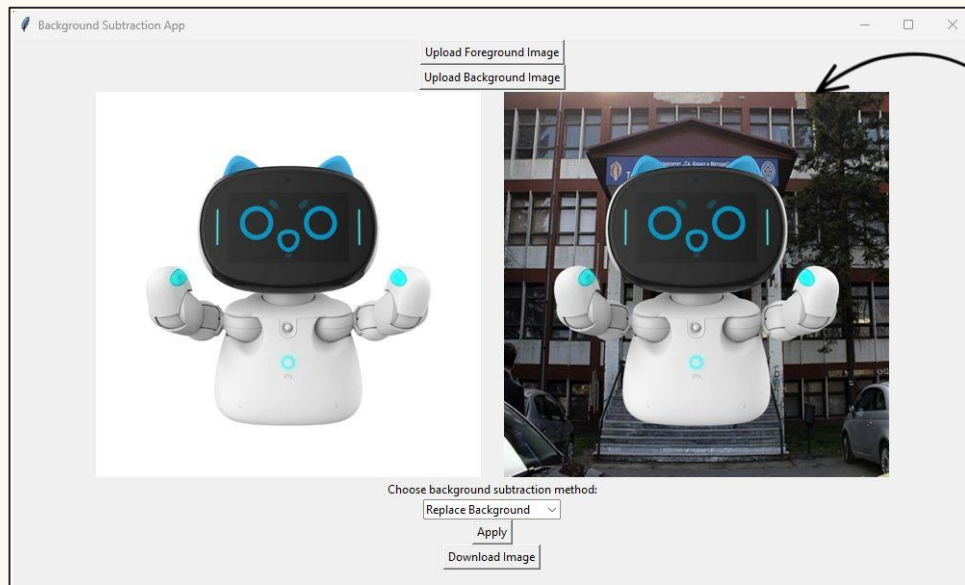
3. Избор на метод 3: Замена на позадина

4. Притискање Apply





Пример:



5. Краен резултат





Користена литература



<https://www.geeksforgeeks.org/background-subtraction-opencv/>



<https://docs.python.org/3/library/tkinter.html>



https://ai.google.dev/edge/mediapipe/solutions/vision/face_detector/python



<https://learnopencv.com/background-subtraction-with-opencv-and-bgs-libraries/>



<https://pillow.readthedocs.io/en/stable/reference/ImageTk.html>





**Ви благодариме
за вниманието!**

