Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**ЛАБОРАТОРНАЯ РАБОТА №5**

**Дисциплина: Алгоритмы цифровой обработки мультимедиа**

**Тема: Выявление движения на видео**

Работу выполнил: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А. А. Эзри

Направление подготовки: 02.03.02 Фундаментальная информатика и информационные технологии

Преподаватель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.А. Крамаренко

**Цель работы:** реализовать алгоритм выявления движения на видео.

**Ход работы:**

**Задание 1.** Реализовать метод, который читает видеофайл и записывает в один файл только ту часть видео, где в кадре было движение, можно воспользоваться примерами.

Был реализован метод, который принимает в качестве параметров путь к видеофайлу, размер ядра размытия, значение порога, и минимальную площадь.

В этом методе реализован следующий алгоритм:

1. Начать чтение из файла, прочитать первый кадр, перевести в чёрно-белый цвет и применить размытие Гаусса;
2. Подготовить файл для записи;
3. Далее начать цикл, который завершится по завершению файла, внутри этого цикла:
   1. скопировать старый кадр;
   2. прочитать новый кадр, перевести в чёрно-белый цвет, применить размытие Гаусса;
   3. если чтение неуспешно, остановить цикл;
   4. найти разницу между двумя кадрами в отдельный фрейм (frame\_diff) - cv2.absdiff;
   5. провести операцию двоичного разделения для фрейма (frame\_diff) - cv2.threshold;
   6. найти контуры объектов для фрейма (frame\_diff) - cv2.findContours;
   7. пройтись по контурам объектов для фрейма (frame\_diff) и найти контур площадью большей, чем наперед заданный параметр - cv2.contourArea;
   8. если такой контур найден, значит было движение, записать кадр в файл.

Для операции порога используем параметр cv2.THRESH\_BINARY – двоичное разделение. Для нахождения контуров используем cv2.RETR\_EXTERNAL (нам интересны только внешние контуры) и cv2.CHAIN\_APPROX\_SIMPLE (аппроксимация).

**Задание 2.** Провести эксперименты, выбирая различные значения параметров: размытие Гаусса, граница разделения для метода threshold, площадь минимального объекта, подобрать оптимальные значения параметров для данного видео.

Были проверены следующие варианты:

|  |  |  |
| --- | --- | --- |
| Размер ядра | Пороговое значение | Минимальная площадь |
| 7 | 32 | 10 |
| 7 | 32 | 20 |
| 7 | 50 | 10 |
| 7 | 64 | 10 |

А также с размером ядра 5 и 9 (заметной разницы не было).

**Листинг программы**

Файл Lab\_5.py

import cv2  
import numpy as np  
  
  
def frame\_preprocess(frame: np.ndarray, blur\_kernel\_size: int):  
 frame = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)  
 frame = cv2.GaussianBlur(frame, (blur\_kernel\_size, blur\_kernel\_size), 0)  
 return frame  
  
  
def motion\_detection(path: str, blur\_kernel\_size: int, thresh: float, min\_area: float, write: bool):  
 video = cv2.VideoCapture(path)  
 w = int(video.get(cv2.CAP\_PROP\_FRAME\_WIDTH))  
 h = int(video.get(cv2.CAP\_PROP\_FRAME\_HEIGHT))  
 fps = int(video.get(cv2.CAP\_PROP\_FPS))  
 fourcc = cv2.VideoWriter.fourcc(\*'mp4v')  
  
 ok, frame\_prev = video.read()  
 if not ok:  
 print("Не удалось открыть видеопоток")  
 exit(1)  
 frame\_prev = frame\_preprocess(frame\_prev, blur\_kernel\_size)  
 if write:  
 video\_writer = cv2.VideoWriter(f"Lab\_5\_output\_{blur\_kernel\_size}\_{thresh}\_{min\_area}.mp4", fourcc, fps, (w, h))  
 else:  
 video\_writer = None  
 while True:  
 ok, frame\_raw = video.read()  
 if not ok or cv2.waitKey(1) & 0xFF == 27:  
 break  
  
 frame = frame\_preprocess(frame\_raw, blur\_kernel\_size)  
 frame\_diff = cv2.absdiff(frame\_prev, frame)  
 ret, frame\_threshold = cv2.threshold(frame\_diff, thresh, 255, cv2.THRESH\_BINARY)  
 contours, hierarchy = cv2.findContours(frame\_threshold, cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE)  
 has\_motion = False  
 for cnt in contours:  
 area = cv2.contourArea(cnt)  
 if area >= min\_area:  
 has\_motion = True  
 break  
  
 if has\_motion:  
 print('Motion detected')  
 cv2.putText(frame\_threshold, 'Motion detected', (10, 35), cv2.FONT\_HERSHEY\_SIMPLEX, 1, (100, 255, 100), 2, cv2.LINE\_AA)  
 if write:  
 video\_writer.write(frame\_raw)  
 cv2.imshow('video', frame\_threshold)  
 frame\_prev = frame  
  
 video.release()  
 if write:  
 video\_writer.release()  
 cv2.destroyAllWindows()  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 path = '../../Video/Lab\_5\_main\_video.mov'  
 write = True  
  
 motion\_detection(path, 7, 32, 10, write)

**Ответы на вопросы**

1. Опишите алгоритм выявления движения на видео.
2. Опишите особенности применения новых для Вас методов