**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
 РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет информационных технологий**

**Кафедра параллельных вычислений**

**ОТЧЕТ**

**О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

студента (ки) \_\_\_\_2\_\_\_\_\_ курса, \_\_19211\_\_ группы

**Олимпиева Юрия Юрьевича**

Направление 09.03.01 – «Информатика и вычислительная техника»

Преподаватель:

**Н.А. Беляев**

2020\_год

**СОДЕРЖАНИЕ**

Цели……….………………………………………………………………….3

Задание...….………………………………………………………………….3

Описание работы....………………………………………………………….3

Заключение.………………………………………………………………….3

Приложение 1………………………………………………………………..4

Приложение 2………………………………………………………………..5

Приложение 3………………………………………………………………..6

**Цели**

1. Ознакомиться с программированием периферийных устройств на примере ввода данных с Web-камеры с использованием библиотеки OpenCV. **Задание**

1. Реализовать программу с использованием OpenCV, которая получает поток видеоданных с камеры и выводит его на экран.

2. Выполнить произвольное преобразование изображения.

3. Измерить количество кадров, обрабатываемое программой в секунду. Оценить долю времени, затрачиваемого процессором на обработку (ввод, преобразование, показ) видеоданных, получаемых с камеры.

4. Составить отчет по лабораторной работе.

**Описание работы**

* Подключил библиотеку openCV к проекту Microsoft Visual Studio.
* Написал программу для получения видео с встроенной web-камеры и вывода видео на экран (приложение 1).
* Реализовал преобразование цвета выводимого видео (приложение 2).
* Реализовал переворот картинки видео на 180 градусов через матричные преобразования (приложение 2).
* Измерил посредством методов первой лабораторной работы количество кадров, обрабатываемое программой в секунду (n ~16-20) (приложение 3).
* Составил отчёт о выполненной работе.

**Заключение**

Изучил основы работы с библиотекой openCV, познакомился с программированием периферийных устройств.

Приложение 1

#include <iostream>

#include <opencv2/opencv.hpp>

#include <opencv2/highgui.hpp>

#include <opencv2/highgui/highgui\_c.h>

using namespace std;

using namespace cv;

int main(void) {

int key = 0;

VideoCapture flow(0);

if (!flow.isOpened())

exit(-1);

namedWindow("Window",CV\_WINDOW\_AUTOSIZE );

while (1) {

Mat frame;

flow >> frame;

Point center = Point2d(320.0, 240.0);

Mat M = getRotationMatrix2D(center, 180.0, 1.0);

warpAffine(frame, frame, M, Size(frame.size()));

cvtColor(frame, frame, COLOR\_BGR2GRAY);

imshow("Window", frame);

key = waitKeyEx(1);

if (key == 27)

break;

}

return 0;

}

**Команды компиляции:**

gcc -O OPEN\_CV.cpp

**Команды для запуска:**

a.exe

Приложение 2

#include <iostream>

#include <opencv2/opencv.hpp>

#include <opencv2/highgui.hpp>

#include <opencv2/highgui/highgui\_c.h>

using namespace std;

using namespace cv;

int main(void) {

int key = 0;

VideoCapture flow(0);

if (!flow.isOpened())

exit(-1);

int Start;

int Finish;

int frame\_counter = 0;

namedWindow("Window",CV\_WINDOW\_AUTOSIZE );

Start = clock();

while (1) {

frame\_counter++;

Mat frame;

flow >> frame;

Point center = Point2d(320.0, 240.0);

Mat M = getRotationMatrix2D(center, 180.0, 1.0);

warpAffine(frame, frame, M, Size(frame.size()));

cvtColor(frame, frame, COLOR\_BGR2GRAY);

imshow("Window", frame);

key = waitKeyEx(1);

if (key == 27)

break;

}

Finish = clock();

printf("Result time:\t%f", ((float)(Finish - Start) / (float)CLK\_TCK)/ frame\_counter);

return 0;

}

Приложение 3

