**МИНИСТЕРСТВО ОБРАЗОВАНИЯ МОСКОВСКОЙ ОБЛАСТИ**

**Государственное бюджетное профессиональное образовательное учреждение Московской области «Люберецкий техникум имени Героя Советского Союза, летчика-космонавта Ю.А.Гагарина»**

**ОТЧЕТ**

**УЧЕБНОЙ ПРАКТИКИ**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, имя, отчество студента)

по профессиональному модулю

**\_\_\_\_\_\_\_\_***ПМ.01 «Разработка модулей программного обеспечения***\_\_\_\_\_\_\_\_**

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ для компьютерных систем»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Специальность \_*09.02.07 "Информационные системы и программирование"*\_

Код, название

Курс \_3\_ Группа № \_195\_

Период практики с «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 20\_\_\_ г. по «\_\_\_\_» \_\_\_\_\_\_\_\_\_ 20\_\_ г.

Руководитель практики

от техникума \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Люберцы 2021

**Оглавление**

**1.**

**2.**

**3.**

**4. Введение**

**1.Индивидуальное задание на практику**

**2.Дневник прохождения практики**

Студента 3 курса

Дьяков Артём Сергеевич

Специальность: ИС и программирование

|  |  |  |  |
| --- | --- | --- | --- |
| **№ п/п** | **Дата** | **Краткое содержание выполнения работ** | **Отметка руководителя практики**  **(оценка, подпись)** |
| 1 | 2 |  | 3 |
|  |  |  |  |

Руководитель практики

от техникума \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

подпись Фио

# Введение

Целью учебной практики является: закрепление теоретических знаний, полученных при изучении базовых дисциплин; изучение организационной структуры предприятия и действующей на нем системы управления; ознакомление с содержанием основных работ и исследований, выполняемых на предприятии или в организации по месту прохождения практики.

Целью учебной практики является реализация компьютерного зрения с помощью библиотеки OpenCV, и разработка приложения с помощью WPF и 3DS Max

Для достижения данной цели требуется решить следующие задачи:

1. Знакомство с литературой
2. Изучить библиотеку OpenCV
3. Реализация приложения на WPF

В ходе учебной практики планируется освоение следующего вида деятельности: «Разработка модулей программного обеспечения для компьютерных систем » (ПК):

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств

ПК 1.4 Выполнять тестирование программных модулей

ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода

ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.

ПК 2.1. Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент.

ПК 2.2. Выполнять интеграцию модулей в программное обеспечение.

ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

# Изучение предметной области

OpenCV — это open source библиотека компьютерного зрения, которая предназначена для анализа, классификации и обработки изображений. Широко используется в таких языках как C, C++, Python и Java.

Мой курс помог мне разобраться и научил работать с 3D studio Max, что поможет мне в будущем с работой.

Разработка приложения на WPF помогла нам научиться разрабатывать соединять простейшее приложение с базой данных SQL management Studio 18.

# Изучение программных средств

1. Для подключения OpenCV на python достаточно:

* Открываем PyCharm и нажимаем New project, выбираем папку, где хотим, чтобы был наш проект, и окружение Virtualenv, слева появятся папки
* File - Settings - Project Interpretater - шестеренка(add) - new enviroment. После этого Apply. Слева появится папка venv. Теперь все установленные библиотеки будут в виртуальном окружении и не будут засорять память
* Там же нажимаем на +, вводим opencv-python, install. Теперь появятся библиотеки нужные (https://youtu.be/iMM4TImNJFM)
* Заходим в Terminal Local python Имя\_файла.py

1. Недостатком C# является само подключение, для которого нужны изменения в фалах проекта и добавление дополнительных библиотек:
   * Скачать Open CV с сайта https://opencv.org/releases/ и выбрать версию 4.5.3 для Windows.
   * Установить Open CV (желательно сохранить в диске С или D D:\opencv)
   * Создаем пустой проект C++
   * Кликаем по названию проекта и кликаем на "Добавить" --> “Создать элемент” --> “Файл C++(.cpp)”
   * После этого по названию проекта кликаем правой кнопкой мыши и выбираем "Свойства"
   * Выбираем конфигурацию All Configurations, платформу х64 (в двух местах))
   * Находим C/C++ --> Раскрываем и находим "Общие" --> Находим каталоги включаемых файлов --> указываем путь openCv "...\opencv\build\include"
   * Далее выбираем "Компоновщик" --> "Общие" --> "Дополнительные каталоги библиотек" --> указываем путь "...\opencv\build\x64\vc15\lib"
   * Так же "Компоновщик" находим "Ввод" --> "Дополнительные зависимости и из папки lib выбираем opencv\_world453d.lib
   * Из папки "...opencv\build\x64\vc15\bin" копируете opencv\_world453d.dll в папку ...\Название вашего проекта\x64\Debug
2. WPF это библиотека для создания десктопных приложений для работы с ним мы включаем Visual Studio и создаём проект с WPF.
3. 3Ds Max это среда для создания 3D моделей и образов.

# OpenCV

1. Теория

Файл Contour\_main.cpp - это код по нахождению контуров

Файл momentmain.cpp - это дополнительно использование ползунка createTrackbar

Файл VideoTH.cpp - это оператор Собеля

Оператор Собеля — дискретный дифференциальный оператор, вычисляющий приближённое значение градиента яркости изображения. Результатом применения оператора Собеля в каждой точке изображения является либо вектор градиента яркости в этой точке, либо его норма. Используется в области обработки изображений, в частности, часто применяется в алгоритмах выделения границ.

VideoCapture cap – оператор для использования веб камеры

detectMultiScale — общая функция для распознавания как лиц, так и объектов. Чтобы функция искала именно лица, мы передаём ей соответствующий каскад.

Функция detectMultiScale принимает 4 параметра:

* + - 1. Обрабатываемое изображение в градации серого.
* Параметр scaleFactor. Некоторые лица могут быть больше других, поскольку находятся ближе, чем остальные. Этот параметр компенсирует перспективу.
* Алгоритм распознавания использует скользящее окно во время распознавания объектов. Параметр minNeighbors определяет количество объектов вокруг лица. То есть чем больше значение этого параметра, тем больше аналогичных объектов необходимо алгоритму, чтобы он определил текущий объект, как лицо. Слишком маленькое значение увеличит количество ложных срабатываний, а слишком большое сделает алгоритм более требовательным.
* minSize — непосредственно размер этих областей.

Сохранение изображения

import cv2

image = cv2.imread("./импорт/путь.расширение")

cv2.imwrite("./экспорт/путь.расширение", image)

Рисование прямоугольников

import cv2

output = image.copy()

cv2.rectangle(output, (2600, 800), (4100, 2400), (0, 255, 255), 10)

viewImage(output, "Обводим прямоугольником лицо пёсика")

Эта функция принимает 5 параметров:

* Само изображение.
* Координата верхнего левого угла (x1, y1).
* Координата нижнего правого угла (x2, y2).
* Цвет прямоугольника (GBR/RGB в зависимости от выбранной цветовой модели).
* Толщина линии прямоугольника.

Распознавание лиц

[practic/openCV/python at main · Grommester/practic (github.com)](https://github.com/Grommester/practic/tree/main/openCV/python)

Конторы и центры масс

[practic/openCV at main · Grommester/practic (github.com)](https://github.com/Grommester/practic/tree/main/openCV)

Распознавание дорожной разметки

[practic/openCV/doroda at main · Grommester/practic (github.com)](https://github.com/Grommester/practic/tree/main/openCV/doroda)

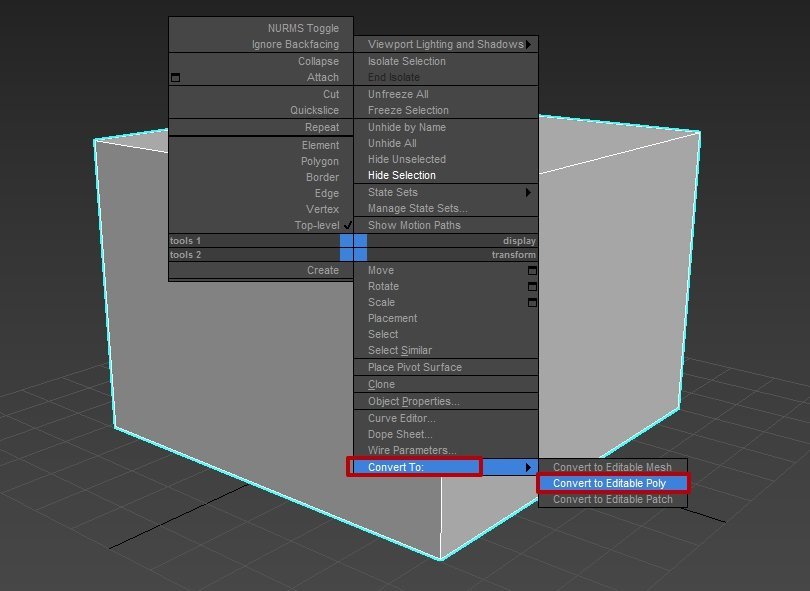
Просмотр видео с веб камеры оператор собеля

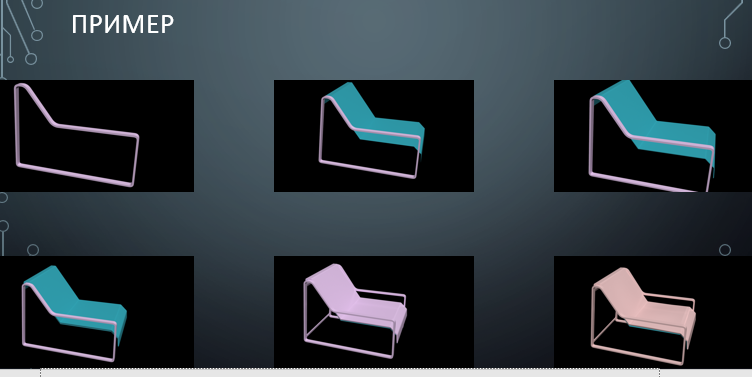
[practic/openCV/1 at main · Grommester/practic (github.com)](https://github.com/Grommester/practic/tree/main/openCV/1)

Скриншоты выполнения расположены по ссылкам выше

# Ваш курс

* **Файлы** с расширением .**max** **открываются** в программе 3ds **max** стандартным способом
* Полигон — плоскость, состоящая из нескольких точек в пространстве, соединённых рёбрами.
* Полигональное моделирование — это один из видов трёхмерного моделирования, позволяющий моделировать любой объект (ну или почти любой) и за счёт соединения полигонов создавать из них группы и формировать нужный облик модели.
* Создаём объект для работы и конвертируем его в Editable Poly
* Рассмотрим подобъекты Editable Poly:
* **Vertex (вершины)**
* **Edge (рёбра)**
* **Border (граница)**
* **Polygon (полигон)**
* **Element (элемент)**





**WPF**

# Выводы

По окончании учебной практики были решены следующие задачи:

1. Знакомство с литературой
2. Изучить библиотеку OpenCV
3. Реализация приложения на WPF

В ходе этой практики были сформированы следующие профессиональные компетенции:

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием.

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств

ПК 1.4 Выполнять тестирование программных модулей

ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода

ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.

ПК 2.1. Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействия компонент.

ПК 2.2. Выполнять интеграцию модулей в программное обеспечение.

ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

**4. Список литературы**

**5. ПРИЛОЖЕНИЯ (при наличии)**