

Санкт-Петербургский государственный университет
Направление подготовки «Прикладная математика - процессы управления»
Образовательная программа «Программирования и информационные
технологии»

Отчёт по практическим заданиям №3 и №4.
СОЗДАНИЕ ПРОГРАММЫ ДЛЯ МОДЕЛИРОВАНИЯ СИСТЕМ
РАСПОЗНАВАНИЯ ЛЮДЕЙ ПО ЛИЦАМ;
СОЗДАНИЕ ПРОГРАММЫ ДЛЯ МОДЕЛИРОВАНИЯ
ПАРАЛЛЕЛЬНОЙ СИСТЕМЫ РАСПОЗНАВАНИЯ ЛЮДЕЙ ПО
ЛИЦАМ
по курсу “Прикладные задачи построения современных
вычислительных систем”

Выполнил студент
группы 17.Б11-пу
Макоев Артур Аланович

Санкт-Петербург

2021

Содержание

1. Цель работы.....	3
2. Признаковые пространства.....	3
3. Задание №3.....	7
4. Задание №4.....	15
5. Заключение	19

1. Цель работы

Целью работы является реализация простого классификатора лиц и анализ точности распознавания в зависимости от используемых для классификации признаков.

Исходный код программы находится по адресу github.com/zarond/BiometricTasks. Программа написана на языке Python, для графического интерфейса использовалась библиотека Tkinter, для математических вычислений библиотеки Numpy, SciPy, для работы с изображениями использовались Pillow и OpenCV.

2. Признаковые пространства

В работе рассматриваются следующие признаковые пространства:

- Гистограмма яркости – полный диапазон значений яркости делится на N равных частей, и для каждого отрезка из диапазона яркостей считается количество пикселей со значениями яркости в этом диапазоне. Параметр – целое число N .
- DFT – дискретное преобразование Фурье. Параметр – целое число N , размер квадратной подматрицы из коэффициентов Фурье.
- DCT – дискретное косинусное преобразование. Параметр – целое число N , размер квадратной подматрицы из коэффициентов косинусного преобразования.
- Scale – в качестве признаков берется то же самое изображение, но в уменьшенном масштабе. Параметр S – вещественное число от 0.01 до 1.
- Градиент – считается функция вертикального градиента следующим образом: по оси Y рассматриваем прямую $y=a$ и два окна размером $[image_width, N]$, которые строятся сверху и снизу от прямой. Затем сравниваем два этих фрагмента изображения в окнах с помощью суммы разностей, при этом один из фрагментов зеркально отражается по оси Y .

На языке Python код выглядит так:

```
def gradient_feature(image:np.array, n = 2):
    size = image.shape[0]//n
    result = np.empty(size-1)
    for i in range(size-1):
        result[i] = np.sum(image[i*n:i*n+n,:]-
np.flip(image[i*n+n:i*n+2*n,:],axis=0))
    return result
```

Параметр – целое число N , высота окна.

Классификатор работает по принципу сравнения признаков искомого элемента с эталонными элементами с помощью Евклидова расстояния, классифицируя искомый элемент как класс самого близкого к нему эталона.

Примеры признаковых пространств с различными параметрами:

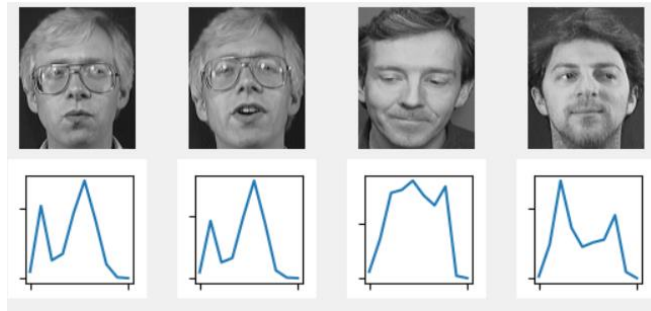


Рис. 1 Гистограмма, $N = 10$

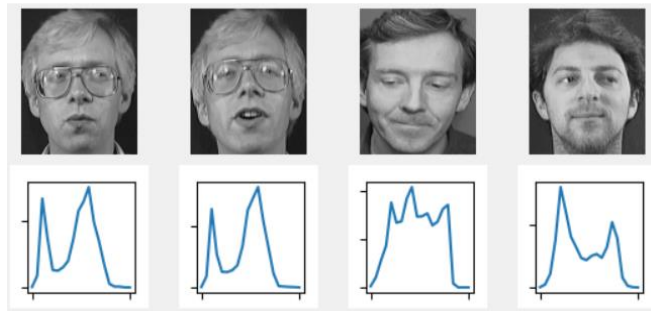


Рис. 2 Гистограмма, $N = 20$

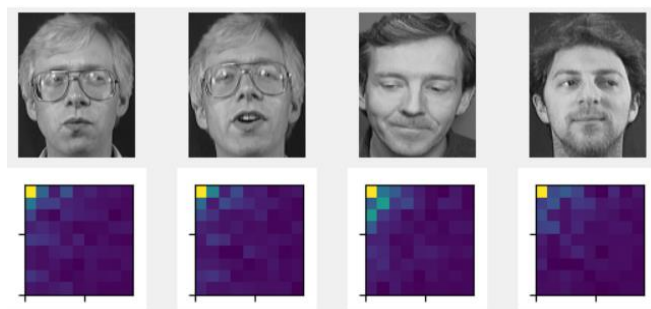


Рис. 3 DFT, $N = 10$

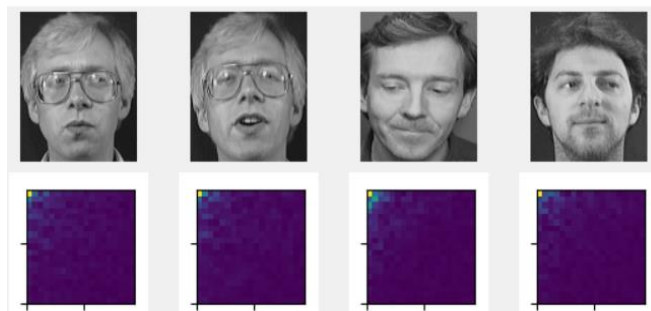


Рис. 4 DFT, $N = 20$

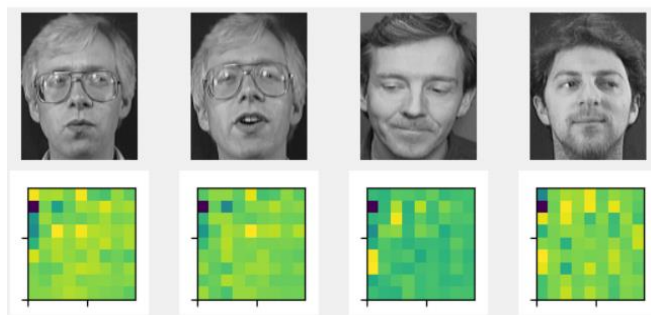


Рис. 5 DCT, $N = 10$

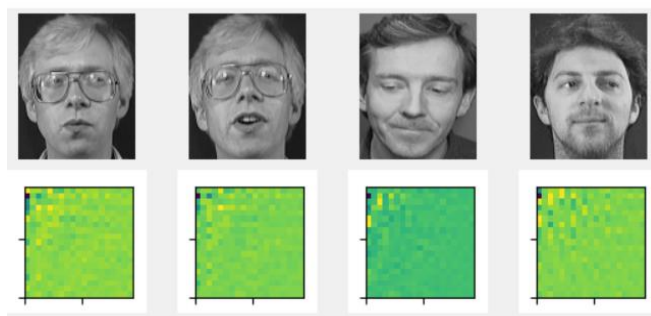


Рис. 6 DCT, $N = 20$



Рис. 7 Scale, $s = 0.5$



Рис. 8 Scale, $s = 0.09$

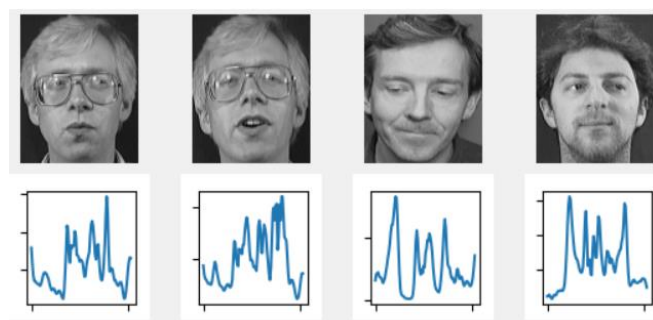


Рис. 9 Gradient, $N = 4$

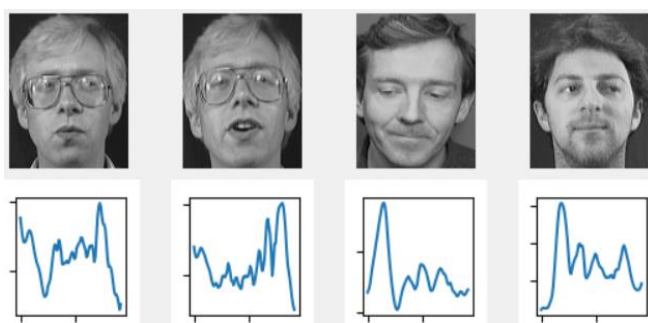


Рис. 10 Gradient, $N = 10$

Для сравнения признаков использовались четыре изображения, два из которых принадлежат одному человеку.

Признаки в виде гистограммы выглядят многообещающе, потому что даже невооруженным глазом видна схожесть между изображениями одного класса и различие с другими. При условии одинакового освещения в пределах одного класса, повороты головы не сильно меняют яркость пикселей, а только их положение.

Дискретное преобразование Фурье с изменением параметра начинает восстанавливать исходное изображение. Обрезание коэффициентов аналогично применению фильтра Гаусса к изображению. Вид признаков дискретного косинусного преобразования отличается от Фурье тем, что коэффициенты меньше отличаются друг от друга по абсолютному значению.

Метод масштабирования уменьшает различие изображений в мелких деталях, оставляя информацию о крупных пятнах света и тени, то есть кодируя форму лица с разной степенью детализации. Этот метод является потенциально дающим высокую точность распознавания.

Метод градиентов слабо зависит от поворота головы из стороны в сторону ввиду особенностей подсчета, но в полученных примерах признаки двух изображений из одного класса мало похожи друг на друга, что позволяет усомниться в его эффективности. В отличие от других методов, в которых параметр изменял количество учитываемых деталей изображения, влияние увеличения параметра в методе градиентов на изображение не так очевидно.

3. Задание №3

В программе используется база данных лиц ORL Face DataBase состоящая из 40 классов с 10 элементами в каждом. Перед импортом базы данных можно настроить масштабирование изображений.

Далее настраивается диапазон для количества тренировочных изображений в классе. Нижнее значение диапазона используется для построения финального графика, а верхнее значение указывает сколько изображений из класса идет в тренировочную, а сколько в тестовую выборку.

Затем, используя кросс-валидацию на тренировочной выборке, ищем самое оптимальное значение параметра для преобразования в признаковое пространство. Поиск проводится на трех разных настройках кросс-валидации, для каждой настройки строится график зависимости точности распознавания от параметра. На полученном значении параметра с максимальной точностью распознавания уже проводим исследование на тестовой выборке.

Кол-во тренировочных изображений	6	7	8	9
Кол-во фолдов для кросс-валидации	[2,3,6]	[2,3,7]	[2,4,8]	[3,4,9]

Строим график зависимости между количеством эталонных изображений в классе и точностью распознавания.

Пользовательский интерфейс устроен следующим образом:

- Выбирается база данных и параметр масштабирования импортируемых изображений. Нажимается кнопка «Load data».
- Выбирается число тренировочных изображений в классе.
- Выбирается метод, нажимается кнопка «Confirm».
- Показ примеров признаков с задаваемым параметром с помощью кнопки «Show feature examples».
- Кнопка «Compute» для вычисления параметра для классификатора с помощью кросс-валидации.
- После завершения вычислений отобразится информация о точности классификатора и о найденном параметре, построятся графики.
- Кнопка «auto mode» включает режим распознавания, наглядно показывающий изображения из тестового набора, которые сопоставляются изображениям из найденного классификатором класса. Если классификатор сработал верно, то рамка изображения – зеленого цвета, в ином случае – красного.
- Кнопка «Find one face» – аналогично предыдущему, только с одним лицом за раз.
- Кнопки «Test and graph» и «Test and graph realtime» строят графики с точностью распознавания классификатора по мере последовательного распознавания изображений из тестового набора.
- Кнопки «Save results» и «Load results» предназначены чтобы сохранять и загружать результаты вычислений и состояние программы, в том числе множества тестовых и тренировочных изображений.

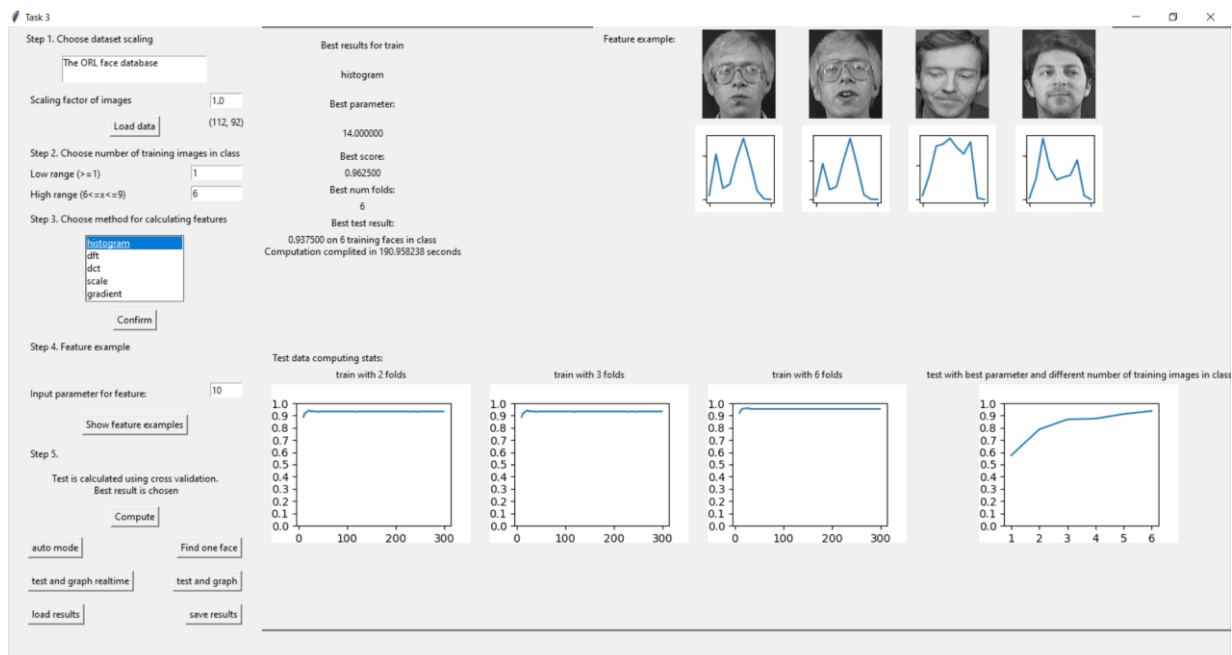


Рис. 11 Интерфейс программы.

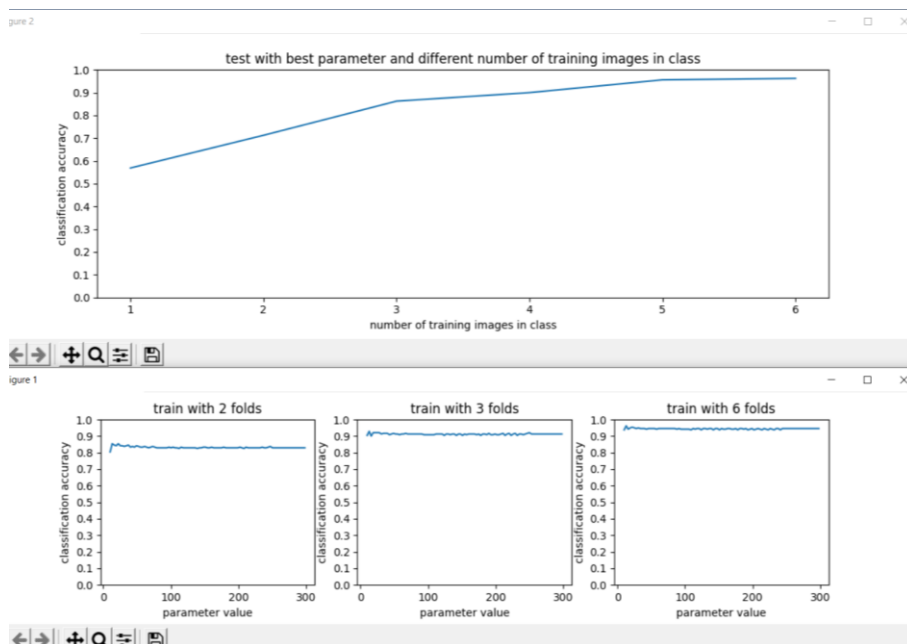


Рис. 12 Построенные графики - вывод программы

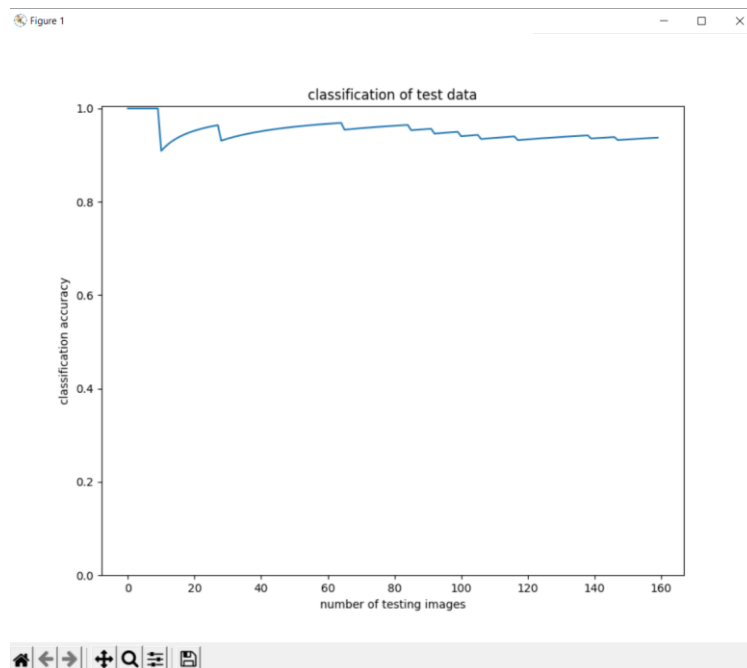


Рис. 13 Тестирование классификатора на тестовом наборе изображений.

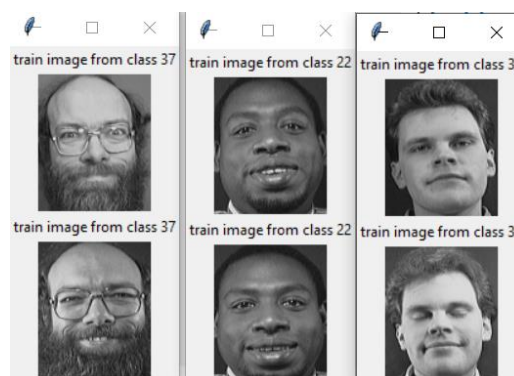


Рис. 14 Поиск одного лица за раз

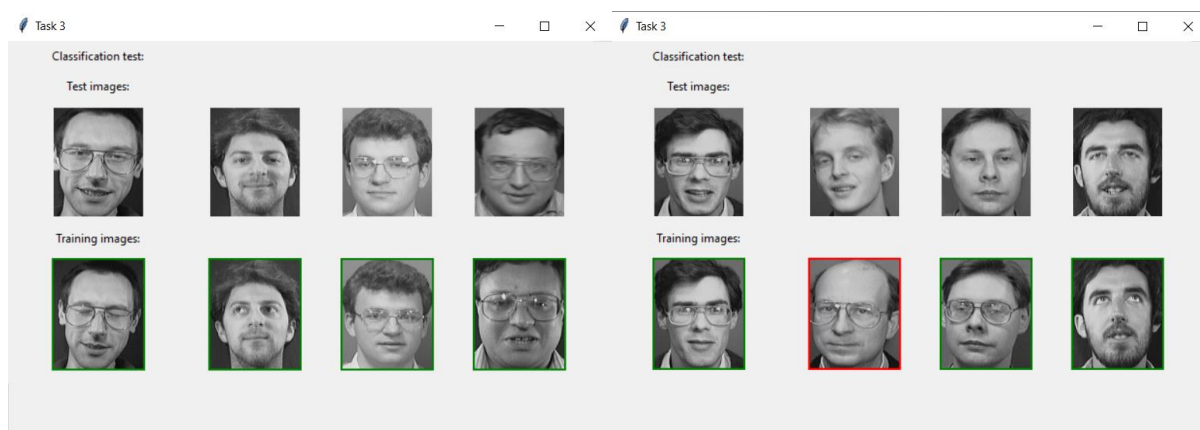


Рис. 15 Работа режима "auto mode"

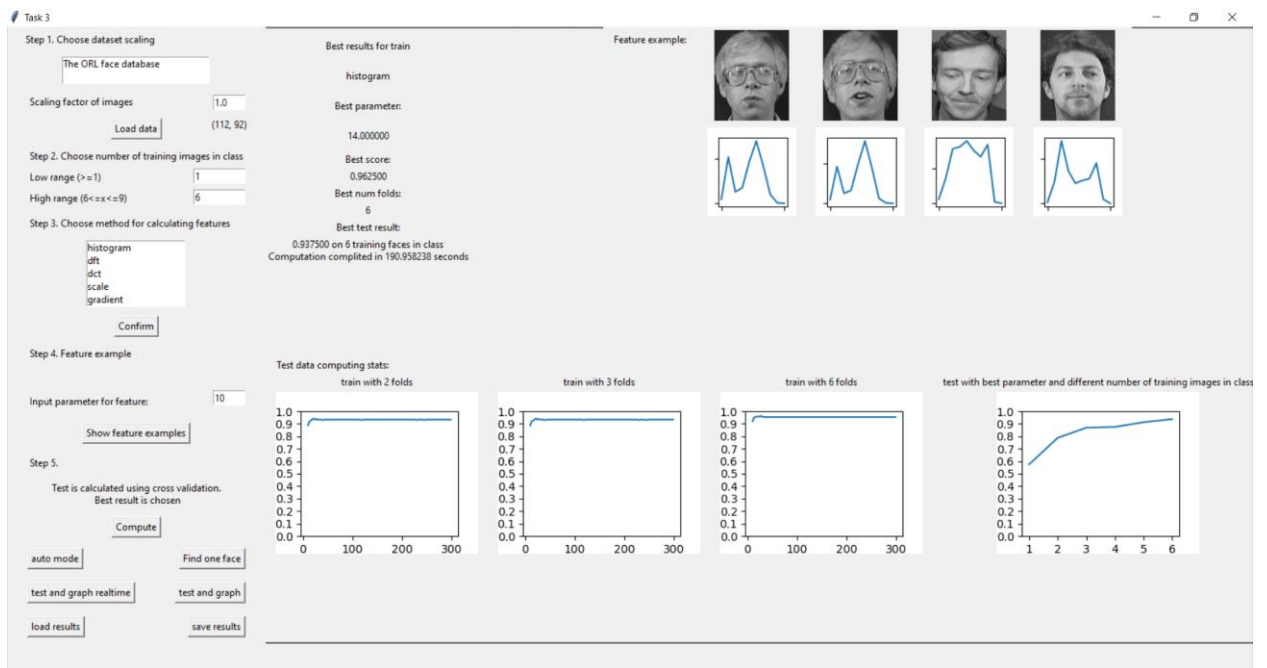


Рис. 16 Признак – гистограмма

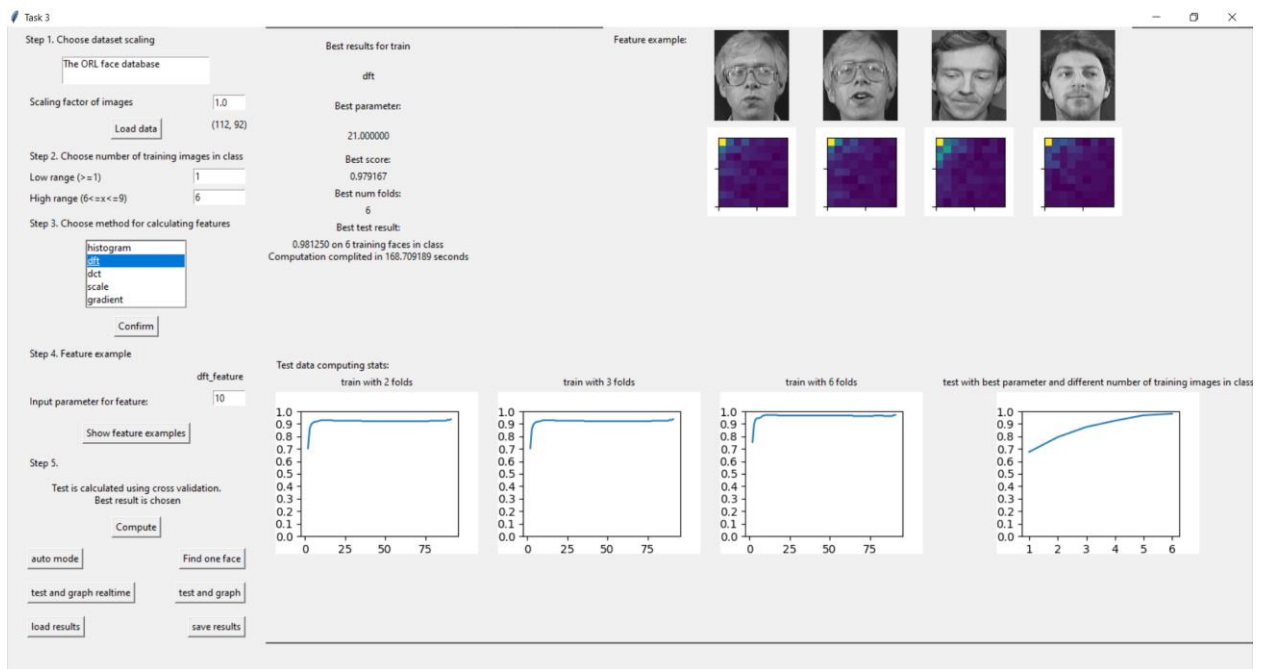


Рис. 17 Признак - коэффициенты Фурье образа

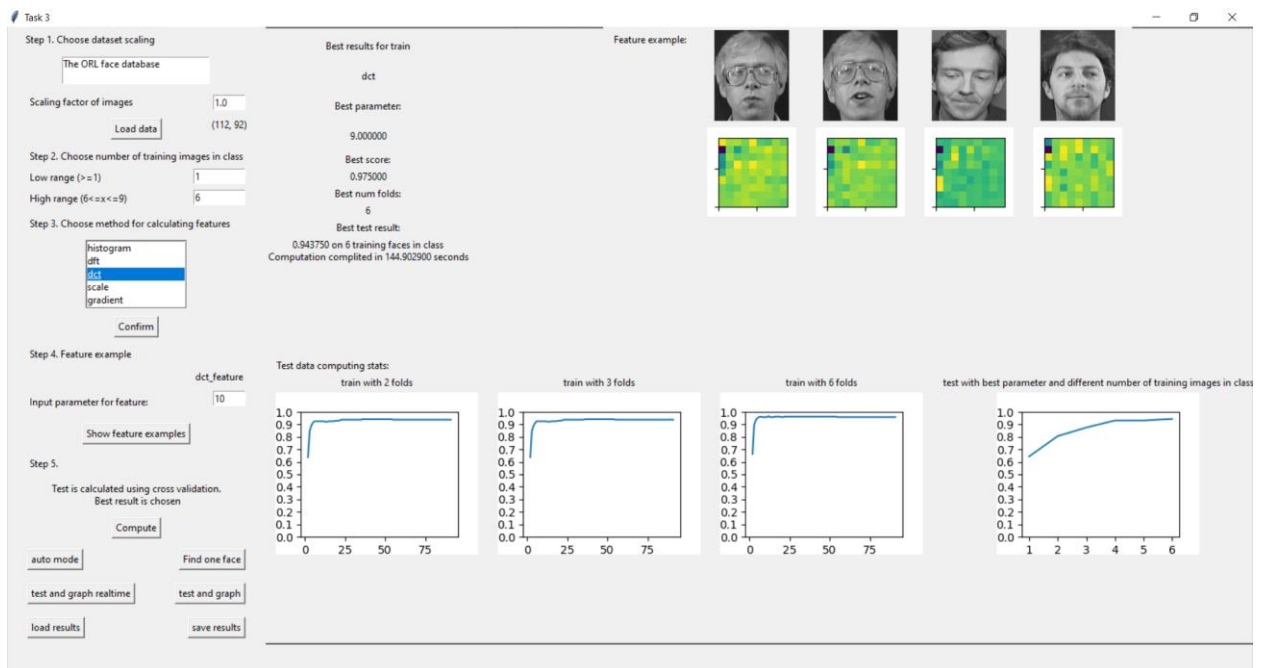


Рис. 18 Признак - косинусное преобразование

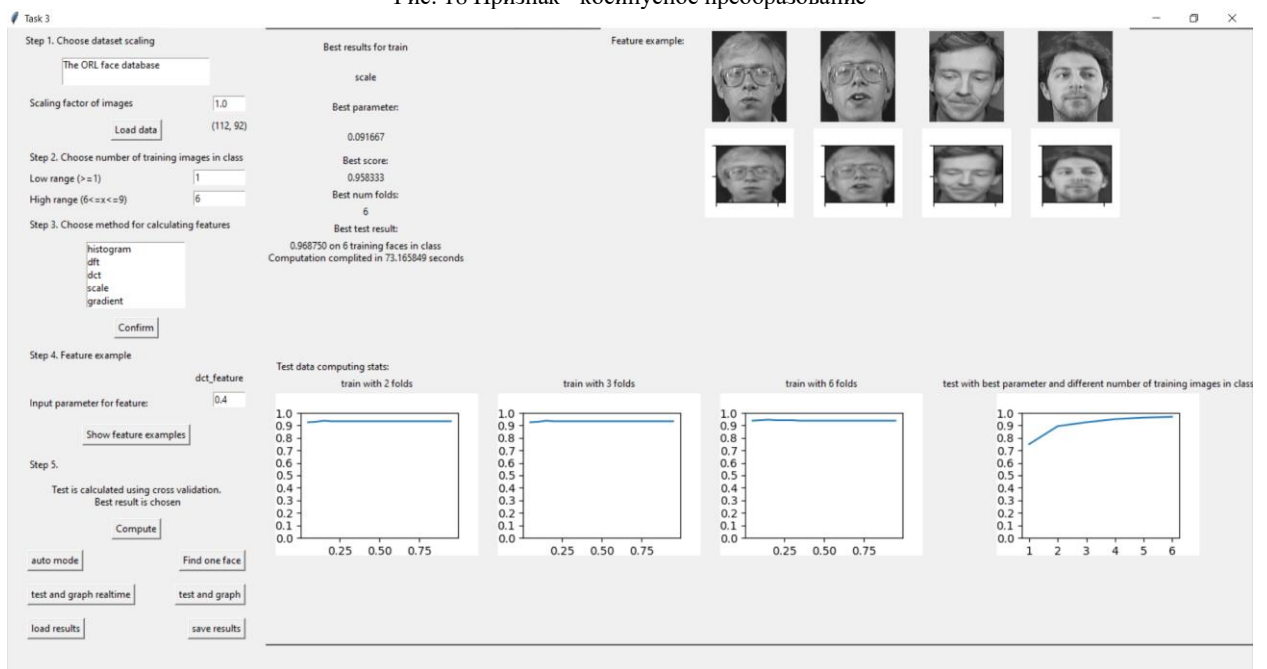


Рис. 19 Признак - масштабированное изображение

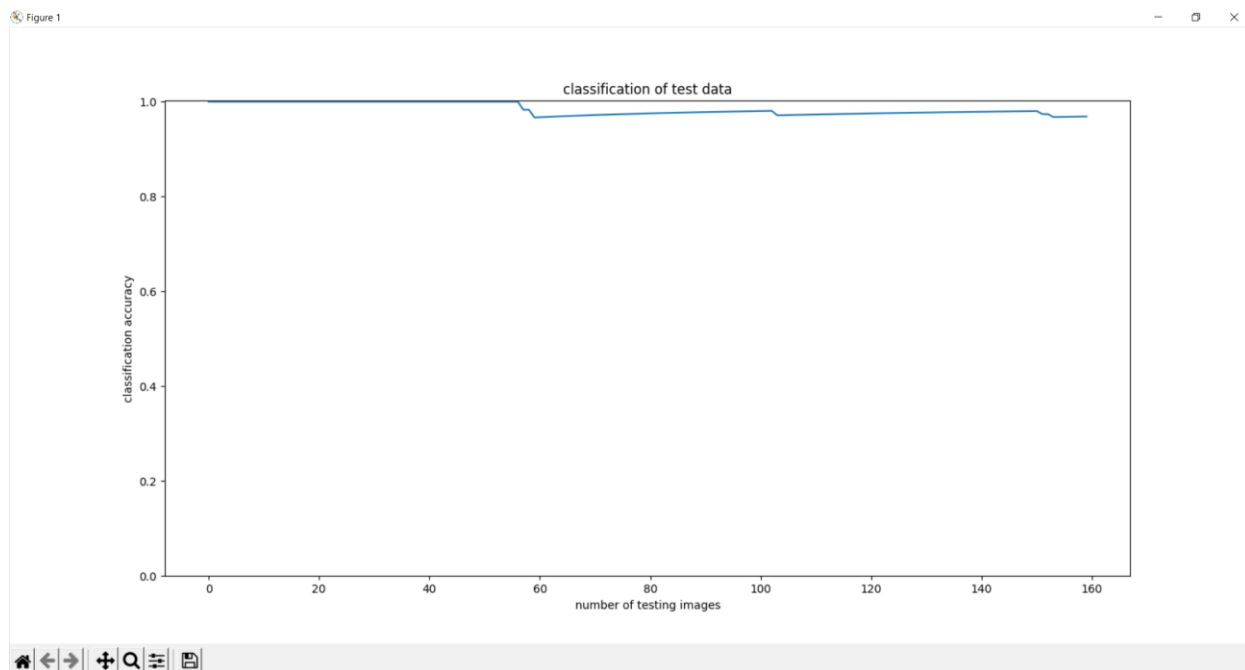


Рис. 20 Классификация с помощью признака – масштабированного изображения

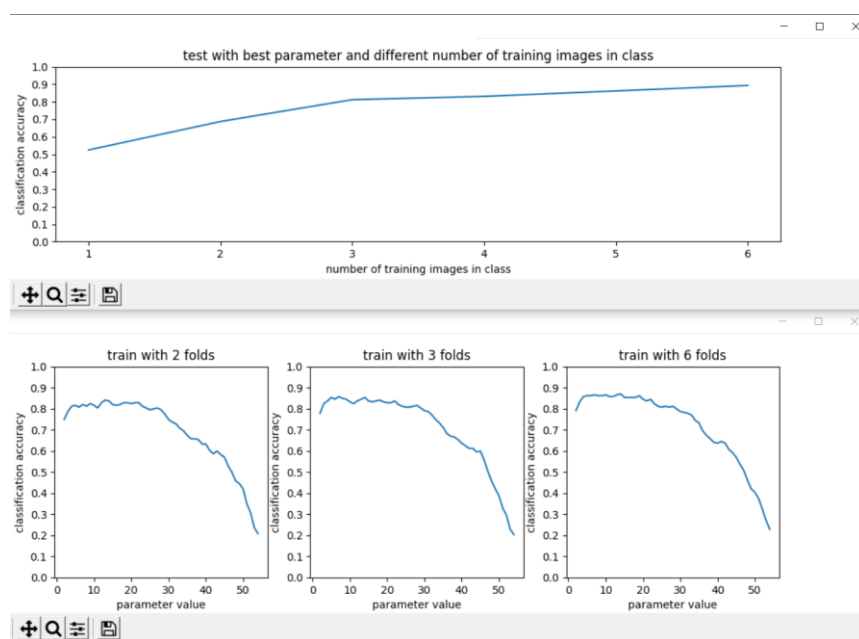


Рис. 21 Результаты обучения классификатора с признаком - градиентом



Рис. 22 Классификация с градиентом

Остальные данные о работе программы приведу в виде таблицы:

Точность распознавания:

	Число тренировочных изображений в классе			
Признак:	6	7	8	9
histogram	0.938	0.967	0.987	0.975
DFT	0.981	0.975	0.975	0.975
DCT	0.944	0.967	0.987	1.000
Scale	0.969	0.950	1.000	1.000
Gradient	0.900	0.917	0.937	0.975

Параметр признака:

	Число тренировочных изображений в классе			
Признак:	6	7	8	9
histogram	14	13	16	12
DFT	21	29	13	4
DCT	9	6	7	9
Scale	0.091	0.164	0.087	0.094
Gradient	11	6	14	13

Время работы алгоритма обучения (в секундах):

	Число тренировочных изображений в классе			
Признак:	6	7	8	9
histogram	191	255	333	417
DFT	169	219	298	377
DCT	144	196	263	331
Scale	22	31	40	52
Gradient	150	196	267	326

Из полученных результатов видно, что точность классификатора выше 0.9 и близка к единице. Все методы выделения признаков, кроме градиента, показывают высокий результат.

В качестве эксперимента провел вычисления точности классификатора для изображений, которые были масштабированы в два раза по обеим осям (scaling factor = 0.5) при импорте:

Точность распознавания:

	Число тренировочных изображений в классе	
Признак:	6	8
histogram	0.981	1.000
DFT	0.962	0.950
DCT	0.950	0.987
Scale	0.943	0.962
Gradient	0.868	0.900

Параметр признака:

	Число тренировочных изображений в классе	
Признак:	6	8
histogram	62	30
DFT	10	15
DCT	7	16
Scale	0.158	0.162
Gradient	7	8

Время работы алгоритма обучения (в секундах):

	Число тренировочных изображений в классе	
Признак:	6	8
histogram	120	217
DFT	46	84
DCT	52	84
Scale	16	29
Gradient	41	71

4. Задание №4

Программа Task4 реализует классификатор по принципу голосования на основе классификаторов по признакам из предыдущего задания. Результат классификации определяется как самый популярный ответ среди пяти классификаторов.

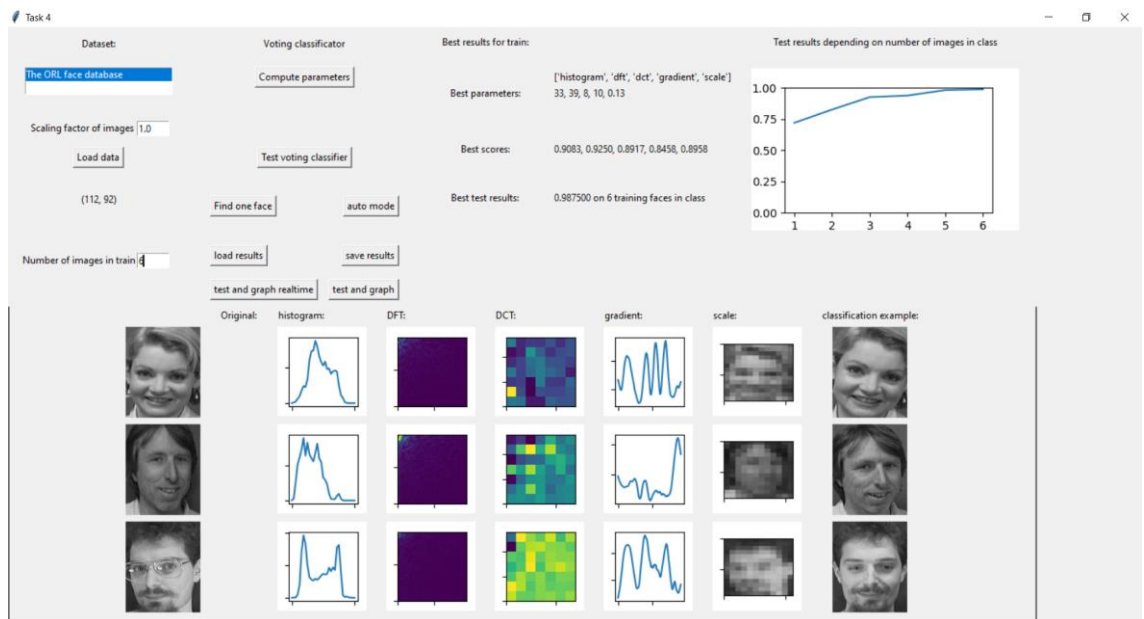


Рис. 23 Интерфейс программы.

Интерфейс программы схож с предыдущей, кнопка «Compute» заменена на 2 кнопки – «Compute parameters» и «Test voting classifier». Первая кнопка запускает поиск наилучших параметров для отдельных классификаторов, вторая высчитывает точность классификатора голосования на тестовой выборке и строит график точности распознавания в зависимости от числа тренировочных изображений в классе.

Кнопка «auto mode» запускает режим распознавания, в котором показаны результаты распознавания классификаторами одного изображения по отдельности и вместе.

Кнопки «Test and graph» строят графики точности распознавания всех классификаторов при последовательном распознавании тестовой выборки.

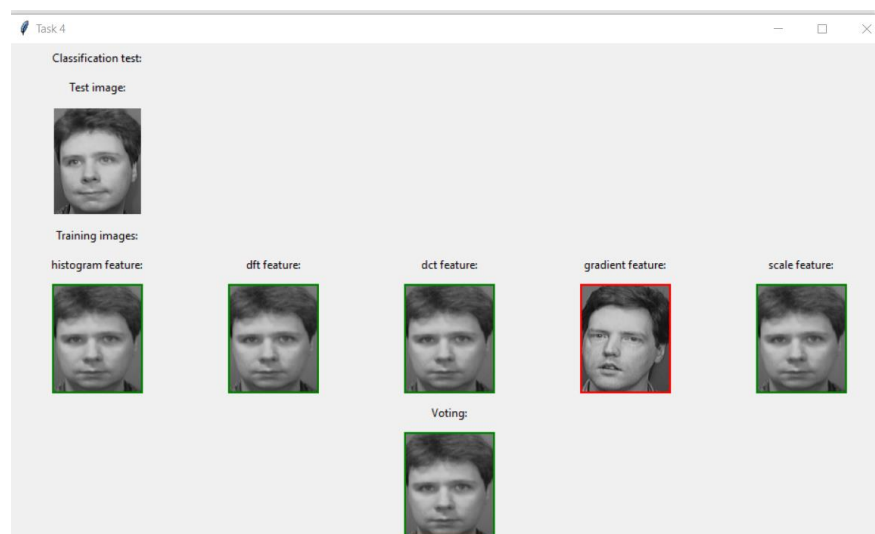


Рис. 24 Режим "auto mode"

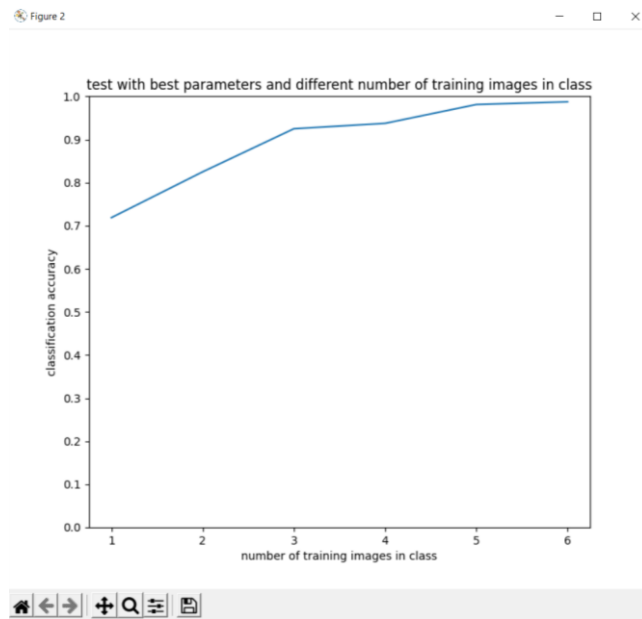


Рис. 25 Точность распознавания в зависимости от количества тренировочных изображений в классе

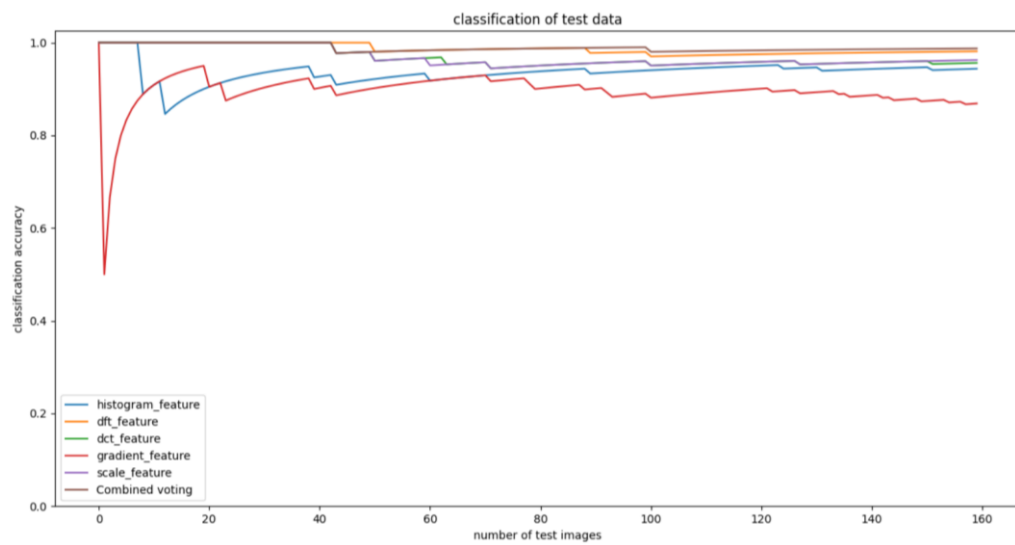


Рис. 26 Точность классификации на тестовой выборке по типам классификаторов для 6 изображений в классе

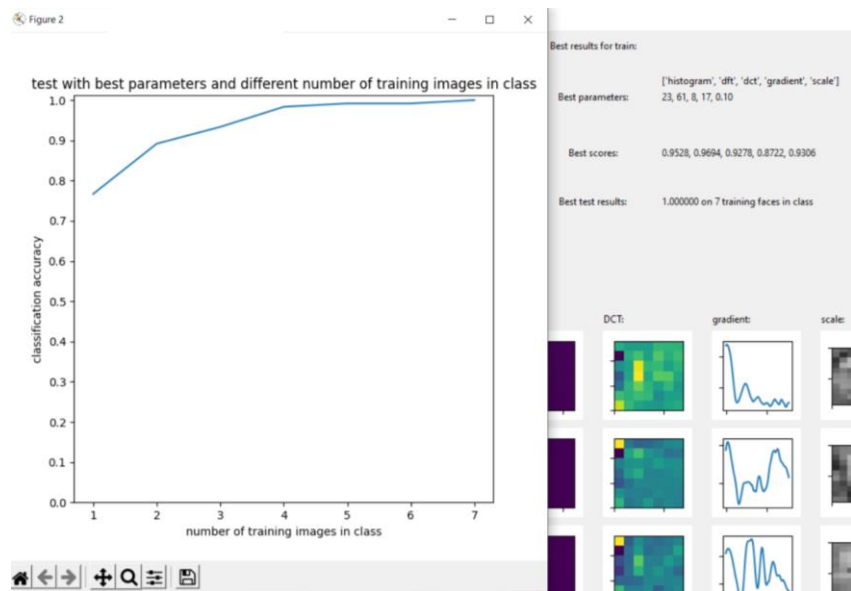


Рис. 27 Результаты работы для 7 изображений в классе

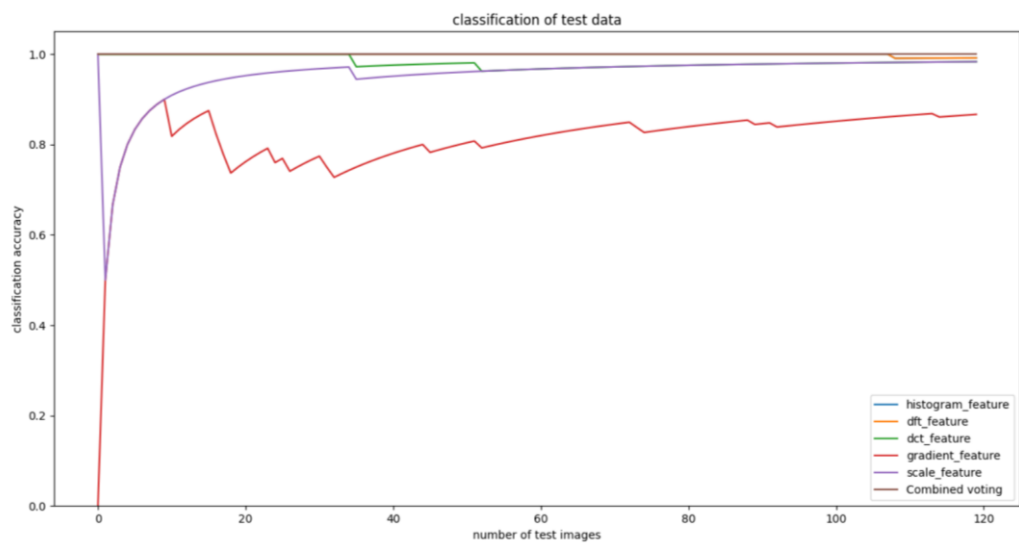


Рис. 28 Точность классификации для 7 изображений в классе

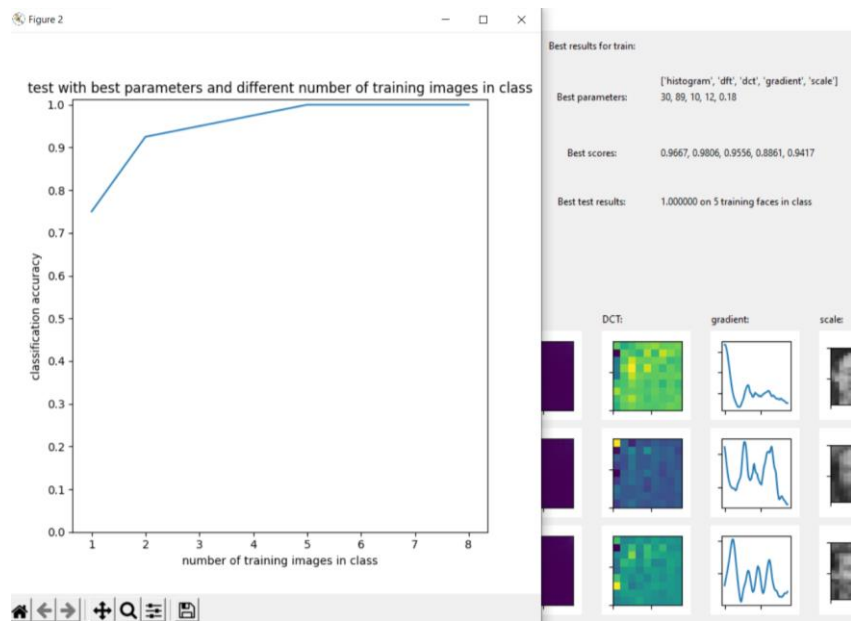


Рис. 29 Результаты для 8 изображений в классе

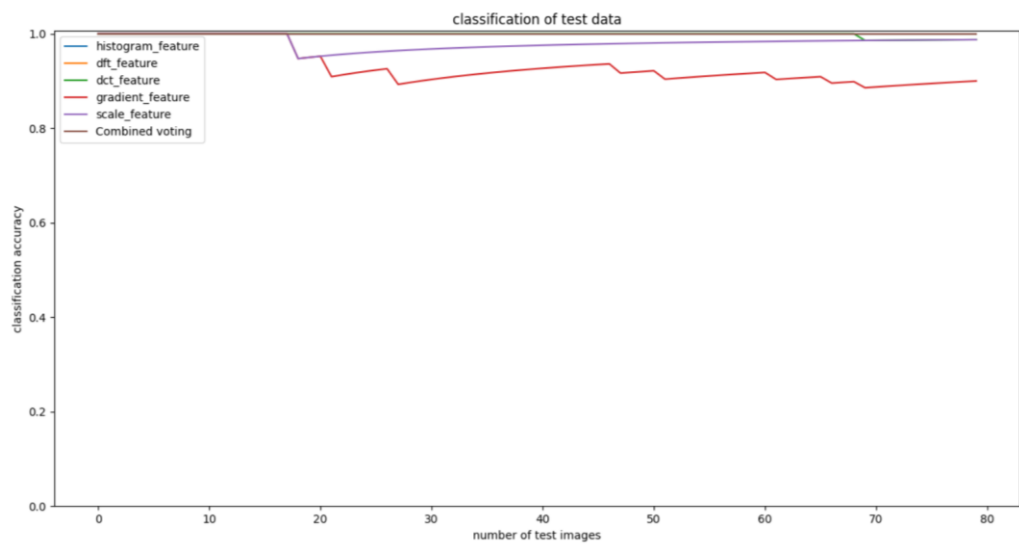


Рис. 30 Точность классификации для 8 изображений в классе

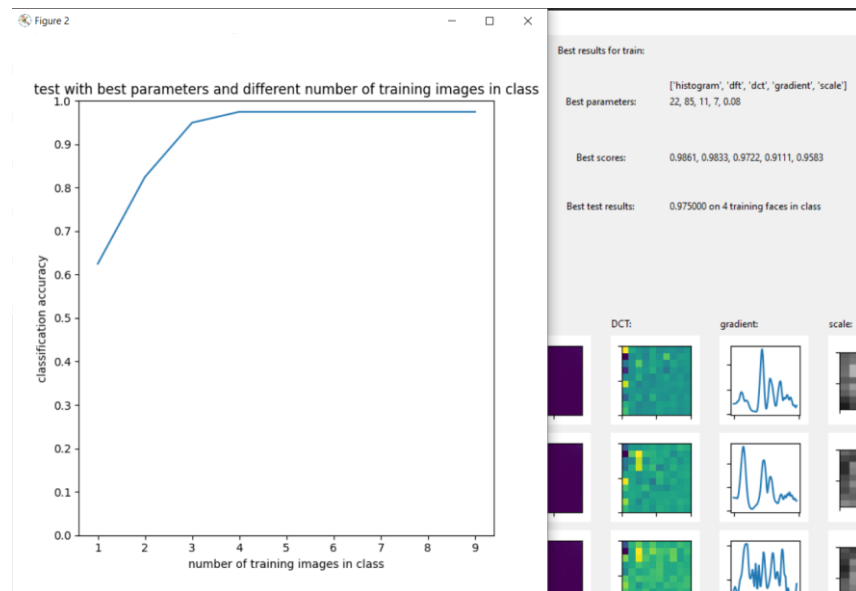


Рис. 31 Результаты для 9 изображений в классе

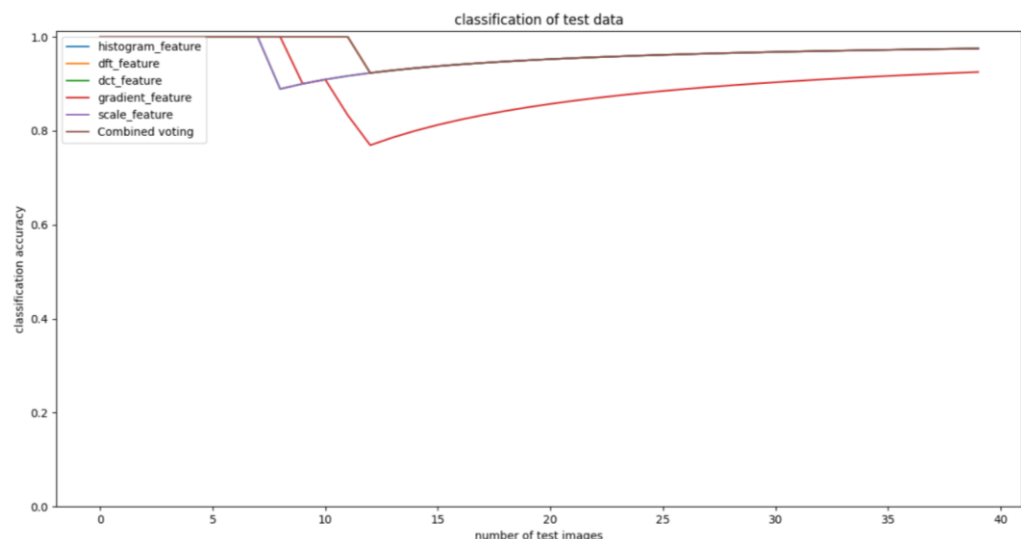


Рис. 32 Точность классификации для 9 изображений в классе

Результаты экспериментов в виде таблицы:

Кол-во тренировочных изображений в классе	Параметры классификаторов	Точность распознавания
6	33,39,8,10,0.17	0.987
7	23,61,8,17,0.10	1.000
8	30,89,10,12,0.18	1.000
9	22,85,11,7,0.08	0.975

5. Заключение

Мною было изучено применение различных методов выделения признаков в задаче классификации лиц и написана программа, реализующая классификацию. С помощью кросс-валидации были подобраны наилучшие параметры, для которых точность распознавания выше 90% для всех типов признаков. Был создан классификатор голосования, который дает точность выше, чем входящие в него классификаторы по отдельности. Точность классификатора голосования близка к единице.