Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Новгородский государственный университет имени Ярослава Мудрого»

Кафедра информационных технологий и систем

**Программа обработки видео**

Лабораторная работа по дисциплине:

«Распознавание образов и обработка изображений»

Выполнил студент группы 8091:

\_\_\_\_\_\_\_\_\_\_ / Кузин И.А.

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

Принял преподаватель:

\_\_\_\_\_\_\_\_\_\_ / Назаров A. Г.

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

**Великий Новгород**

**2022**

1. **Цель работы**

Изучение основ обработки потока изображений и алгоритмов детекции движения.

1. **Описание среды окружения**

Для решения поставленной задачи использовалось следующее окружение:

* ОС: Windows 10
* Среда программирования: Qt версии 6.2.2
* Библиотека OpenCV 4.5.5
* Компилятор: Qt 6.2.3 MinGW 64-bit

1. **Описание работы программы-детектора движения**

Для реализации алгоритма фоновой сегментации были использованы средства Opencv. Использовались методы:

*createBackgroundSubtractorMOG2 (history, varTrashold, detectShadows),*

где *history* – длинна истории (кол-во кадров для выборки фонового изображения).

*varTrashold* - Пороговое значение для квадрата расстояния Махаланобиса между пикселем и моделью, чтобы определить, хорошо ли пиксель описывается фоновой моделью.

*detectShadows* - Если значение **true**, то алгоритм отметит тени, которые обнаружит, в результате чего незначительно снижается скорость работы алгоритма детекции.

Видео состоит из кадров, где для каждого кадра применяется набор функций:

1. *segmentor->apply(frame, fgmask);*

Применяя вычитатель сегментов с захваченным кадром и маской переднего плана, получаем информацию о сцене и извлекаем фон и передний план, после чего на выходе, генерируется изображение в градациях серого с заполненным чёрным фоном, а часть переднего плана заполняется не блочными пикселями.

1. Для того, чтобы избавиться от лишних шумов на маске переднего плана и выделить интересующие нас объекты:

*cv::threshold(fgmask, fgmask, 25, 255, cv::THRESH\_BINARY);*

Позволяет избавиться от тёмных шумов с маски переднего плана.

*cv::Mat kernel = cv::getStructuringElement(cv::MORPH\_RECT, cv::Size(noise\_size, noise\_size));*

Метод в результате работы возвращает структурирующий элемент указанного размера и формы для морфологических операций.

*cv::erode(fgmask, fgmask, kernel);*

Выполняется размытие с помощью структурирующего элемента.

*cv::dilate(fgmask, fgmask, kernel, cv::Point(-1,-1), 3);*

Данная функция расширяет исходное изображение, используя указанный структурирующий элемент, который определяет форму окрестности пикселя, по которой берется максимум.

1. Для механизма поиска контуров объектов на маске используются данные методы:

*cv::findContours(fgmask, contours, cv::RETR\_TREE*, *cv::CHAIN\_APPROX\_SIMPLE)*

1. В случае обнаружения контуров, возможно выполнить предположение, что движение было успешно обнаружено.
2. На данном этапе происходит отрисовка прямоугольников для каждого контура на кадре захвата.

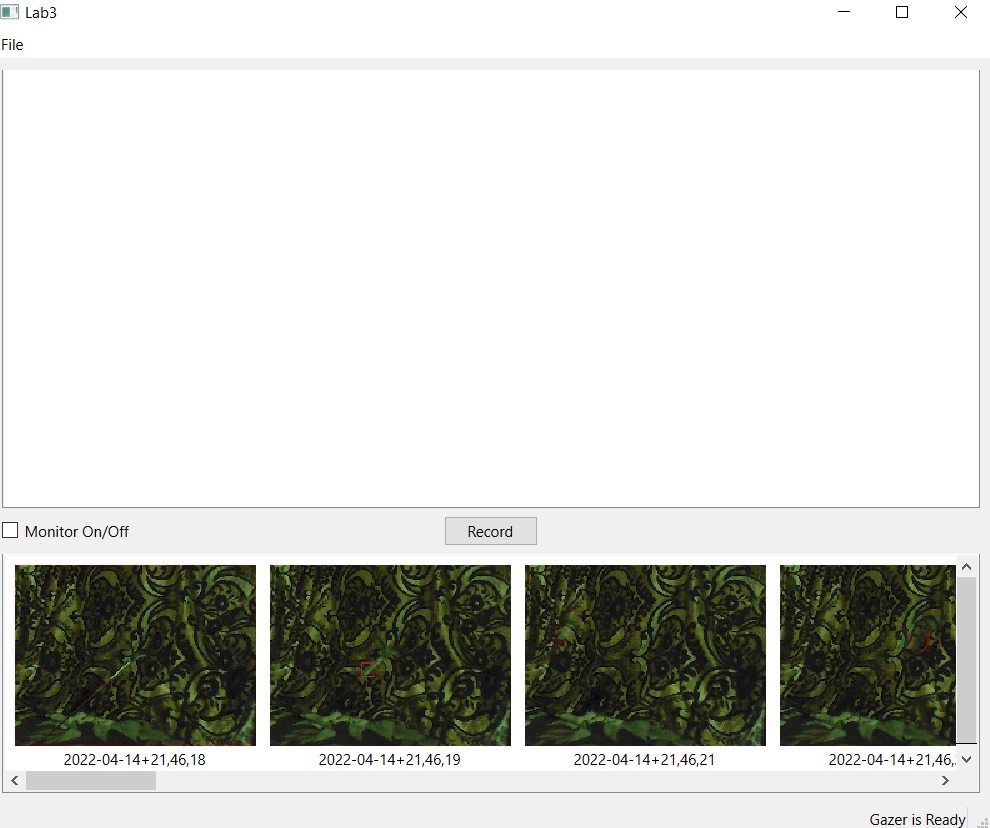


Рисунок 1. Интерфейс главного экрана программы

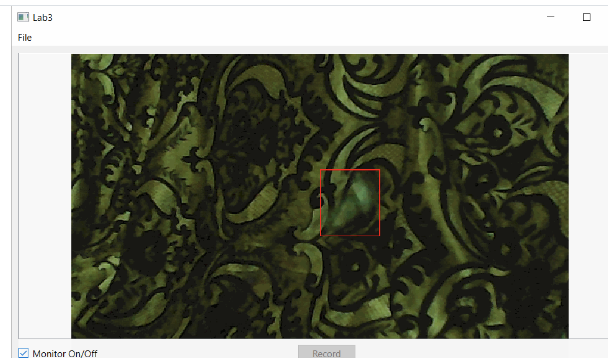


Рисунок 2. Детекция обнаруженного движения на загруженном видеофайле

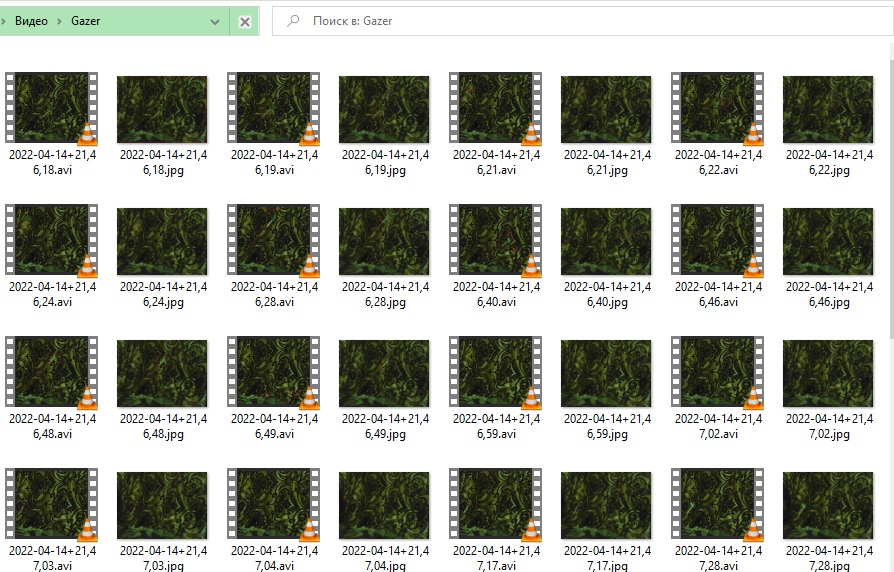


Рисунок 3. Сохраненные видеофайлы в процессе работы программы

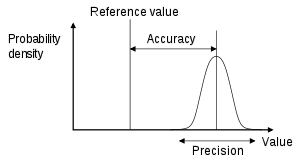
1. **Производимые математические расчеты**

Среднее значения пикселя:

Среднеквадратичное отклонение:

Вычисления производятся поканально.

Описание разницы между понятиями точность и кучность:



Точность является показателем того, насколько матожидание близко к истинному, а кучность то, какой результаты имеют разброс (чем выше разброс, тем выше среднеквадратичное отклонение).

1. **Ответы на вопросы**
2. Возможность обнаруживать движение из готового видеофайла есть, для этого необходимо покадрово считывать загруженный видеофайл, обнаружение образа будет аналогичным, что и в трансляции реального времени.
3. В теории, распараллелить обнаружение кадров и считывание видеопотока возможно, для этого потребуется наладить связь между двумя потоками для отправки кадра, но для вывода изображения с образом, необходимо реализовать синхронизацию кадров обработки и кадров вывода. В результате, пользователю все равно потребуется выдавать изображение с образом, поэтому в постановке данной задачи в разделении на потоки нет необходимости, т.к. данный механизм приведет только к усложнению в реализации алгоритмов работы программы.
4. **Исходники созданной программы**

Исходный код данного проекта доступен к просмотру на GitHub по ссылке:

https://github.com/ivan-kuzin/Pattern-recognition

Для запуска программы, необходимо запустить файл ***Lab3.exe*** из каталога release. (Приложен в формате архива).

1. **Вывод**

В ходе выполнения данной лабораторной работы, мною были изучены основы обработки потока изображений и алгоритмами детекции движений, а также реализовал программу с использованием полученных знаний.