Tehničko veleučilište u Zagrebu

Obrada videozapisa

Seminar iz kolegija  
Napredne tehnike programiranja web servisa

|  |  |
| --- | --- |
| Nositelj kolegija: Alen Šimec | Autor seminara: Stjepan Salopek |

U Zagrebu, 11.1.2024.

Tablica sadržaja

[Sažetak 2](#_Toc155873886)

[Podnaslov 2](#_Toc155873887)

[Uvod 3](#_Toc155873888)

[Odabir tehnologija i postavljanje okruženja 3](#_Toc155873889)

[Rad sa videozapisima 3](#_Toc155873890)

[Učitavanje i spremanje videozapisa 3](#_Toc155873891)

[Vizualna obrada i prilagođeni filtri 3](#_Toc155873892)

[Rad sa vremenskim zapisima 3](#_Toc155873893)

[Dodavanje tekstualnog sadržaja 3](#_Toc155873894)

[Vizualno-vremenska obrada 3](#_Toc155873895)

[Metode spajanja više videozapisa u jedan 3](#_Toc155873896)

[Komentar na rad sa zvukom u videozapisima 4](#_Toc155873897)

[Zaključak 4](#_Toc155873898)

Sažetak

Ovaj seminarski rad za cilj ima obradu najčešće vrste videozapisa, a to su kraći videozapisi koji su sveprisutni na društvenim mrežama. Tema ne uključuje obradu alatima sa grafičkim sučeljima, nego će se promatrati proces koji se događa u pozadini. Dakle, procesi obrade će se promatrati iz perspektive programera koji su napisali aplikacije za obradu videozapisa.

Na početku će se obrazložiti odabir tehnologija, nakon čega slijede primjeri obrade. Odabrani primjeri korišteni su na kraćem videozapisu, no služe kao temelj obrade svake vrste isječaka, filmova i serija. Sastoje se od kratkog opisa, programskog kôda te slika koje prikazuju tijek ili rezultat.

Radi jednostavnosti, seminarski rad neće obrađivati temu rada sa zvukom, ali će se radi cjelovitosti i ta tema prokomentirati.

Podnaslov

Abcd

Programski kôd

abcd

Uvod

U današnjem vremenu, obrada videozapisa važan je čimbenik u mnogim područjima djelovanja te se informacije šire pretežito vizualnim putem. Osim industrija koje izravno rade sa videozapisima poput filmske ili televizijske industrije, obrada vizualnih podataka iznimno je bitna za svako kvalitetno poduzeće. Razlog tome je broj korisnika na društvenim mrežama i raznim drugim platformama koje služe za marketing proizvoda ili usluge.

Objave u obliku kratkih videozapisa vrlo su česta pojava te sve popularne društvene mreže veliku važnost pridodaju omogućavanju korisnicima jednostavno uređivanje i objavljivanje kratkih videozapisa. Primjer ovakvog načina poslovanja su dijeljenje priča (eng. *stories*) na Instagramu i Facebooku. Takve objave služe kao jednostavan način dijeljenja informacija, a dostupne su 24 sata od objavljivanja.

Postavljanje tehnologija

Prilikom odabira i postavljanja okruženja, potrebno je naći alat za prikaz videozapisa te programski jezik i biblioteka u određenom programskom jeziku koja vrši obradu. Odabrani programski jezik je Python koji je poznat po jednostavnoj implementaciji i velikoj količini biblioteka koje služe za obradu vizualnih podataka, što je i razlog zašto se često odabire u domeni strojnog učenja. Sve korištene tehnologije otvorenog su kôda.

Za svrhe rada korištena je biblioteka MoviePy koja objedinjuje mnoge metode obrade videozapisa, a u pozadini koristi druge biblioteke popularne u radu sa matricama i slikama. Osim toga, da bi se sav kôd mogao izvršiti, potrebno je zasebno instalirati alate OpenCV i ImageMagick koji MoviePy biblioteci pružaju dodatne mogućnosti.

Za prikaz videozapisa često se koriste biblioteke i alati za izradu grafičkog sučelja kao što su Tkinter, PyQt, Kivy i wxPython. S obzirom na to da je svrha seminara na što jednostavniji način približiti obradu videozapisa, niti jedna od navedenih mogućnosti nije odabrana, nego su se videozapisi gledali ugrađenom funkcijom te otvaranjem rezultirajuće MP4 datoteke.

Postavljanje okruženja

Primjeri su rađeni u operacijskom sustavu Windows. Za svrhe rada je na platformi GitHub izrađen repozitorij naziva py\_movie\_demo, čija se poveznica nalazi na kraju seminara, u izvorima. Za pisanje i pokretanje kôda, kao i za rad sa repozitorijem, korišten je Microsoftov alat Visual Studio Code, skraćenog naziva VSCode.

Rad sa videozapisima

Na početku datoteke programskog jezika Python, potrebno je uključiti nekoliko biblioteka i postaviti određene varijable.

Biblioteka sys koristila se da bi se prilikom pokretanja omogućilo spremanje videozapisa. Primjer će biti naveden u sljedećem naslovu.

Biblioteka math korištena je kompleksnije matematičke operacije i konstante.

Objekt moviepy.config uključen je da bi se, u posljednja dva retka isječka kôda, odredila i spremila putanja do ImageMagick biblioteke.

Objekt moviepy.editor glavni je izvor za rad sa videozapisima, a pruža mnoge mogućnosti učitavanja, obrade i spremanja.

import sys

import math

import moviepy.config as moviepyconf

import moviepy.editor as editor

moviepyconf.IMAGEMAGICK\_BINARY =  
 'C:\\putanja\_do\_datoteke\\magick.exe'

Učitavanje i spremanje videozapisa

Programski isječak na kraju trenutnog dijela prikazuje uobičajen slijed naredbi prilikom obrade videozapisa. U početku je korištena Pythonova with..as sintaksa koja će originalni videozapis otključati i omogućiti drugim programima da ga koriste dok se skripta u potpunosti ne izvrši.

Samo učitavanje videozapisa vrši se pozivanjem konstruktora klase VideoFileClip koji vraća objekt sa raznim funkcijama.

Nakon toga može u slijediti jedna od metoda obrade koja će se kasnije spomenuti, a koja zatim vraća zaseban videozapis, ostavljajući izvorni netaknutim.

Zastavica write\_vf\_flag služi za dodatne mogućnosti pokretaču skripte. Točnije, ukoliko se u naredbenom retku prilikom pokretanja pošalje varijabla –write-videofile=true, tada će se obrađeni videozapis spremiti. U protivnome, skripta će ga prikazati bez spremanja na disk.

With editor.VideoFileClip('video.mp4') as original:

  edited\_clip = FUNKCIJA\_OBRADE(original)

  write\_vf\_flag = len([

    arg for arg in sys.argv

    if arg.lower() == '--write\_videofile=true'

  ]) > 0

  if write\_vf\_flag:

    edited\_clip.write\_videofile('output.mp4')

  else:

    edited\_clip.preview()

asdf

Vizualna obrada i prilagođeni filtri

Sljedeći primjer pokazuje česti slučaj vizualne obrade. U prvom dijelu je pomoću funkcije fl\_image izvršena obrada svake sličice videozapisa. Točnije, na svakoj sličici RGB vrijednosti su pomnožene sa brojevima 1, 1 i 10, što je stvorilo videozapis sa prenaglašenom plavom bojom.

U drugom dijelu primjera, stvorena je nova funkcija obrade koja će zamijeniti vrijednosti RGB kanala u BGR. U stvarnosti stvorena funkcija može biti puno kompleksnija. Nakon što se stvori, obrada se vrši prosljeđivanjem njenog funkcijskog objekta metodi fx.

blue\_vid = original.fl\_image(lambda frame: frame \* [1, 1, 10])

def change\_color\_channels(clip):

  return clip.fl\_image(lambda frame: frame[..., [2, 0, 1]])

custom\_fx\_vid = original.fx(change\_color\_channels)

Rad sa vremenskim zapisima

Na sličan način možemo raditi i u vremenskom aspektu. Funkcija fl\_time omogućava mijenjanje vremenske crte videozapisa.

U sljedećem programskom kôdu, napisana je funkcija koja uzima prve dvije sekunde videozapisa te ih periodički prikazuje, prvo unaprijed pa unazad, frekvencijom od jednog perioda po sekundi. Slučaj uporabe ovog učinka bili bi GIFovi koji se često koriste na društvenim mrežama.

Funkcija radi na način da lambda funkciji daje parametar t koji označuje vrijeme sličice videozapisa. Kada se navedeni parametar promjeni, promijeniti će se i sličica. Konkretno, … NASTAVITI ODAVDE

# Apply complex time effects with time filter

#   Add 1 to sin to avoid negative values

#   Set duration to avoid infinite loops

# Resulting video takes first 2 seconds of video and oscillates it in a loop (1x per second)

time\_fx\_vid = original                   \

  .fl\_time(lambda t: 1 + math.sin(2 \* math.pi \* t))   \

  .set\_duration(original.duration)

Dodavanje tekstualnog sadržaja

Vizualno-vremenska obrada

Metode spajanja više videozapisa u jedan

Komentar na rad sa zvukom u videozapisima

Zaključak

Izvori

Poveznica na repozitorij: <https://github.com/shtef21/py_video_demo>