

Распознавание лиц на изображении с помощью Microsoft Visual Studio и библиотеки Emgu CV

Маринчук Александр Сергеевич

Приамурский государственный университет им. Шолом-Алейхема

Студент

Аннотация

В данной статье рассмотрен инструмент Microsoft Visual Studio и подгружаемая библиотека Emgu CV с помощью которых можно распознать лицо человека на загружаемой фотографии. Была создана форма, куда загружаются картинки в формате JPG или PNG, а также подключена библиотека Emgu CV, благодаря которой в массив заносится список прямоугольников вокруг лица человека, рисуемый методом Graphics.

Ключевые слова: Emgu CV, Microsoft Visual Studio, распознавание лиц.

Face recognition on image using Microsoft Visual Studio and Emgu CV library

Marinchuk Alexander Sergeevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article discusses the Microsoft Visual Studio tool and the Emgu CV downloadable library with which you can recognize a person's face in a downloaded photo. A form was created where pictures in JPG or PNG format are uploaded, and the Emgu CV library is connected, thanks to which a list of rectangles around a person's face is drawn into the array, drawn using the Graphics method.

Keywords: Emgu CV, Microsoft Visual Studio, face recognition.

1. Введение

1.1 Актуальность исследования

В настоящее время становится все актуальнее распознавание всевозможных объектов на фотографиях, видео или обнаружение предметов даже в режиме реального времени. Все это становится реальным благодаря нейронным сетям, которые, благодаря определенным алгоритмам, могут находить и обозначать нужные элементы. Уже мало кого удивит данным фактом, так как существуют различные инструменты, помогающие опробовать данные технологии. Например, маски Instagram, где распознается лицо человека и на него накладываются различные эффекты. Люди активно

используют данные инструменты и не только в развлекательных целях, а даже в спецслужбах, где зачастую распознавание лица человека становится ключевым моментом.

1.2 Обзор исследований

В статье А.А. Куликова, Д. В. Демкина и А.Е Мелькова описаны результаты выполненного исследования влияния степени сжатия изображения на характеристики функционирования алгоритма распознавания. [1]. С.Г. Небаба и А.А. Захарова описали алгоритм формирования трехмерной модели, предоставили результаты генерации изображений лица с разными ракурсами на основе этой модели [2]. Н. Матюков и другие рассмотрели задачу распознавания лица на изображении с помощью масок [3]. Проводится обзор методов решения задачи поиска фотографий с человеком с различных мероприятий по его лицу, используются современные методы детекции лиц на изображениях в статье А.Р Идятова., В.Д. Фоменко и А.А. Козько [4]. Y. Wen, K. Zhang, Z. Li, Y. Qiao создали новый метод для обучения нейронной сети, с помощью которого можно качественнее распознавать лицо человека [5].

1.3 Цель исследования

Целью данной статьи является создание приложения для распознавания лиц в Microsoft Visual Studio с использованием библиотеки Emgu CV.

2. Методы исследования

Инструментом для достижения цели станет Microsoft Visual Studio. Microsoft Visual Studio — линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств [6].

Для начала работы следует скачать и установить Microsoft Visual Studio с официального сайта (рис. 1).

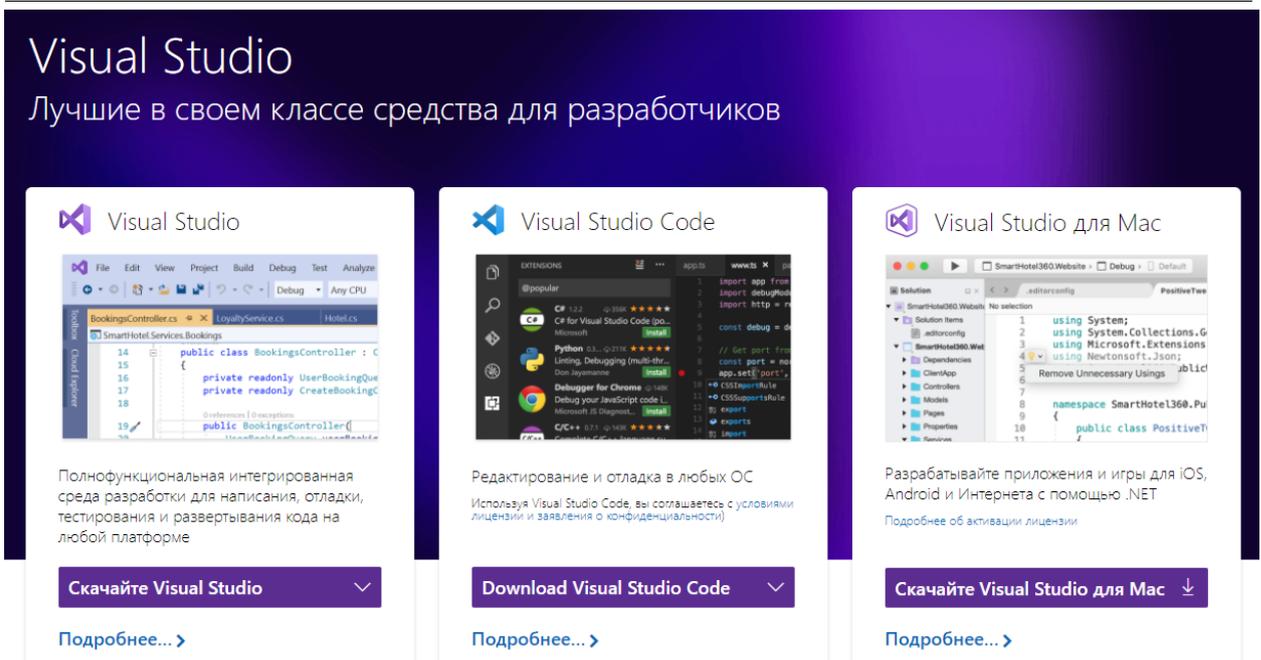


Рисунок 1 – Скачивание приложения

После установки программы следует создать новый проект (рис. 2).



Рисунок 2 – Создание проекта

Далее необходимо добавить на форму кнопку, которая будет открывать изображение для распознавания и компонент pictureBox, где будет выводиться картинка (рис. 3). Для добавления их на форму следует дважды кликнуть на него.

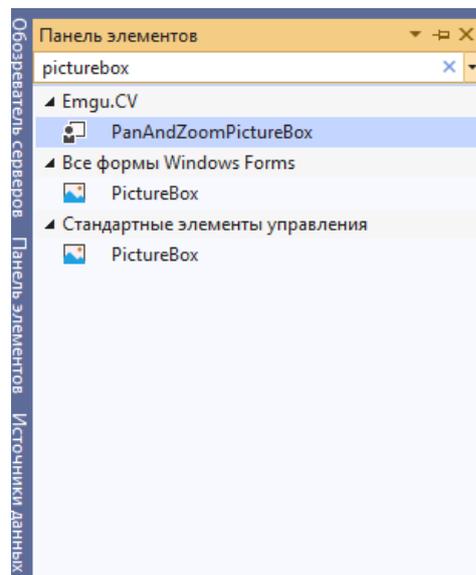


Рисунок 3 – Добавление компонентов

Изменим некоторые свойства у добавленных компонентов. У кнопки поменяем текст, а также ее размер и расположим в нижнем правом углу формы (рис. 4).

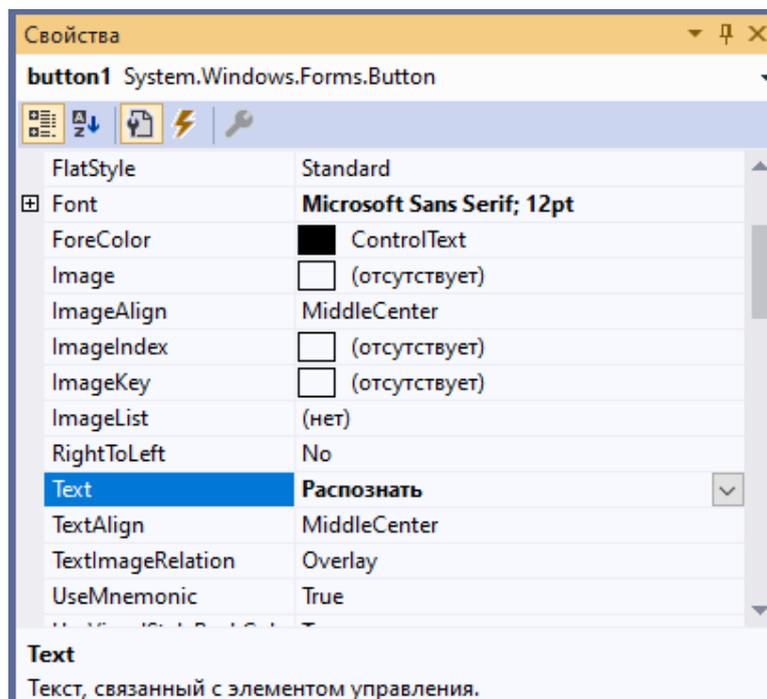


Рисунок 4 – Свойства кнопки

У компонента pictureBox достаточно поменять размер, а в свойствах формы значение поля FormBorderStyle изменим на FixedSingle (рис. 5).

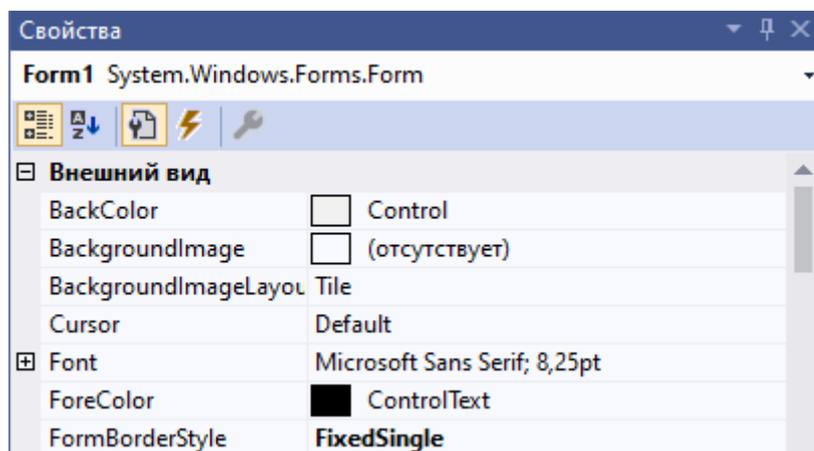


Рисунок 5 – Свойства формы

Также добавим элемент openFileDialog, который понадобится для загрузки изображений. В его свойствах укажем формат загружаемых фотографий JPG и PNG в поле Filter (рис. 6).

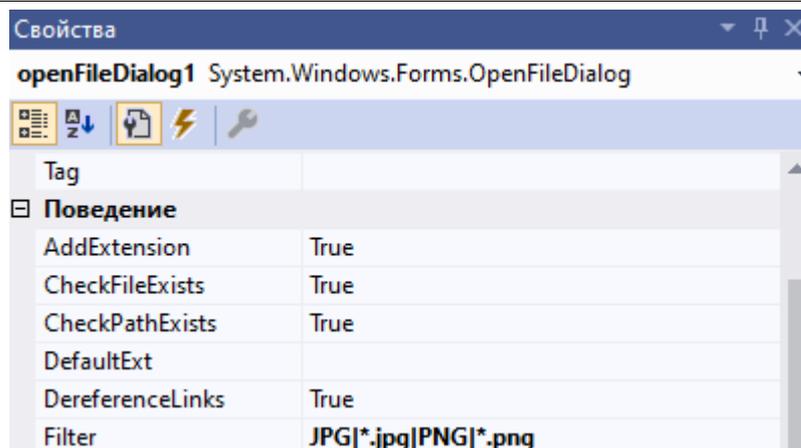


Рисунок 6 – Свойства openFileDialog

Теперь подключим необходимую библиотеку Emgu CV, которая служит для распознавания различных объектов. Перейдем на вкладку обозреватель решений, кликнем правой кнопкой мыши по нашему проекту и перейдем на управление проектами NuGet. Там необходимо ввести в поиске библиотеку Emgu CV и нажать кнопку установить (рис. 7).

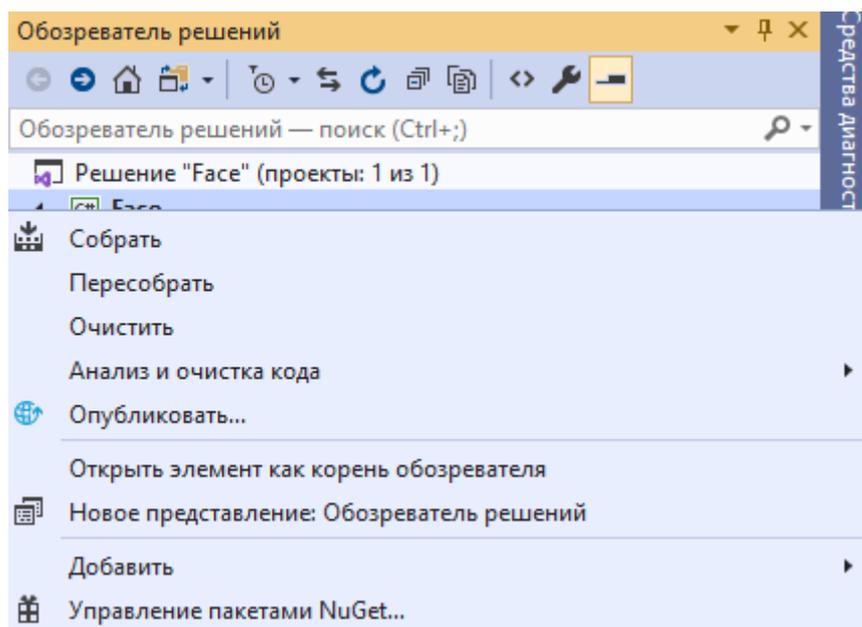


Рисунок 7– Скачивание библиотеки Emgu CV

Далее перейдем на страницу редактирования кода, дважды щелкнув на главную форму. Здесь нам необходимо подключить скачанную библиотеку двумя строчками (рис. 8). Также подключим xml-файл с исходными данными для распознавания лиц на изображении, который можно скачать на Github [7]. Но перед этим необходимо перенести xml-файл прямо в проект, чтобы он отображался в списке рядом с элементами.

```

1  using System;
2  using System.Collections.Generic;
3  using System.ComponentModel;
4  using System.Data;
5  using System.Drawing;
6  using System.Linq;
7  using System.Text;
8  using System.Threading.Tasks;
9  using System.Windows.Forms;
10 using Emgu.CV;
11 using Emgu.CV.Structure;
12
13 namespace Face
14 {
15     public partial class Form1 : Form
16     {
17         private static CascadeClassifier classifier = new CascadeClassifier("haarcascade_frontalface_alt_tree.xml");
18         public Form1()
19         {
20             InitializeComponent();
21         }
22     }
23 }

```

Рисунок 8 – Окно редактирования кода формы

Затем напишем обработчик нажатия кнопки. Для этого дважды щелкнем на кнопку ЛКМ и внутри пропишем следующим код (рис. 9).

```

40 private void button1_Click(object sender, EventArgs e)
41 {
42     try
43     {
44         DialogResult res = openFileDialog1.ShowDialog();
45         if (res == DialogResult.OK)
46         {
47             string path = openFileDialog1.FileName;
48
49             pictureBox1.Image = Image.FromFile(path);
50
51             Bitmap bitmap = new Bitmap(pictureBox1.Image);
52
53             Image<Bgr, byte> grayImage = new Image<Bgr, byte>(bitmap);
54
55             Rectangle[] faces = classifier.DetectMultiScale(grayImage, 1.1, 0);
56
57             foreach (Rectangle face in faces)
58             {
59                 using (Graphics graphics = Graphics.FromImage(bitmap))
60                 {
61                     using (Pen pen = new Pen(Color.Red, 2))
62                     {
63                         graphics.DrawRectangle(pen, face);
64                     }
65                 }
66             }
67             pictureBox1.Image = bitmap;
68         }
69     }
70     else
71     {
72         MessageBox.Show("Изображение не выбрано", "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);
73     }
74 }
75 catch (Exception ex)
76 {
77     MessageBox.Show(ex.Message, "Ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);
78 }
79 }
80
81 }

```

Рисунок 9 – Обработчик нажатия кнопки

По нажатию на кнопке открывается окно загрузки файлов (44 строчка), а если возникли ошибки, сообщаем пользователю о них (70-77 строчки). При

успешном открытии окна изображение загружается в pictureBox (47-49 строчки). Далее прописываются необходимые параметры для работы библиотеки Emgu CV (51-53 строчки). Затем в массив заносится список прямоугольников, обрисовывающих лица (55 строчка). И наконец, эти прямоугольники рисуем методом Graphics, перебирая массив циклом foreach (57-66 строчки).

Запустим приложение и загрузим пару фотографии для проверки (рис. 10).



Рисунок 10 – Проверка приложения

На фотографии лицо расположено внутри красного прямоугольника, что говорит о правильности определения приложением лица человека. Загрузим изображение с парой людей для проведения еще одного теста (рис. 11).

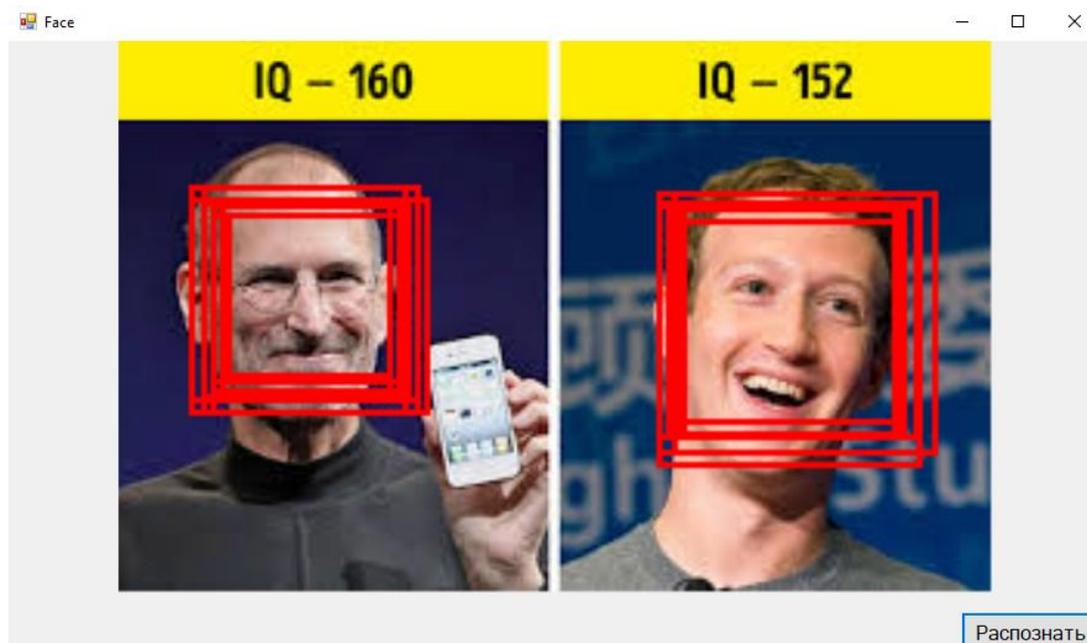


Рисунок 11 – Распознавание нескольких лиц

Распознавание нескольких лиц прошло успешно из чего можно сделать вывод, что приложение может справиться, как с фотографиями, где расплoжен один человек, так и с изображениями, где засняты несколько людей.

3. Выводы

Технологии распознавания лиц уже тесно переплелись с нашей жизнью. Полиция расследует различные преступления, находя злоумышленника по камерам, компании берут на вооружение данные технологии для создания контента и получения прибыли, МЧС находят потерявшихся людей. Все это далеко не все области применения технологии распознавания лиц, которые в будущем будут только развиваться и приносить пользу обществу.

В данной статье были рассмотрены инструмент Microsoft Visual Studio и библиотека Emgu CV для создания приложения по распознаванию лиц с фотографий.

Библиографический список

1. Куликов А.А., Демкин Д.В., Мельков А.Е. Анализ влияния максимальной степени сжатия изображения лица на результат распознавания лица // Перспективы науки. 2014. №3. С. 104-108.
2. Небаба С.Г., Захарова А.А. Применение алгоритма формирования индивидуальной трехмерной модели человеческого лица в системе распознавания личности по изображению лица // Графикон 2016. Протвино: Автономная некоммерческая организация "Институт физико-технической информатики", 2016. С. 310-313.
3. Матюков Н., Степанова А., Федякова А.А., Гребнева Д.А., Нехаев И.Н. Распознавание лиц на изображении с помощью масок и выделение отдельных частей лица // Научному прогрессу – творчество молодых. 2017. №3. С. 85-87.
4. Идятов А.Р., Фоменко В.Д., Козько А.А. Распознавание лиц в задаче поиска человека по лицу на мероприятиях // Вестник современных исследований. 2018. №12.10. С. 190-194.
5. Wen Y. A discriminative feature learning approach for deep face recognition // European conference on computer vision. Springer, Cham, 2016. С. 499-515.
6. Microsoft Visual Studio // Википедия URL: https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio (дата обращения: 25.01.2020).
7. OpenCV // Github URL: <https://github.com/opencv/opencv/tree/master/data/haarcascades> (дата обращения: 25.01.2020).