

# Об одном методе вычисления мультифрактального спектра цифровых изображений

Мирзаянов Глеб Романович, мат-мех СПбГУ

Научный руководитель: Куликов Егор Константинович

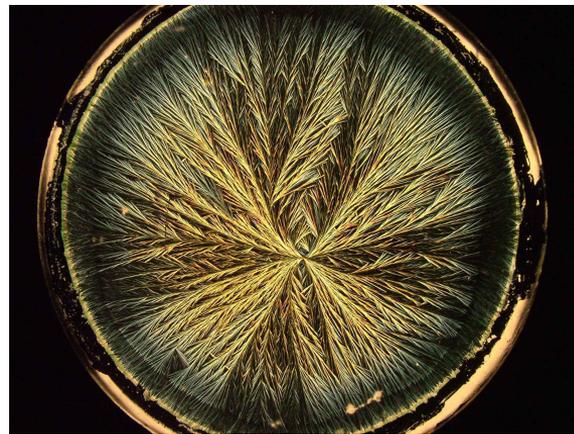
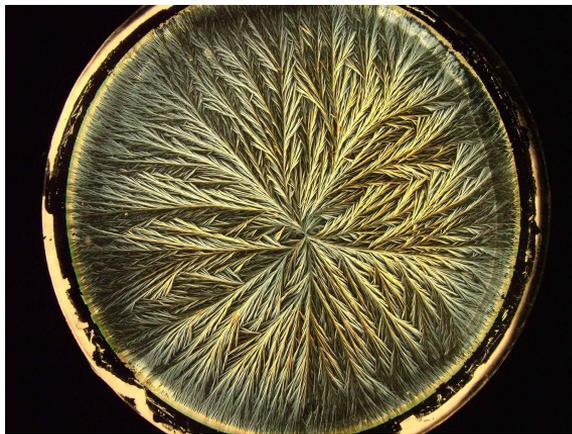
# Решаемая задача



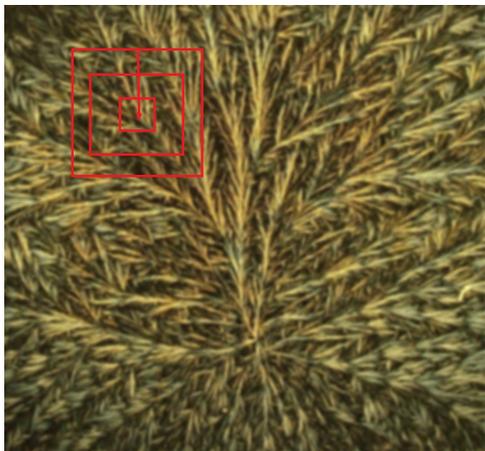
Перед нами стоит одна из актуальных задач нахождения классификационных признаков цифровых изображений, обладающих сложной структурой и чертами самоподобия.

# Область применения

- геология
- биология
- медицина
- техника



# Описание метода



Мера квадрата  $\mu(B(x,r))$  - сумма интенсивностей составляющих его пикселей.

Функция плотности в точке  $x$ :

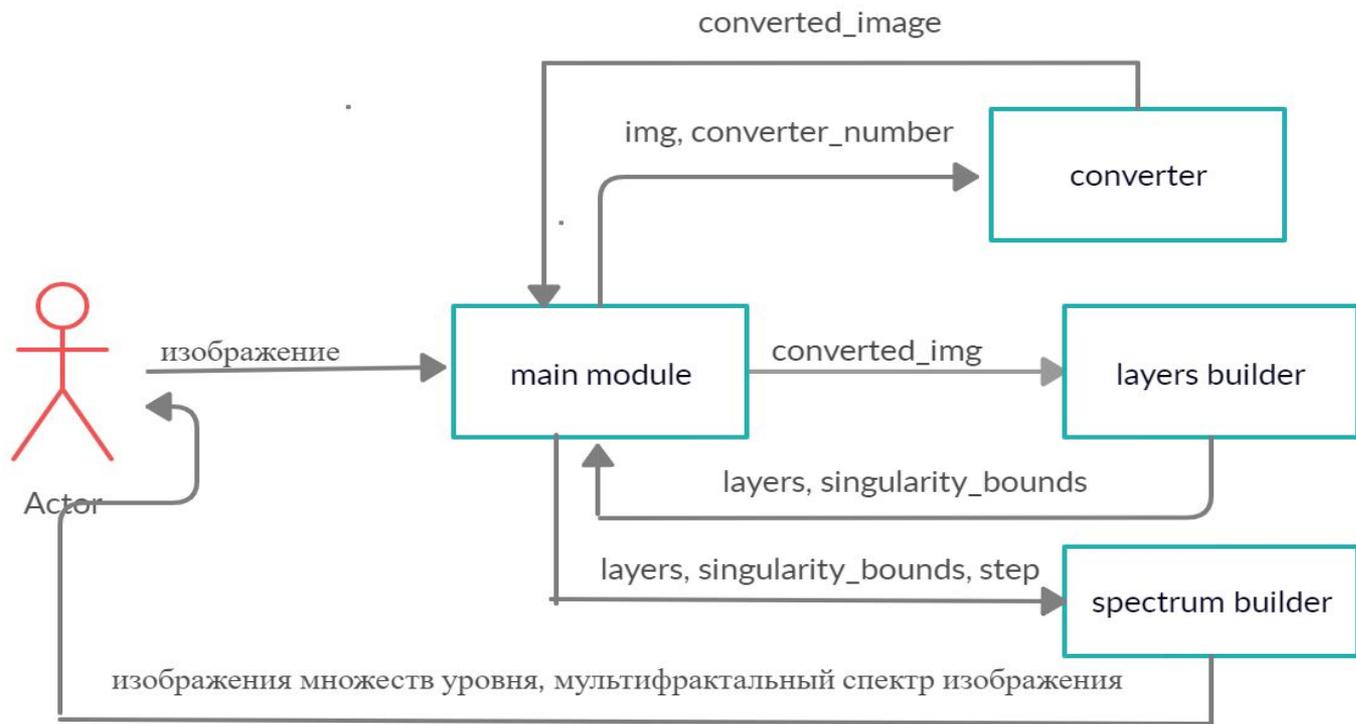
$$d(x) = \lim_{r \rightarrow 0} \frac{\log \mu(B(x, r))}{\log r}.$$

Для приближённого вычисления предела используется МНК.

Далее изображение разбивается на слои со схожими значениями функции плотности, после чего измеряется фрактальная размерность слоя.

$$E(\alpha, \varepsilon) = \{x \in R^2 : d(x) \in [\alpha, \alpha + \varepsilon)\}$$

# Реализация



# Визуализация

Реализованная программа поможет нам увидеть результаты метода на практике. Для наглядности мы построим:

- изображения множеств уровня
- график мультифрактального спектра
- трехмерную визуализацию множеств уровня

```

Fraktal main.py
Project
  Fraktal C:\Users\gleb3\PycharmProjects\F
    Test
    venv library root
    converter.py
    converter_types.py
    fig.png
    layers.py
    layers_builder.py
    least_squares.py
    main.py
    spectrum_builder.py
    structures.py
    visualizer.py
  External Libraries
  Scratches and Consoles
main.py
31 def main():
32     print("Создайте папку C:\\Pictures, сохраните в ней тестовое изображение\n"
33           "В этой же папке будут сохранены изображения, соответствующие множествам ур
34     print("Введите имя файла, например img.jpg")
35     print("Если вы хотите использовать другой путь, введите его целиком в формате "
36           "C:\\test\\image1.jpg")
37     input_data = input()
38     if ":" in input_data:
39         img = cv2.imread(input_data, 1)
40     else:
41         path = os.path.join(inputDirectory, input_data)

```

Python Console main main (1) main (2)

```

1
Создайте папку C:\Pictures, сохраните в ней тестовое изображение
В этой же папке будут сохранены изображения, соответствующие множествам уровня
Введите имя файла, например img.jpg
Если вы хотите использовать другой путь, введите его целиком в формате C:\test\image
>> |

```

Special Variables

# Технологии

-  Python
-  OpenCV
-  NumPy
-  Matplotlib

# Результаты

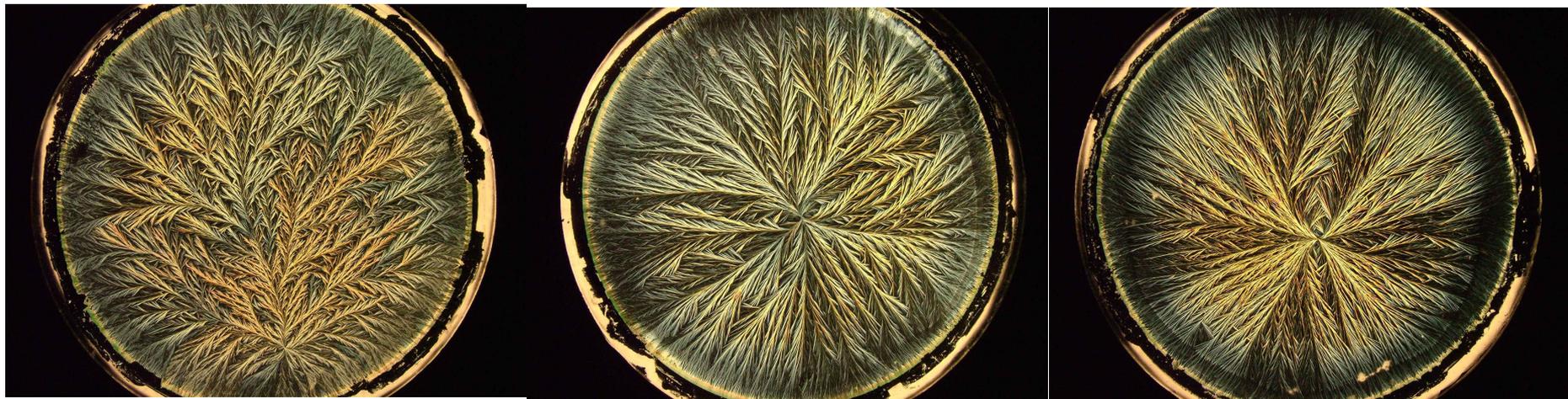


Рис. 1 BD milk

Рис. 2 МНР milk

Рис. 3 UHT milk

# Результаты

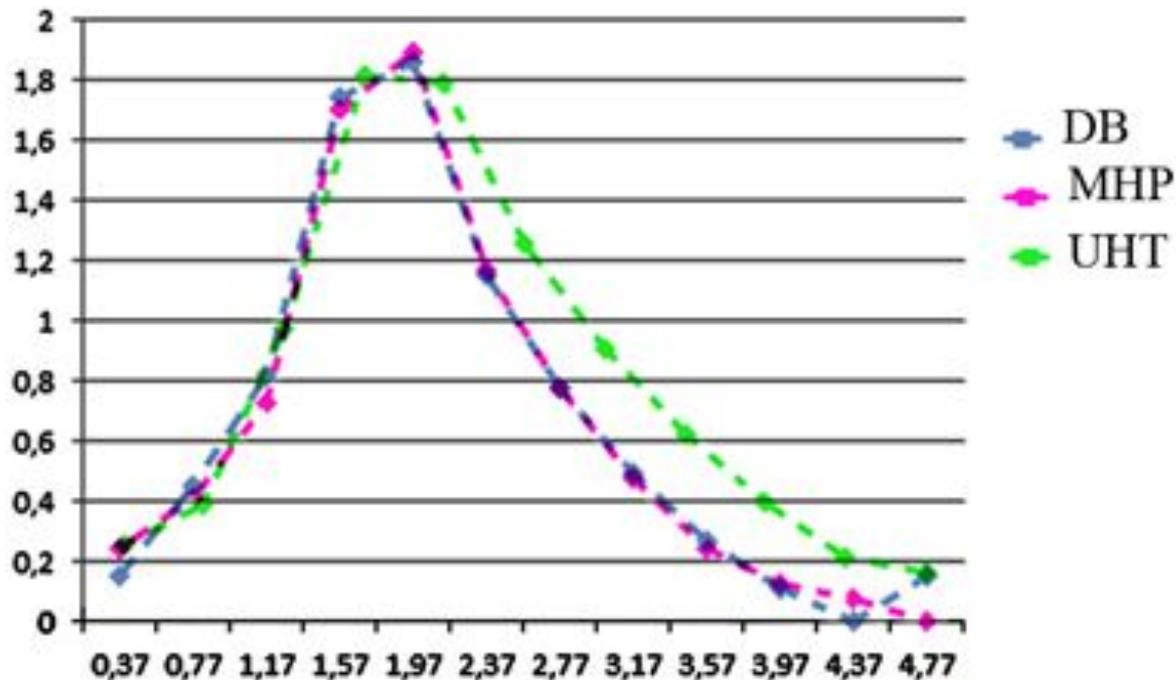
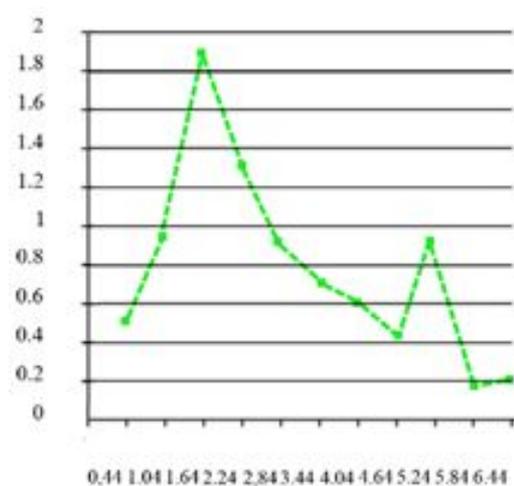
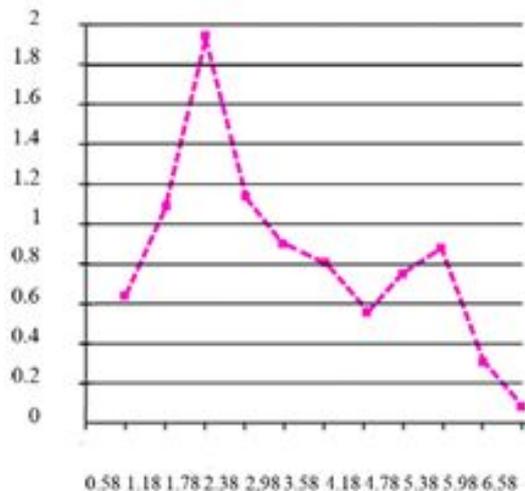
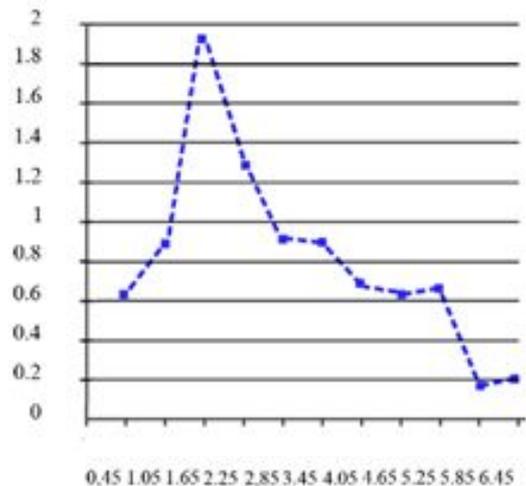


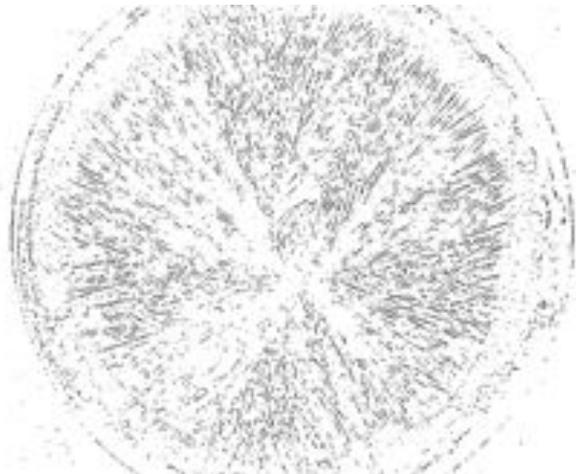
График мультифрактальных спектров изображений молока в полутоновой палитре

# Результаты

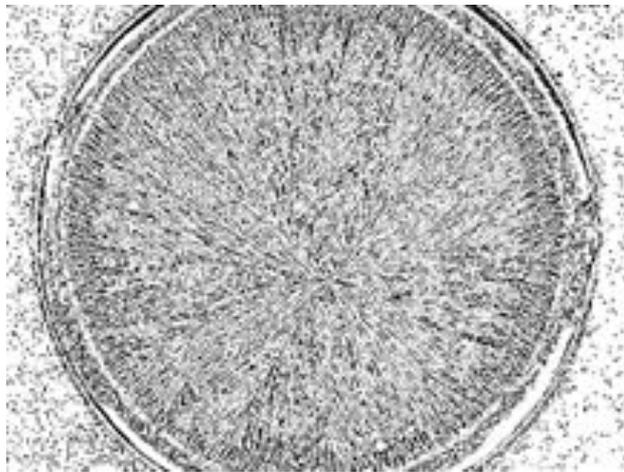


Мультифрактальные спектры для G-компонент изображений BD, MHP, UHT milk

# Результаты



[1,27; 1,67)



[2,07; 2,47)



[2,47; 2,87)

Примеры построения множеств уровня для изображений, имеющих больше всего точек

# Заключение

- Разработан инструментарий со средствами визуализации
- Проведен ряд экспериментов, указывающих на перспективность метода
- Выявлен ряд недостатков:
  - Образование ложных множеств уровней на границе
  - Высокая вычислимая сложность
  - Невозможность обработки некоторых изображений без предварительной корректировки

Выражаю благодарность Наталье Борисовне Ампиловой за помощь в проведении исследования.