

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

«Построение 3D графиков. Работа с mplot3d Toolkit»

**ОТЧЕТ по лабораторной
работе №6 дисциплины
«Технологии распознавания образов»**

Выполнил:

Сотников Андрей Александрович
2 курс, группа ПИЖ-б-о-21-1,
011.03.04 «Программная инженерия»,
направленность (профиль) «Разработка
и сопровождение программного
обеспечения», очная форма обучения

(подпись)

Проверил:

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Проработка примеров:

Построение 3D графиков. Работа с mplot3d Toolkit

```
In [4]: import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits.mplot3d import Axes3D
```

Линейный график

```
In [5]: x = np.linspace(-np.pi, np.pi, 50)
y = x
z = np.cos(x)
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.plot(x, y, z, label='parametric curve')

Out[5]: [<mpl_toolkits.mplot3d.art3d.Line3D at 0x19b690be650>]
```

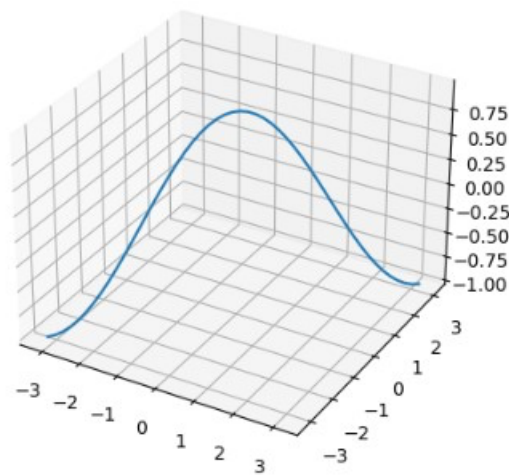


Рисунок 1 – Проработка примеров

Точечный график

```
In [10]: np.random.seed(123)
x = np.random.randint(-5, 5, 40)
y = np.random.randint(0, 10, 40)
z = np.random.randint(-5, 5, 40)
s = np.random.randint(10, 100, 40)
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.scatter(x, y, z, s=s)
```

Out[10]: <mpl_toolkits.mplot3d.art3d.Path3DCollection at 0x19b6d503110>

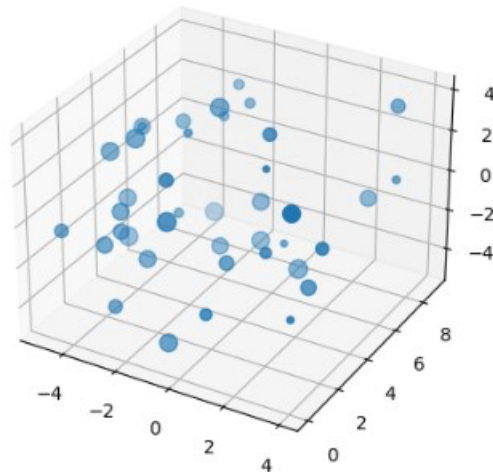


Рисунок 2 – Проработка примеров

Каркасная поверхность

```
In [11]: u, v = np.mgrid[0:2*np.pi:20j, 0:np.pi:10j]
x = np.cos(u)*np.sin(v)
y = np.sin(u)*np.sin(v)
z = np.cos(v)
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.plot_wireframe(x, y, z)
```

Out[11]: <mpl_toolkits.mplot3d.art3d.Line3DCollection at 0x19b6d577c50>

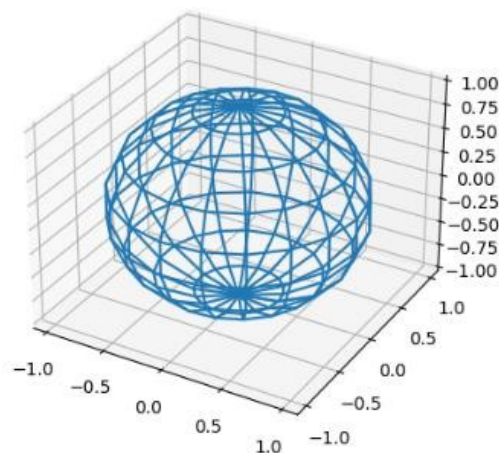


Рисунок 3 – Проработка примеров

Поверхность

```
In [12]: u, v = np.mgrid[0:2*np.pi:20j, 0:np.pi:10j]
x = np.cos(u)*np.sin(v)
y = np.sin(u)*np.sin(v)
z = np.cos(v)
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.plot_surface(x, y, z, cmap='inferno')
```

```
Out[12]: <mpl_toolkits.mplot3d.art3d.Poly3DCollection at 0x19b6d644250>
```

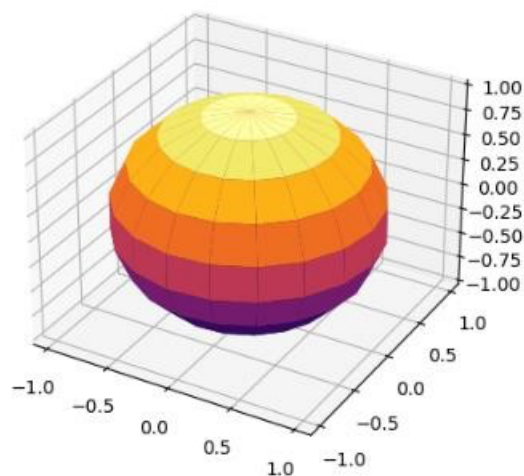


Рисунок 4 – Проработка примеров

Задание №1: Демонстрация работы с 3D графиками

```

In [4]: fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Создание трех лент Мёбиуса
for i in range(3):
    # Параметры ленты Мёбиуса
    theta = np.linspace(0, 2 * np.pi, 50)
    phi = np.linspace(-0.4, 0.4, 20)
    PHI, THETA = np.meshgrid(phi, theta)
    R = 1
    X = R * np.cos(THETA)
    Y = R * np.sin(THETA)
    Z = PHI

    # Преобразование координат для создания ленты Мёбиуса
    X, Y = X * (1 + Z * np.cos(THETA / 2)), Y * (1 + Z * np.cos(THETA / 2))
    Z = R * Z * np.sin(THETA / 2)

    # Смещение каждой ленты Мёбиуса по оси z
    Z += i * 3

    # Отображение ленты Мёбиуса
    ax.plot_surface(X, Y, Z)

plt.show()

```

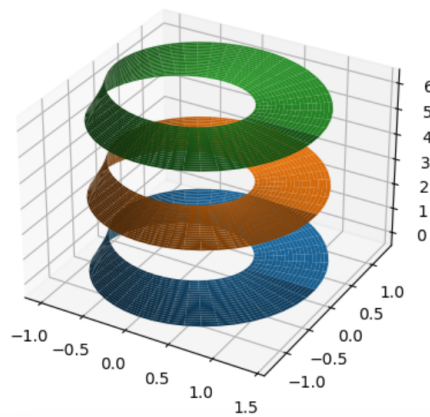


Рисунок 5 – Задание №1

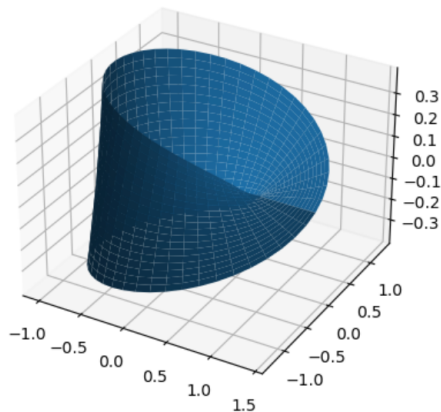
```
In [5]: fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')

# Параметры ленты Мёбиуса
theta = np.linspace(0, 2 * np.pi, 50)
phi = np.linspace(-0.4, 0.4, 20)
PHI, THETA = np.meshgrid(phi, theta)
R = 1
X = R * np.cos(THETA)
Y = R * np.sin(THETA)
Z = PHI

# Преобразование координат для создания ленты Мёбиуса
X, Y = X * (1 + Z * np.cos(THETA / 2)), Y * (1 + Z * np.cos(THETA / 2))
Z = R * Z * np.sin(THETA / 2)

# Отображение ленты Мёбиуса
ax.plot_surface(X, Y, Z)

plt.show()
```



In []:

Рисунок 6 – Задание №1

Контрольные вопросы

1. Как выполнить построение линейного 3D-графика с помощью matplotlib?

Для построения линейного графика используется функция `plot()`.

```
Axes3D.plot(self, xs, ys, *args, zdir='z', **kwargs)
```

2. Как выполнить построение точечного 3D-графика с помощью matplotlib?

Для построения точечного графика используется функция *scatter()*.

```
Axes3D.scatter(self, xs, ys, zs=0, zdir='z', s=20, c=None, depthshade=True,  
*args, **kwargs)
```

3. Как выполнить построение каркасной поверхности с помощью matplotlib?

Для построения каркасной поверхности используется функция *plot_wireframe()*.

```
plot_wireframe(self, X, Y, Z, *args, **kwargs)
```

4. Как выполнить построение трехмерной поверхности с помощью matplotlib?

Для построения поверхности используйте функцию *plot_surface()*.

```
plot_surface(self, X, Y, Z, *args, norm=None, vmin=None, vmax=None,  
lightsource=None, **kwargs)
```