|  |
| --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  федеральное государственное автономное образовательное учреждение высшего образования  «САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ» |

КАФЕДРА №22 РАДИОТЕХНИЧЕСКИХ СИСТЕМ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ОТЧЁТ ПО ПРАКТИКЕ  ЗАЩИЩЁН С ОЦЕНКОЙ  Руководитель |  | | | |
| Преподаватель |  |  |  | А.К. Ермаков |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

ОТЧЁТ ПО ПРАКТИКЕ

|  |  |  |
| --- | --- | --- |
| вид практики | учебная | |
| тип практики | Практика по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности | |
| на тему индивидуального задания | | Умение работать в MATLAB |
|  | | | |
|  | | | |

|  |  |
| --- | --- |
| выполнен | Швечко Алиной Ивановной |
| фамилия, имя, отчество обучающегося в творительном падеже | |

|  |  |  |  |
| --- | --- | --- | --- |
| по направлению подготовки | 11.03.01 |  | «Радиотехника» |
|  | код |  | наименование направления |
|  | | | |
| наименование направления | | | |
| направленности | 01 |  | «Радиотехнические средства передачи, |
|  | код |  | наименование направленности |
| приёма и обработки сигналов» | | | |
| наименование направленности | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Обучающийся группы № | 2221 |  |  |  | А.И. Швечко |
|  | номер |  | подпись, дата |  | инициалы, фамилия |

Санкт–Петербург 2023**Содержание**

1 ЗАДАНИЕ 13

1.1 Словесное описание алгоритма3

1.2 Программа и её описание3

1.3 Результат работы программы4

2 Задание 25

2.1 Словесное описание алгоритма5

2.2 Программа и её описание6

2.3 Результат работы программы8

ВЫВОД9

БИБЛИОГРАФИЯ10

Цель работы: исследование интерактивной среды для разработки размещения приложения и программирования его поведения Matlab App Designer.

**1 Задание:** разработать интерфейс для 1 задания практики № 2. В интерфейсе должен быть реализован выбор минимум трех функций для анимации.

* 1. **Словесное описание алгоритма**

1. выбор трёх и более функций для созданий интерфейса;
2. с помощью программы Matlab App Designer написать код, в котором может содержаться:
3. графическое окно, в котором будет идти анимация функций;
4. кнопки, при помощи которых будет идти запуск или остановка анимации;
5. окно для ввода значения, которое автоматические будет подставлено в функцию;
6. перечень функций с выбором одной из них для вывода на график;
7. окно с результатом.

**1.2 Программа и её описание**

x = 0:0.01:app.xEditField.Value; - задаём шаг;

for n = 1:length(x); - создаём цикл по элементам вектора x;

pause(0.04); - делаем паузу;

if app.sinxnxnButton.Value == 1; - создаём условие;

s\_1 = sin(x.∙n).∙(x.∙n); - вводим матрицу с формулой;

plot(app.UIAxes, x, s\_1) – построение графика;

app.EditField.Value = («График построен»); - по окончании построение графика, выводим надпись;

end – конец условия;

if app.expxn2expxn2Button.Value == 1; - создаём условие;

s\_2 = (exp(x+n)/2) + (exp(-x+n)/(-2)); - вводим матрицу с формулой;

plot(app.UIAxes, s\_2) – построение графика;

app.EditField.Value = («График построен») - по окончании построение графика, выводим надпись;

end – конец условия;

if app.expxnButton.Value == 1 - создаём условие;

s\_3 = exp(x.∙n) - вводим матрицу с формулой;

plot(app.UIAxes, x, s\_3) – построение графика;

app.EditField.Value = («График построен») - по окончании построение графика, выводим надпись;

end – конец условия;

end – конец цикла.

**1.3 Результат работы программы**

На рисунке 1 представлен результат работы алгоритма.

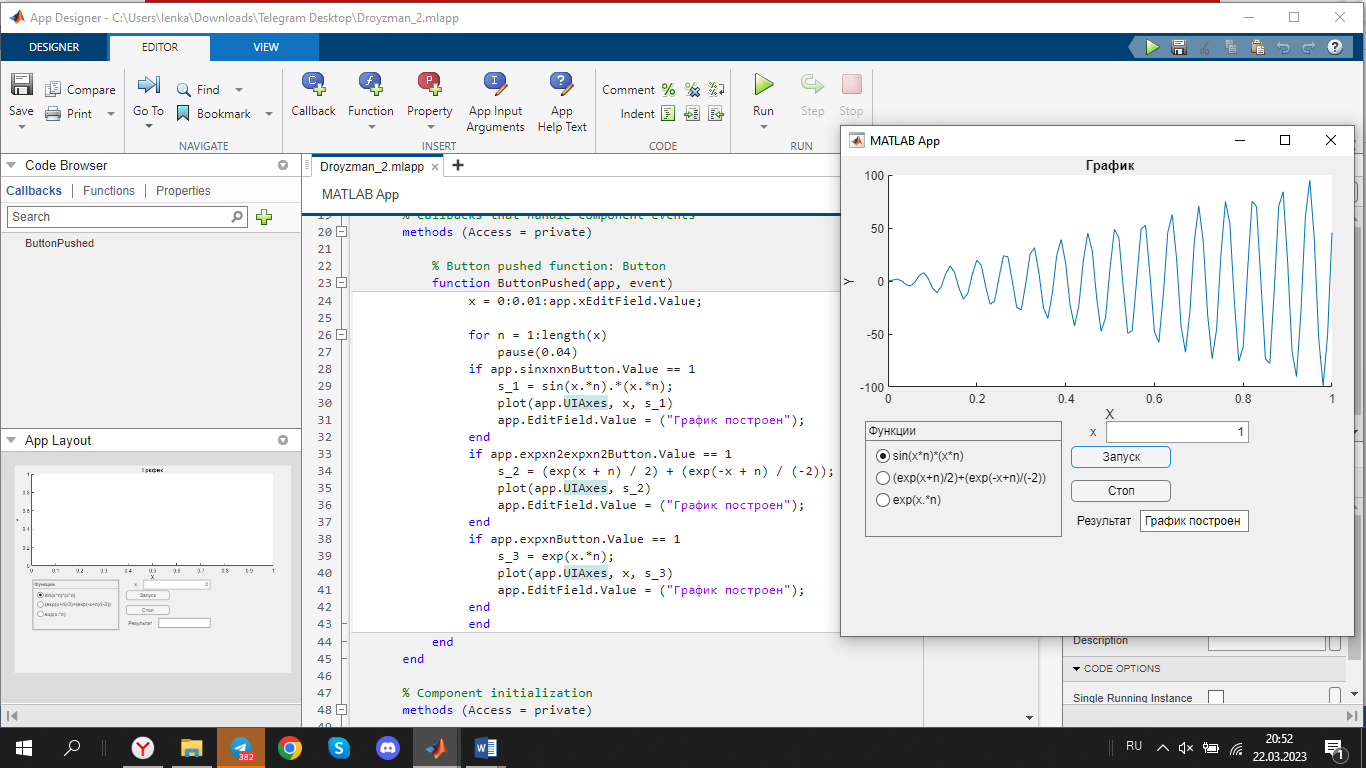


Рисунок 1 - результат работы программы, написанной для задания 1

1. **Задание 2**

Разработать интерфейс для 2 и 3 задания практики № 2. В интерфейсе должен быть реализован выбор файла из папки пользователем и выбор того что должна делать программа (сравнивать и выводить отличия или определять движение из анимации).

* 1. **Словесное описание алгоритма**

1. пользователю предлагается два метода сравнения изображений(кадров);
2. первый метод заключается в сравнении кадров двух отдельный файлов:

2.1) выбор из выплывающего меню метод сравнения двух изображений;

2.2) по нажатии кнопки «Загрузка первого изображения» вызывается функция, при выполнении которой пользователю предлагается выбор файла в проводнике нажатием левой кнопки мыши;

2.3) файл читается и сохраняется в соответствующие свойства класса app.;

2.4) после открытия файла, изображение выводится на первые оси программы;

2.5) по нажатии кнопки «Загрузка второго изображения» вызывается функция, аналогичная предыдущей;

2.6) после загрузки первого и второго изображение идёт нажатие кнопки «Сравнить» и вывод результата на третьи оси программы;

1. второй метод заключается в сравнении текущего и следующего кадров в формате .gif;

3.1) при выборе метода анимации пользователь нажимает кнопку «Загрузка первого изображения» и указывает путь к анимации;

3.2) файл читается и сохраняется в соответствующие свойства класс app.;

3.3) после открытия файла анимации выводится первый кадр на первые оси программы;

3.4) выбор второго кадра анимации ведётся аналогичным способом;

3.5) после загрузки кадров анимации идёт нажатие кнопки «Сравнить», при которой вызывается функция перебора всех кадров анимации (кроме последнего);

3.6) на каждый текущий кадр и последующий кадр переводится в цветовую шкалу RGB(red, green, blue) – идёт вычисление их разности;

3.7) разность последующего и текущего кадра переводится в оттенки серого – итог выводится на третьи оси программы.

**2.2 Программа и её описание**

а) Для Button\_1Pushed:

[fnp, pnm] = uigetfile(‘∙.jpg; ∙.png; ∙.gif’); - получаем строки с именем файла и папки;

if ishar (fnm) – проверка существования файла;

[app.Image\_1, app.Map\_1] = imread([pnm, fnm]); - читаем файл и записываем в свойства Image\_1 и Map\_1;

imshow(app.Image\_1(:, :, :, 1), app.Map\_1, ‘parent’, app.UIAxes\_1); - выводим изображение на оси UIAxes\_1;

end – конец цикла.

б) Для Button\_2Pushed:

[fnp, pnm] = uigetfile(‘∙.jpg; ∙.png; ∙.gif’); - получаем строки с именем файла и папки;

if ishar (fnm) – проверка существования файла;

[app.Image\_2, app.Map\_2] = imread([pnm, fnm]); - читаем файл и записываем в свойства Image\_2 и Map\_2;

imshow(app.Image\_2(:, :, :, 1), app.Map\_2, ‘parent’, app.UIAxes\_2); - выводим изображение на оси UIAxes\_2;

end – конец цикла.

в) Для Button\_3Pushed:

switch app.DropDown\_1.Value – выбор метода согласно свойству Value выпадающего меню DropDown\_1;

case 'Два изображения' – первый метод;

if isempty(app.Map\_1) – проверка наличия массива цветов Map\_1;

im1 = app.Image\_1 – присваиваем значение массива цветов RGB;

else – иначе;

im1 = ind2rgb(app.Image\_1(:, :, :, 1), app.Map\_1) – переводим в массив RGB;

end – конец цикла;

if isempty(app.Map\_2) – проверка наличия массива цветов Map\_2;

im2 = app.Image\_2 – присваиваем значение массива цветов RGB;

else – иначе;

im2 = ind2rgb(app.Image\_2(:, :, :, 1), app.Map\_2) – переводим в массив RGB;

end – конец цикла;

app.Image\_3 = im2 - im1 – вычисляем разность массивов цветов RGB;

app.Image\_3 = rgb2gray(app.Image\_3) – переводим в оттенки серого;

imshow(app.Image\_3, 'Parent', app.UIAxes\_3) – выводим изображение на оси UIAxes\_3;

case 'Анимация' – второй метод;

% Создаем усредненный кадр (второе изображение)

im0 = zeros(size(app.Image\_1));

im0 = im0(:, :, :, 1);

for i = 1:size(app.Image\_1, 4) – перебор кадров;

im0 = im0 + ind2rgb(app.Image\_1(:, :, :, i), app.Map\_1) – преобразование цветовой шкалы Matlab в RGB и добавление в сумму;

end – конец цикла;

im0 = im0 / i – вычисление среднего арифметического и округление до целых чисел;

imshow(im0, 'Parent', app.UIAxes\_2) – выводим изображение на оси UIAxes\_2;

for i = 1:size(app.Image\_1, 4) – перебор кадров;

im1 = ind2rgb(app.Image\_1(:, :, :, i), app.Map\_1) – преобразование цветовой шкалы Matlab в RGB;

app.Image\_3 = im1 - im0 – вычисляем разность массивов цветов RGB текущего и усредненного кадров;

app.Image\_3 = rgb2gray(app.Image\_3) – переводим в оттенки серого;

imshow(im1, 'Parent', app.UIAxes\_1) – выводим изображение на оси UIAxes\_1;

imshow(app.Image\_3, 'Parent', app.UIAxes\_3) – выводим изображение на оси UIAxes\_3;

drawnow – принудительная отрисовка фигур;

end – конец цикла;

otherwise

end – конец цикла.

* 1. **Результат работы программы**

В результате работы программы, внутри интерфейса выводится исходная анимация и выделенное движение на ней (Рисунок 2 – выделение движения) или же выводятся найденные отличия между двумя изображениями (Рисунок 3 – поиск отличий).

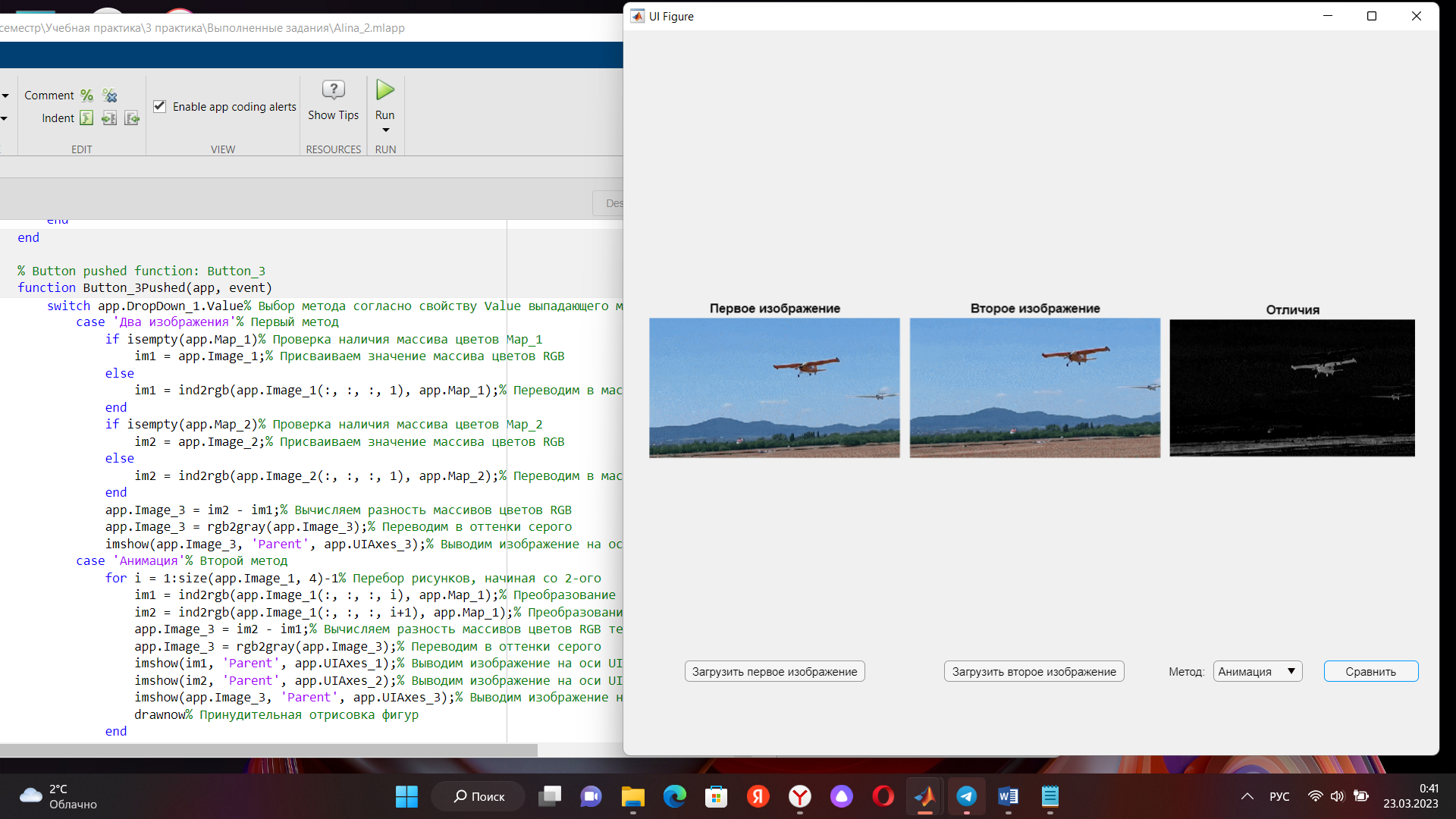


Рисунок 2 – выделение движения

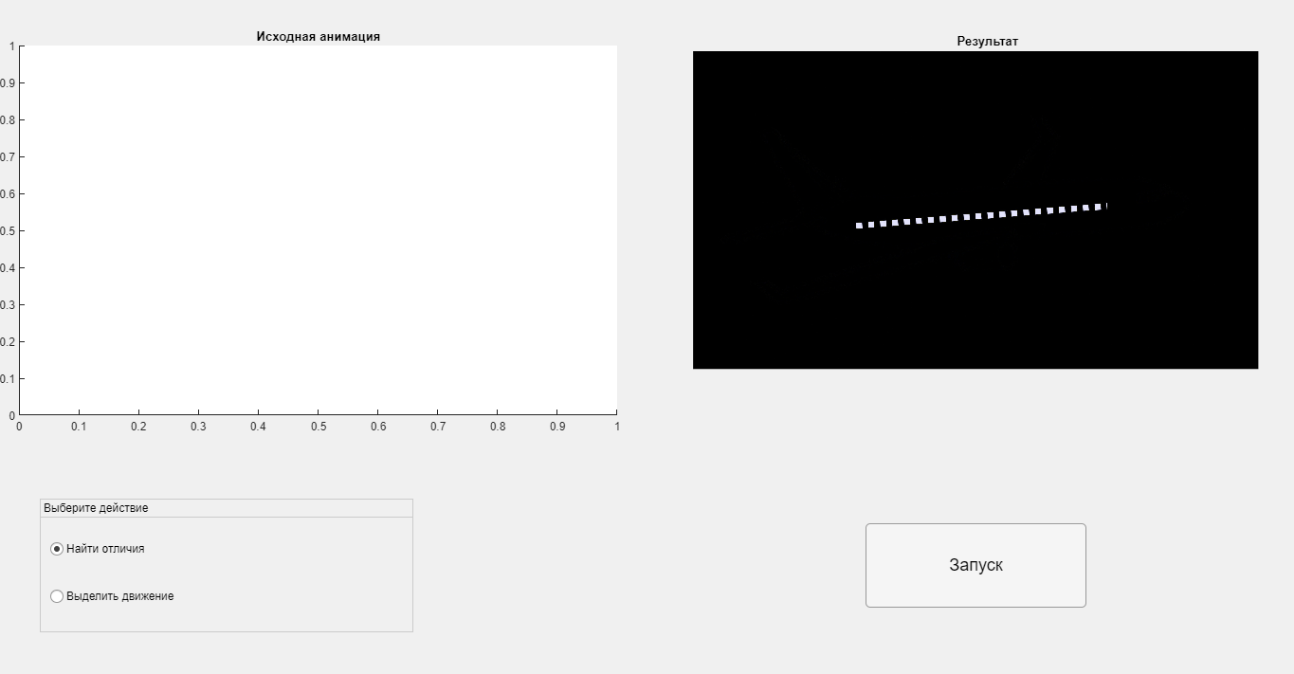


Рисунок 3 – поиск отличий

**Вывод**

В результате данной работы были разработаны: интерфейс, в котором можно реализовать выбор трёх функций для анимации; интерфейс, в котором реализован выбор файла из папки пользователем и выбор того, что должна делать программа (сравнение и вывод отличия или определение движение из анимации). В ходе работы мной были изучены основы работы с MATLAB AppDesigner, процессы создания и оформления интерфейсов, написание кода для работы интерфейса.

**Библиография**

1 ГОСТ 2.105-2019. Национальный стандарт Российской Федерации Единая система конструкторской документации. Общие требования к текстовым документам. – М.: Стандартинформ, 2019.

2 ГОСТ 7.32-2017. Система стандартов по информации, библиотечному и издательскому делу. Отчёт о научно-исследовательской работе. Структура и правила оформления. – М.: Стандартинформ, 2018.