

Министерство науки и высшего образования Российской Федерации  
Санкт-Петербургский политехнический университет Петра Великого  
Физико-механический институт

Работа допущена к защите  
Руководитель ОП  
\_\_\_\_\_ К.Н. Козлов  
«\_\_\_\_\_» \_\_\_\_\_ 2024 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
РАБОТА БАКАЛАВРА  
ОНЛАЙН-СЕРВИС ИИ-ДЕКОНВОЛЮЦИИ И ДЕНОЙЗИНГА ДЛЯ  
КОНФОКАЛЬНЫХ МИКРОСКОПОВ**

по направлению подготовки 01.03.02 Прикладная математика и информатика  
Направленность (профиль) 01.03.02\_02 Системное программирование

Выполнил

студент гр. 5030102/00201

И.М. Золин

Руководитель

старший преподаватель ВШПМиВФ, научный сотрудник ЛМН,

В.С. Чуканов

Консультант

доцент ВШБСиТ, научный сотрудник ЛМН,

Е.И. Пчицкая

кандидат физико-математических наук

Консультант

по нормоконтролю

Л.А. Арефьева

Санкт-Петербург  
2024

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
ПЕТРА ВЕЛИКОГО  
Физико-механический институт

УТВЕРЖДАЮ  
Руководитель ОП  
\_\_\_\_\_ К.Н. Козлов  
«\_\_\_\_\_» \_\_\_\_\_ 2024г.

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

студенту Золину Ивану Максимовичу гр. 5030102/00201

1. Тема работы: Онлайн-сервис ИИ-деконволюции и денойзинга для конфокальных микроскопов.

2. Срок сдачи студентом законченной работы: 25.05.2024

3. Исходные данные по работе:

Изображения различных флуоресцентных объектов: сферы, нейроны, трубочки, клетки; параметры съёмки. Данные для экспериментов были получены с использованием конфокальных микроскопов биологических объектов различного типа при сотрудничестве с Лабораторией молекулярной нейродегенерации.

Инструментальные средства:

- Языки программирования Python, JavaScript
- Среда разработки PyCharm
- Программная библиотека компьютерного зрения OpenCV
- Программная библиотека глубокого обучения PyTorch
- Система контроля версий git

4. Содержание работы (перечень подлежащих разработке вопросов):

- 4.1. Введение. Обоснование актуальности
- 4.2. Постановка задачи
- 4.3. Обзор существующих решений
- 4.4. Обзор алгоритма автоматической сегментации флуоресцентных сфер
- 4.5. Обзор метода глубокого обучения шумоподавления

- 4.6. Результаты обучения
  - 4.7. Тестирования метода шумоподавления
  - 4.8. Обзор онлайн-сервиса
  - 4.9. Заключение
5. Дата выдачи задания: 10.01.2024.

Руководитель ВКР \_\_\_\_\_ В.С. Чуканов

Консультант \_\_\_\_\_ Е.И. Пчицкая

Задание принял к исполнению 10.01.2024

Студент \_\_\_\_\_ И.М. Золин

## РЕФЕРАТ

На 40 с., 21 рисунок, 4 таблицы, 0 приложений

**КЛЮЧЕВЫЕ СЛОВА:** ФЛУОРЕСЦЕНТНАЯ МИКРОСКОПИЯ, КОНФОКАЛЬНАЯ МИКРОСКОПИЯ, ОНЛАЙН-СЕРВИС, КОМПЬЮТЕРНОЕ ЗРЕНИЕ, ДЕНОЙЗИНГ, СЕГМЕНТАЦИЯ, СВЁРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ, АРХИТЕКТУРА СЕТИ.

Тема выпускной квалификационной работы: «Онлайн-сервис ИИ-деконволюции и денойзинга для конфокальных микроскопов».

Данная работа посвящена разработке онлайн-сервиса, производящего различные процессы улучшения качества трёхмерных изображений люминисцентных объектов малого размера, регистрируемых конфокальными микроскопами. В рамках данного исследования были сформулированы следующие задачи:

- Изучение существующих алгоритмов, решающих схожие задачи.
- Разработка алгоритма машинного обучения для автоматической сегментации флуоресцентных сфер.
- Разработка алгоритма глубокого обучения денойзинга (шумоподавления) для трёхмерных изображений.
- Создание онлайн-сервиса, обеспечивающего взаимодействие с реализованными алгоритмами.

В результате проделанной работы был разработан онлайн-сервис, включающий в себя следующий функционал:

- Ручной и автоматический подходы к сегментации флуоресцентных сфер. Автоматический подход базировался на методах компьютерного зрения.
- Алгоритм глубокого обучения для денойзинга трёхмерных изображений, основанный на сети Noise2Noise (N2N) и фильтре Non-local means (NLM), обучен и протестирован на данных из разных микроскопов.
- Реализованный метод деконволюции Ричарсона-Люси для генерация экспериментальной функции рассеяния точки (эФРТ) и улучшения трёхмерных изображений.

По итогам работы сделан вывод, что онлайн-сервис эффективен для получения более точных изображений клеточных структур и материалов, что позволяет детально анализировать наблюдаемые объекты в медико-биологических исследованиях.

## ABSTRACT

40 pages, 21 figures, 4 tables, 0 appendices

**KEYWORDS:** FLUORESCENCE MICROSCOPY, CONFOCAL MICROSCOPY, ONLINE SERVICE, COMPUTER VISION, DENOISING, SEGMENTATION, CONVOLUTIONAL NEURAL NETWORKS, NETWORK ARCHITECTURE.

The subject of the graduate qualification work is «Online AI deconvolution and denoising service for confocal microscopes».

This work is devoted to the development of an online service that produces various processes to improve the quality of three-dimensional images of small-sized luminescent objects recorded by confocal microscopes. The following objectives were formulated as part of this research:

- Study of existing algorithms that solve similar problems.
- Development of a machine learning algorithm for automatic segmentation of fluorescent spheres.
- Development of a deep learning denoising (noise reduction) algorithm for three-dimensional images.
- Creation of an online service that provides interaction with the implemented algorithms.

As a result of this work, an online service was developed that includes the following functionality:

- Manual and automatic approaches to fluorescent sphere segmentation. The automatic approach was based on computer vision methods.
- A deep learning algorithm for denoising 3D images based on Noise2Noise (N2N) network and Non-local means (NLM) filter is trained and tested on data from different microscopes.
- Implemented Richardson-Lucy deconvolution method to generate expert point scattering function (ePRF) and improve 3D images.

It is concluded that the online service is effective in obtaining more accurate images of cellular structures and materials, which allows detailed analysis of observed objects in biomedical research.

## СОДЕРЖАНИЕ

Введение .....	7
Глава 1. Постановка задачи и обзор литературы .....	8
1.1. Постановка задачи .....	8
1.1.1. Постановка задачи денойзинга.....	9
1.1.2. Постановка задачи автоматической сегментации сфер.....	10
1.2. Обзор литературы .....	12
1.2.1. Существующие методы деконволюции .....	12
1.2.2. Элементы свёрточных нейронных сетей .....	13
Глава 2. Описание методов.....	17
2.1. Описание алгоритма 3D денойзинга .....	17
2.1.1. Описание алгоритма NLM (Non-local means) .....	17
2.1.2. Описание модели Noise2Noise .....	19
2.1.3. Описание предлагаемого решения .....	22
2.2. Описание алгоритма автоматической сегментации флуоресцентных сфер.....	23
Глава 3. Результаты.....	25
3.1. Результаты денойзинга .....	25
3.1.1. Используемые метрики .....	26
3.1.2. Результаты тренировки нейронной сети Noise2Noise .....	26
3.1.3. Эксперименты денойзинга .....	28
3.2. Результаты автоматической сегментации.....	35
3.3. Описание архитектуры онлайн-сервиса деконволюции.....	36
Список использованных источников.....	40

## ВВЕДЕНИЕ

Флуоресцентная микроскопия - это метод получения снимков светящихся объектов малых размеров. Он широко использует в разных областях: от материаловедения до нейробиологии. Флуоресцентная микроскопия имеет ряд преимуществ перед другими формами микроскопии, предлагая высокую чувствительность и специфичность.

Однако данный метод обладает и своими недостатками. Один из них заключается в том, что исходные изображения, получаемые с микроскопа, содержат шумы и искажения, которые могут существенно затруднить последующий анализ объекта. Поэтому учёным и экспертам требуются предварительный ряд обработок над изображениями (деконволюция, денойзинг). Кроме того, существуют ручные процессы, которые занимают много времени и необходимы для обработки изображений, поэтому требуют автоматизации. Один из таких процессов — это сегментация калибровочных мелких светящихся объектов (например, флуоресцентных сфер), необходимых для основных этапов обработки. В последние годы наблюдается значительный прогресс в области флуоресцентной микроскопии, что позволяет ученым получать изображения с все более высокой разрешающей способностью.

Целью работы является разработка ПО в виде онлайн-сервиса, способного улучшать изображения биологических образцов с помощью различных методов, включая аналитические и нейросетевые алгоритмы. Задачи, которые решает данный сервис, включают автоматическую сегментацию флуоресцентных сфер, алгоритм денойзинга (обесшумливания) трёхмерных изображений, комбинирующий в себе аналитический подход и метод глубокого обучения, а также деконволюцию (улучшение качества) изображений. Результат работы позиционируется как инструмент, который поможет ученым и исследователям получать более точные изображения клеточных структур и материалов, что позволит им делать более качественные выводы.

# ГЛАВА 1. ПОСТАНОВКА ЗАДАЧИ И ОБЗОР ЛИТЕРАТУРЫ

## 1.1. Постановка задачи

Исходные изображения, наблюдаемые с микроскопа, крайне неточны - содержат шумы и искажения из-за несовершенства системы регистрации и волновой природы света. Получаемое изображение  $I_y$  является результатом свёртки (конволюции) точного объекта  $I_x$  с функцией рассеяния точки (ФРТ) системы  $H$ , которая его формирует. [Sibarita2005] Помимо этого, важно помнить, что системы регистрации изображения, в рамках данной задачи конфокальной микроскопии, вносят добавочный фотонный шум  $N$ . Схема получения наблюдаемого изображения приведено на рис.1.1. Формальная запись наблюдаемого изображения будет выглядеть следующим образом:

$$I_y = I_x \circledast H + N \quad (1.1)$$

где  $I_x$  - точное изображение объекта,  $I_x \in \Omega_x$  - множество точных изображений объектов,  $I_y$  - наблюдаемое изображение объекта  $I_y \in \Omega_y$  - множество наблюдаемых изображений объектов,  $H$  - функция рассеивания точки,  $N$  - шум, вносимый регистрируемым прибором,  $\circledast$  - операция свёртки.

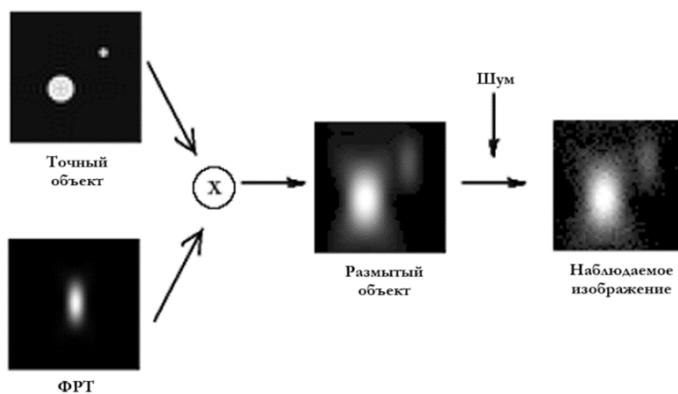


Рис.1.1. Схема свёртки точного объекта с ФРТ вместе с фотонным шумом

Не умоляя общности, рассмотрим изображения  $I_x, I_y$ , как дискретный набор значений. Тогда множества  $\Omega_x, \Omega_y$  могут быть представлены, как пространства  $\mathbb{R}^{l,r,c}$ , где  $l, r, c$  - число слоёв, высота и ширина каждого слоя изображения. В случае наличия нескольких каналов (например,  $w$ ), пространства могут быть представлены, как  $\mathbb{R}^{l,r,c,w}$ . Также важно отметить, что шум  $N$  рассматривается,

как элемент пространства  $\mathbb{R}^{l,r,c}(\mathbb{R}^{l,r,c,w})$ . Подробнее о природе шума и постановке задачи денойзинга будет рассмотрено в следующем пункте.

Методы восстановления "точного" изображения по наблюдаемому называются *деконволюцией* (процесс обратный к свёртке). Общая задача восстановления изображения может быть сведена к выбору функции  $\mathcal{F}(I_y)$ :

$$\hat{I}_x = \mathcal{F}(I_y) \quad (1.2)$$

где  $\hat{I}_x$  - аппроксимация  $I_x$ ,  $\hat{I}_x \in \Omega_x$

Однако зачастую процесс восстановления изображений рассматривается как два последовательных этапа: денойзинг (обесшумливание) изображений и сама деконволюция (т.е. процесс улучшения качества размытого снимка некоторого образца). Это обусловлено тем, что методы деконволюции искажают снимки при наличии шума.

### 1.1.1. Постановка задачи денойзинга

Флуоресцентная микроскопия характеризуется пуассоновско-гауссовым шумом (mixed Poisson–Gaussian, MPG), где пуассоновский шум является доминирующим источником шума.[2] Это объясняется тем, что количество фотонов, захваченных микроскопом, чрезвычайно мало. Следовательно, измеренный оптический сигнал во флуоресцентной микроскопии квантуется из-за дискретной природы фотонов, и в изображениях, полученных с помощью флуоресцентной микроскопии, преобладает пуассоновский шум, а не гауссовский шум, который характерен для фотографии[\*].

Шум может существенно ухудшать качество изображений и затруднять дальнейший анализ. Поэтому важно разработать алгоритм денойзинга, который будет эффективно удалять шум, сохраняя при этом важные структурные детали образцов.

Введём формальные обозначения для постановки задачи. Пусть  $I_z = I_x * H$  - размытое изображение (свёртка точного объекта и ФРТ), не содержащая шумов.  $I_z \in \Omega_z$ ,  $\Omega_z \in \mathbb{R}^{l,r,c}(\mathbb{R}^{l,r,c,w})$  - множество размытых изображений объектов. Тогда задача денойзинга сводится к нахождению такой функции  $\mathcal{F}_1(I_y)$ , что:

$$\hat{I}_z = \mathcal{F}_1(I_y) \quad (1.3)$$

где  $\hat{I}_z$  - аппроксимация  $I_z$ ,  $\hat{I}_x \in \Omega_x$ .

Задача деконволюции сводится к поиску функции  $\mathcal{F}_2(I_z)$ :

$$\hat{I}_x = \mathcal{F}_2(I_z) \quad (1.4)$$

где  $\hat{I}_x$  - аппроксимация  $I_x$ ,  $\hat{I}_x \in \Omega_x$ .

Общая задача восстановления изображений выглядит следующим образом:

$$\hat{I}_x = \mathcal{F}_2(\mathcal{F}_1(I_y)) \quad (1.5)$$

Задача денойзинга трёхмерных изображений формулируется следующим образом:

- Разработать алгоритм денойзинга трёхмерных изображений, где на вход поступает шумное изображение  $I_y \in \mathbb{R}^{l,r,c}$ , а на выходе получаем результат денойзинга:  $I_z \in \mathbb{R}^{l,r,c}$  (см. рис.1.2).
- Исследовать природу шума на реальных снимках с конфокальных микроскопов и доказать экспериментальным способом, что он распределяется в большей мере по Пуассону с некоторым параметром  $\lambda$ :

$$P(\lambda) = \frac{\lambda^k e^{-\lambda}}{k!} \quad (1.6)$$

где  $\lambda$  - частота возникновения, уровень шума,  $k$  - наблюдаемое значение случайной величины.

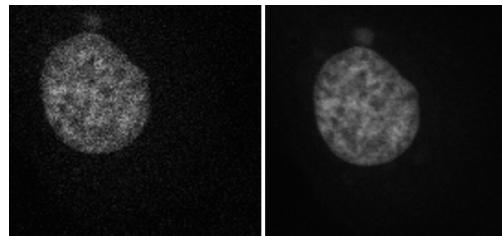


Рис.1.2. Пример работы денойзинга: слева - шумное изображение клетки, справа - обесшумленное (результат денойзинга)

### 1.1.2. Постановка задачи автоматической сегментации сфер

Для восстановления исходного изображения с помощью итерационных методов деконволюции необходимо в явном виде нахождение ФРТ. На практике точное значение ядра свёртки  $H$  найти очень сложно, поэтому используют приближенное значение. Методы, создающие такие приближения, делятся на две группы: теоретические и экспериментальные.

Теоретические методы вычисляют ФРТ на основе определенных моделей и функций, тогда как экспериментальные методы включают съемку объектов очень малого размера, например, калибровочных флуоресцентных сфер. Функции расстояния точки, полученные теоретическим путем, называют теоретическими ФРТ (тФРТ), а функции, полученные экспериментальным путем, — экспериментальными ФРТ (эФРТ). Оба метода обладают своими преимуществами и недостатками: так, например, теоретическая ФРТ не содержит шумов, которые могли бы приводить к неточностям при деконволюции, однако она же не отражает неточностей системы регистрации. Экспериментальный способ вычисления ФРТ имеет абсолютно обратные свойства.

Построение эФРТ происходит путём деконволюции размытой сферы, полученной из усреднения множества исходных, наблюдаемых с микроскопа снимков сфер, точным моделируемым изображением сферы. Более наглядное описание см на рис.1.3. Стоит отметить тот факт, что чем больше размытых микроскопом снимков сфер участвует в усреднении, тем точнее получается эФРТ и, как следствие, результаты аналитической деконволюции. Однако при извлечении большого количества сфер (80-100 и более), процесс сегментации может занять около 10-15 минут. Поэтому необходим автоматический подход, использующий методы компьютерного зрения, который позволил бы ускорить данный процесс, а также оставил возможность ручной корректировки результатов.

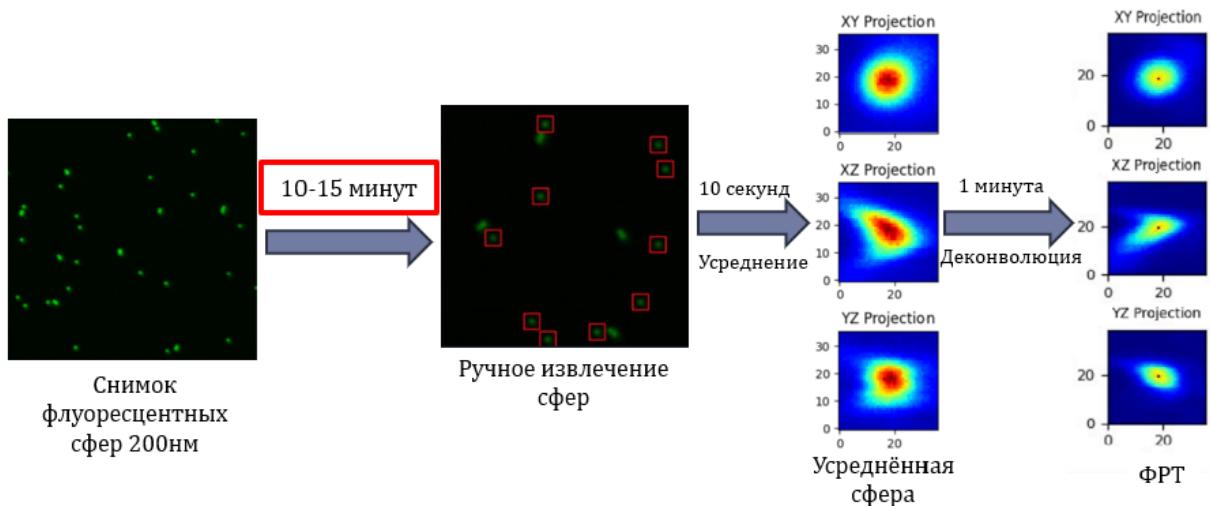


Рис.1.3. Схема экспериментального способа вычисления ФРТ

## 1.2. Обзор литературы

В данном параграфе рассмотрено 2 ключевых подхода, которые будут использоваться для описания существующих методов, так и создания новых в следующей главе.

### 1.2.1. Существующие методы деконволюции

*Деконволяция* - это процесс восстановления оригинального изображения, которое было искажено в процессе съёмки. Глобально эти методы можно разбить на 2 группы: использующие функцию рассеяния точки ( $H$ ) в процессе обработки, и не использующие. Вторую группу называют *методами "слепой" деконволяции*, обычно дают менее точные результаты, если сравнивать с первой группой.

Рассмотрим подкласс алгоритмов первой группы, основанных на природе шума, которые называются *статистические алгоритмы деконволяции*. Данный подкласс получен из алгоритма оценки максимального правдоподобия (MLE), написанного для минимизации пуассоновского шума на изображении.

*Метод Ричардсона-Люси (RL)* основан на методе максимального правдоподобия и предназначен для работы с шумами, следующими закону распределения Пуассона. Этот алгоритм итеративно улучшает оценку объекта, исходя из текущей оценки, наблюдаемого изображения и известной ФРТ.

$$\hat{I}_x^{k+1} = I_x^k \left( \frac{I_z}{I_x^k \circledast H} \right) \circledast H^* \quad (1.7)$$

где  $I_x^k$  - текущее приближение на итерации  $k$ ,  $I_z$  - наблюдаемое (размытое) изображение,  $H$  – функция рассеяния точки,  $H^*$  - обратное ядро свёртки (ФРТ, повернутая относительно начала координат на 180 градусов).

Этот алгоритм обладает свойством неотрицательности, т.е. если начальное значение неотрицательно, то и все последующие итерации будут неотрицательными. Однако, при наличии шума алгоритм может сходиться к решению, доминируемому шумом, что ухудшает качество изображения. Помимо использования методов денойзинга для решения проблемы усиления шума, существуют модификации RL-алгоритма с добавлением регуляризационных членов. Разберём наиболее часто используемые модификации:

A. Регуляризация Тихонова-Миллера (TM):

$$\hat{I}_x^{k+1} = I_x^k \left( \frac{I_z}{I_x^k \circledast H} \otimes H^* \right) \cdot (1 - 2\lambda_{TM} \nabla^2 I_x^k) \quad (1.8)$$

где  $\lambda_{TV}$  - параметр регуляризации, а  $\nabla^2$  - лапласиан объекта.

B. Общая вариация(Total variation, TV):

TV-регуляризация сохраняет контуры и границы объектов, уменьшая шум в однородных областях.

$$TV(I_x) = \int |\nabla I_x| dx dy dz \quad (1.9)$$

$$\hat{I}_x^{k+1} = I_x^k \left( \frac{I_z}{I_x^k \circledast H} \otimes H^* \right) - \lambda_{TV} \cdot TV(I_x^k) \quad (1.10)$$

где  $\lambda_{TV}$  - параметр регуляризации.

### 1.2.2. Элементы свёрточных нейронных сетей

В данном подпараграфе будет представлен краткий обзор архитектурных элементов свёрточных нейронных сетей (CNN), которые играют ключевую роль в обработке и анализе изображений.

#### 1.2.2.1. Свёрточный слой

*Свёрточный слой (Convolutional Layer)* является ключевым элементом CNN, часто используется для обработки изображений. Этот слой выполняет операцию свёртки (convolution), которая позволяет извлекать различные признаки из входных данных.

Операция свёртки двух функций  $f$  и  $g$  записывается так:

$$(f * g)(x) = \int_{\mathbb{R}^k} f(y)g(x - y) dy \quad (1.11)$$

где  $f, g : \mathbb{R}^k \Rightarrow \mathbb{R}$  - интегрируемые функции по мере Лебега на  $\mathbb{R}^n$ , а  $(f * g)(x) : \mathbb{R}^k \Rightarrow \mathbb{R}$ . Назовём  $f(x)$  вход,  $g(x)$  - ядро, а результат  $(f * g)(x)$  - карта признаков.

Распишем понятия в рамках нашей задачи:

*Фильтр(ядро свёртки)* – это матрица небольшого размера, которая скользит по изображению и выполняет операцию свёртки. Пусть задана размером  $k \times k$ . *Свёртка* - это операция, при которой фильтр скользит по входному изображению и вычисляет сумму произведений элементов фильтра и соответствующих элементов

области изображения. Если входное изображение имеет размер  $r \times c$  и фильтр  $k \times k$ , то выходное будет иметь размер  $(r - k + 1) \times (c - k + 1)$  без применения дополнения (*padding*). Если у нас  $n$  фильтров, то размер будет:  $(r - k + 1) \times (c - k + 1) \times n$ .

Свёрточные слои могут иметь несколько каналов. В этом случае, каждый канал имеет свой собственный фильтр, и результаты свёртки с каждым каналом суммируются.

Рассмотрим формулу свёрточного слоя двухмерного изображения с одним фильтром и одним каналом изображения. [Goodfellow-et-al-2016]

$$O(i,j) = (I * K)(i,j) = \sum_{u=0}^{k-1} \sum_{v=0}^{k-1} I(i+u, j+v) \cdot K(u,v) \quad (1.12)$$

где  $I$  - входное изображение размером  $r \times c$ ,  $K$  - фильтр размером  $k \times k$ ,  $O$  - выходное изображение, результат свёртки,  $i = 0, 1, \dots, r - k$  и  $j = 0, 1, \dots, c - k$  - точки карты признаков.

Также рассмотрим применения "дополнения"(Padding) и "шага"(Stride) для свёрточных слоёв:

- A. Padding добавляет рамку из нулей вокруг входного изображения, чтобы контролировать размер выходного изображения. Например, чтобы сохранить размер выходного изображения, если применить  $p = \frac{k-1}{2}$ , для фильтра размером  $k \times k$
- B. Stride – это шаг, с которым фильтр скользит по изображению. Если шаг равен  $s$ , то выходное изображение будет иметь размер:

$$\left(\frac{r-k}{s}\right) + 1 \times \left(\frac{c-k}{s}\right) + 1 \quad (1.13)$$

#### 1.2.2.2. Слой объединения (Pooling-слой)

*Слой объединения (Pooling Layer)* используется для уменьшения пространственных размеров карт признаков и количества параметров, сохраняя при этом наиболее важную информацию.

Наиболее распространённые слои объединения:

- Максимальное объединение (Max Pooling). Этот метод выбирает максимальное значение из квадратной области во входной карте признаков.
- Усреднение (Average Pooling). Этот метод вычисляет среднее арифметическое значение в каждой квадратной области входной карты признаков.

Pooling-слои помогают уменьшить количество параметров в модели и контролировать переобучение.

### 1.2.2.3. Слой повышения дискретизации (*UpSampling-слой*)

*Слой повышения дискретизации (UpSampling Layer)* выполняет обратную функцию Pooling-слоя. Его основная задача состоит в увеличении разрешения входных данных, поэтому он не содержит весов. Данный слой особенно полезен в задачах сегментации изображений и генеративных моделях.

Основные методы повышения дискретизации:

- Билинейная интерполяция (Bilinear Interpolation). Этот метод использует линейную интерполяцию между ближайшими пикселями для заполнения новых пикселей в увеличенном изображении.
- Максимальное объединение (Max Unpooling). Этот метод сохраняет максимальные значения из предыдущего слоя объединения и заполняет ими новые увеличенные области.
- Транспонированная свертка (Transpose Convolution). Этот метод применяет операцию, обратную свертке. В отличие от обычной свертки, транспонированная свертка увеличивает пространственные размеры входных данных, заполняя новые значениями, вычисляемыми на основе весов фильтра.

### 1.2.2.4. Слой активации

Слой активации (Activation Layer) вводит нелинейность в модель, что позволяет сети обучаться сложным функциям отображения. Наиболее распространённые функции активации:

- Сигмоидальная функция (*Sigmoid*):

Она преобразует входные значения в диапазон от 0 до 1. Недостаток этой функции – проблема исчезающего градиента(vanishing gradient problem) при обратном распространении ошибки.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (1.14)$$

- *ReLU* (*Rectified Linear Unit*):

Эта функция обнуляет отрицательные значения и сохраняет положительные без изменений. Она решает проблему затухания градиента, но имеет свои минусы: некоторые нейроны могут "умирать" становясь неактивными на протяжении всего обучения. Также ReLU несимметрична относительно нуля, что может привести к проблеме "расслоения" когда нейроны выдают

только положительные значения.

$$ReLU(x) = \max(0, x) \quad (1.15)$$

- *Leaky ReLU:*

В отличие от ReLU, Leaky ReLU возвращает само значение при положительном входе, а при отрицательном – линейную функцию от входа, умноженную на небольшой коэффициент, называемый "утечкой"(leak). Это предотвращает "умирание"нейронов и способствует лучшей сходимости и более точному обучению сети.

$$LaekyReLU(x) = \max(\alpha \cdot x, x) \quad (1.16)$$

- *Гиперболический тангенс (Tanh):*

Эта функция преобразует входные значения в диапазон от -1 до 1. Она также помогает уменьшить проблему исчезающего градиента и имеет более пологую кривую по сравнению с сигмоидальной функцией, что позволяет сети лучше распознавать сложные зависимости в данных. Гиперболический тангенс также имеет гладкую производную, что полезно для алгоритмов оптимизации, требующих вычисления градиента.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (1.17)$$

#### *1.2.2.5. Слой пакетной нормализации*

*Слой пакетной нормализации (Batch Normalization Layer)* - это метод нормализации промежуточных слоёв нейронной сети для ускорения и стабилизации процесса обучения. Этот метод помогает справляться с проблемами, такими как затухание или взрыв градиентов, и улучшает общую производительность сети.

Пакетная нормализация решает проблему так называемого "внутреннего ковариационного сдвига"(internal covariate shift), который возникает, когда распределение активаций каждого слоя меняется в процессе обучения. Это может замедлять обучение, так как каждый слой сети должен адаптироваться к изменяющемуся распределению входов.

## ГЛАВА 2. ОПИСАНИЕ МЕТОДОВ

Данная глава посвящена методам и алгоритмам, используемых в исследовании для обработки и анализа трёхмерных изображений. Глава включает в себя описание разработанных алгоритмов 3D денойзинга и автоматической сегментации флуоресцентных сфер, а также описание уже существующих подходов для предлагаемого решения.

### 2.1. Описание алгоритма 3D денойзинга

В этом параграфе подробно рассмотрены методы шумоподавления двумерных снимков, такие как Non-local Means и Noise2Noise, а также предлагаемое решение в виде комбинации этих методов и их модификации, способное эффективно устранять шум на трёхмерных изображениях с микроскопа.

#### *2.1.1. Описание алгоритма NLM (Non-local means)*

Классически существуют два типа моделей по шумоподавлению, а именно локальные и нелокальные. Преимущество первого подхода заключается в его скорости, однако его недостаток состоит в невозможности эффективного сохранения краёв изображений, что проявляется в виде размытия границ и деталей на изображении. Примером локального подхода является Гауссово размытие, а нелокального - метод Non-local means(NLM)[\*].

NLM (Non-local means) - это алгоритм для устранения шума на изображении, который обновляет значения пикселей на основе взвешенного среднего значений других пикселей, определённых как наиболее схожие. В отличие от локальных методов, которые используют значения соседних пикселей, NLM использует информацию из всего изображения, что позволяет ему лучше сохранять детали и текстуры. Диаграмма деятельности метода Non-local means приведена ниже на рис.2.1 .

#### **Алгоритм метода нелокального среднего (NLM)**

##### A. Создание окна поиска:

Создаётся большое окно поиска  $S$  размером  $K \times K$ , центрированное на пикселе  $p$ . Одновременно создаётся патч(блок пикселей)  $N_p$  размером  $L$ , центрированном на пикселе  $p$ .

**B. Вычисление схожести патчей:**

В окне поиска устанавливается новый патч  $N_q$ , центрированный на пикселе  $q$ . Затем вычисляется евклидово расстояние  $d(p,q)$  между  $N_p$  и  $N_q$ . Далее производится взвешивание патчей с использованием гауссовой функции:

$$\tilde{w}(p,q) = e^{-\frac{d(p,q)}{h^2}} \quad (2.1)$$

где  $h$  - параметр фильтрации (сглаживания),

$d(p,q) = \|I_y(N_p) - I_y(N_q)\|_2^2$  - евклидово расстояние между  $N_p, N_q$

**C. Нормализация весов:**

Проход по всем точкам  $q (N_q \in S)$  и нормализация весов:

$$w(p,q) = \frac{\tilde{w}(p,q)}{\sum_{q \in S} \tilde{w}(p,q)} \quad (2.2)$$

Согласно формулам, коэффициент веса удовлетворяет условиям:

$$\begin{cases} \sum_q w(p,q) = 1 \\ 0 \leq w(p,q) \leq 1 \end{cases} \quad (2.3)$$

**D. Вычисление результирующего значения:**

Вычисление результирующего значения обесшумленного пикселя с помощью взвешенной суммы:

$$I_z(p) = \sum_{q \in S} w(p,q) I_y(q) \quad (2.4)$$

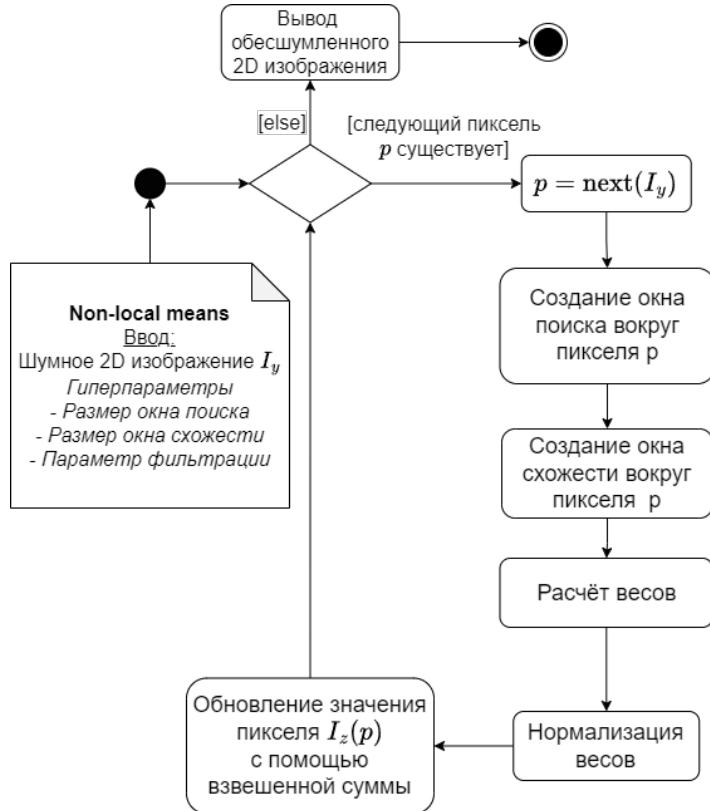


Рис.2.1. Диаграмма деятельности Non-local means

### 2.1.2. Описание модели Noise2Noise

В качестве метода глубокого обучения была выбрана модель Noise2Noise. [\*] Данный подход часто применяется для обработки шумных изображений, таких как флуоресцентные микроскопические снимки с пуассоновско-гауссовым шумом (MPG). В данном описании рассматривается модифицированная версия модели Noise2Noise, включающая дополнительные слои активации, адаптивное обучение и настроенные гиперпараметры.

**Архитектура сети** изображена на рис.2.2 и представляет собой свёрточную нейронную сеть (CNN), состоящую из энкодера (Encoder) и декодера (Decoder), разбитых на 5 уровней повышения/понижения дискретизации. Каждый уровень состоит из последовательно идущих 2D свёрточных слоёв  $3 \times 3$ , пакетной нормализации, слоя активации Leaky ReLU и соответствующего слоя повышения/понижения дискретизации.

Из описания архитектуры можно заметить, что данная нейронная сеть является *полносвёрточной*, так как состоит из слоёв свёртки, повышения и понижения дискретизации, активации и пакетной нормализации. Для данной задачи сеть обладает важным свойством - способность обрабатывать

входные данные различных размеров. Для данной сети имеется 5 уровней понижения/повышения дискретизации, единственным ограничением на размер входных изображений будет кратность  $2^5 = 32$  вдоль всех осей.

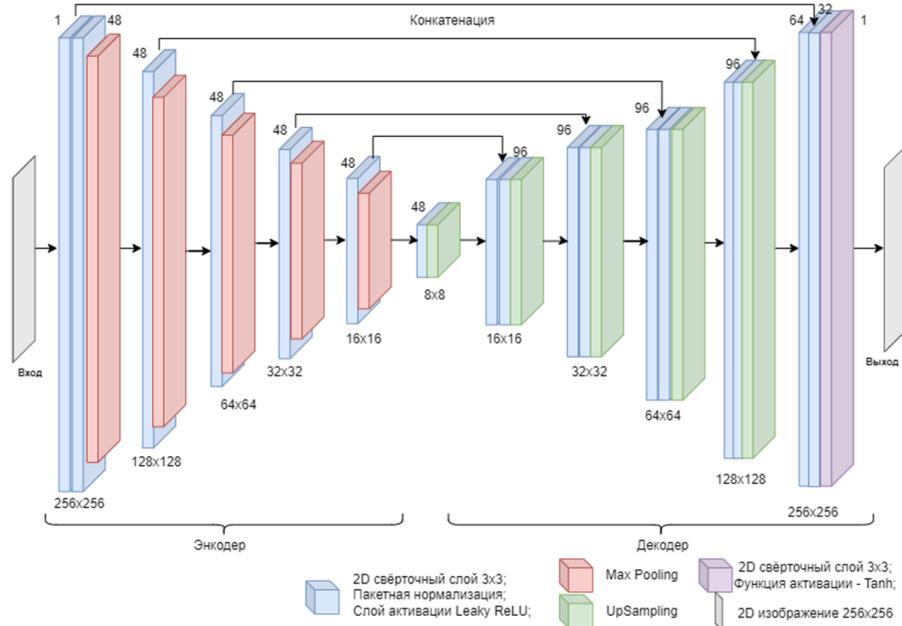


Рис.2.2. Архитектура сети Noise2Noise

**Основная идея** Noise2Noise заключается в том, что обучение происходит на паре зашумлённых изображений. Данный факт позволяет сети эффективно удалять шум без необходимости использования чистых изображений в процессе обучения:

- Вместо использования чистого изображения в качестве цели (label), как это делается в традиционных методах, Noise2Noise использует другое зашумленное изображение того же объекта в качестве цели.

$$I_{y1} = I_z + N_1, I_{y2} = I_z + N_2, \hat{I}_z = \mathcal{F}_1(I_{y1}) \quad (2.5)$$

где  $I_z$  - точный сигнал,  $N_1, N_2$  - разные шумы, порождающие разные входное и целевое изображения соответственно:  $I_{y1}, I_{y2}, \hat{I}_z$  - восстановленное изображение, результат  $\mathcal{F}_1.[*]$

Поскольку  $N_1, N_2$  - случайные величины, они не коррелируют между собой, что позволяет модели учиться выделять общий сигнал  $I_z$  присутствующий в обоих изображениях.

- Это позволяет модели учиться на данных, где шум разный (например, разные условия съёмки), но сам объект остается неизменным.

Данную задачу можно отнести к классу задач с семантическим разрывом, следовательно можно выделить класс функций  $\mathcal{F}_1(I_y, \theta)$ , который позволит найти

искомую функцию  $\mathcal{F}_1(I_y)$ :

$$\mathcal{F}_1(I_y) = \mathcal{F}_1(I_y, \hat{\theta}) \quad (2.6)$$

где параметр  $\hat{\theta}$  ищется следующим образом:

$$\hat{\theta} = \arg \min_{\theta} \mathcal{L}(\hat{I}_z, I_{y2}) = \arg \min_{\theta} \mathcal{L}(\mathcal{F}_1(I_y), I_{y2}) \quad (2.7)$$

где  $\theta$  - веса рассматриваемой модели,  $\mathcal{L}$  - функция ошибки.

В качестве функции потерь  $\mathcal{L}$  была взята среднеквадратичная ошибка (MSE), так как она хорошо штрафует за выбросы:

$$MSE(\hat{I}_z, I_{y2}) = \frac{1}{N} \sum_{i=1}^N (\hat{I}_{zi} - I_{y2,i})^2 \quad (2.8)$$

где  $i = 1..N$  - параметр итерирования по всем точкам изображения,  $N$  - общее число точек в изображении.

### Основные модификации модели Noise2Noise

- *Нелинейный слой активации (tanh):*

В исходной архитектуре Noise2Noise используется линейный выходной слой. В модифицированной версии добавлен нелинейный слой активации (tanh) после финального свёрточного слоя. Это помогает лучше моделировать сложные зависимости и улучшает качество денойзинга.

Гиперболический тангенс:

$$y_i = \frac{e^{x_i} - e^{-x_i}}{e^{x_i} + e^{-x_i}} \quad (2.9)$$

преобразует входное значение в выходное в диапазоне  $[-1, 1]$  на  $i$ -ом слое.

- *Адаптивное обучение:*

Для ускорения и улучшения процесса обучения используется стратегия адаптивного обучения, известная как *one cycle policy*. Используется линейно-косинусное изменение скорости обучения - сначала происходит линейное увеличение скорости обучения, а затем её косинусное уменьшение.

$$\begin{cases} lr(t) = lr_{low} + \frac{t}{start} \times (lr_{max} - lr_{low}), & t \leq t_{start}, \\ lr(t) = lr_{end} + \frac{lr_{max} - lr_{end}}{2} \times (\cos(\pi \times t) + 1), & t > t_{start} \dots \end{cases} \quad (2.10)$$

$t \in [0, 1]$  - процент выполнения обучения

$t_{start}$  - момент перехода от линейного к косинусному изменению

$$\begin{aligned}
 lr_{max} &= 0.0001 - \text{максимальная скорость обучения} \\
 lr_{low} &= \frac{lr_{max}}{25} - \text{стартовая скорость обучения, } div = 25 \\
 lr_{end} &= \frac{lr_{max}}{10^4} - \text{минимальная скорость обучения}
 \end{aligned}$$

### *2.1.3. Описание предлагаемого решения*

Помимо реализованных алгоритмов денойзинга двухмерных изображений, описанных выше, был разработан алгоритм шумоподавления трёхмерных снимков, комбинирующий в себе оба метода, позволяющий совмещать некоторые достоинства каждого из них.

Общая структура предлагаемого алгоритма представлена на рис.2.3. Процесс обработки трёхмерных изображений делится на три этапа:

A. *Предварительная обработка трёхмерного изображения*, включающая следующие шаги:

1. Перевод изображения в монохром
2. Z-нормализация

$$x_{normalized} = \frac{x - \mu}{\sigma} \quad (2.11)$$

где  $\mu$  - среднее,  $\sigma$  - СКО конкретного слоя

3. MinMax нормализация

$$x_{scaled} = \frac{x - x_{min}}{x - x_{max}} \quad (2.12)$$

где  $x_{min}, x_{max}$  - минимальные максимальные значения всего трёхмерного изображения

B. *Основной этап обработки изображения.*

Результат предобработки передаётся на вход основному этапу, где пользователь задаёт способ обработки изображений: нелокальный фильтр, нейросетевой подход или их комбинации в различном порядке. Денойзинг производится для каждого слоя независимо друг от друга.

C. *Склейка слоёв.*

На выход подаётся трёхмерное изображение, склеенное из обработанных ранее слоёв.

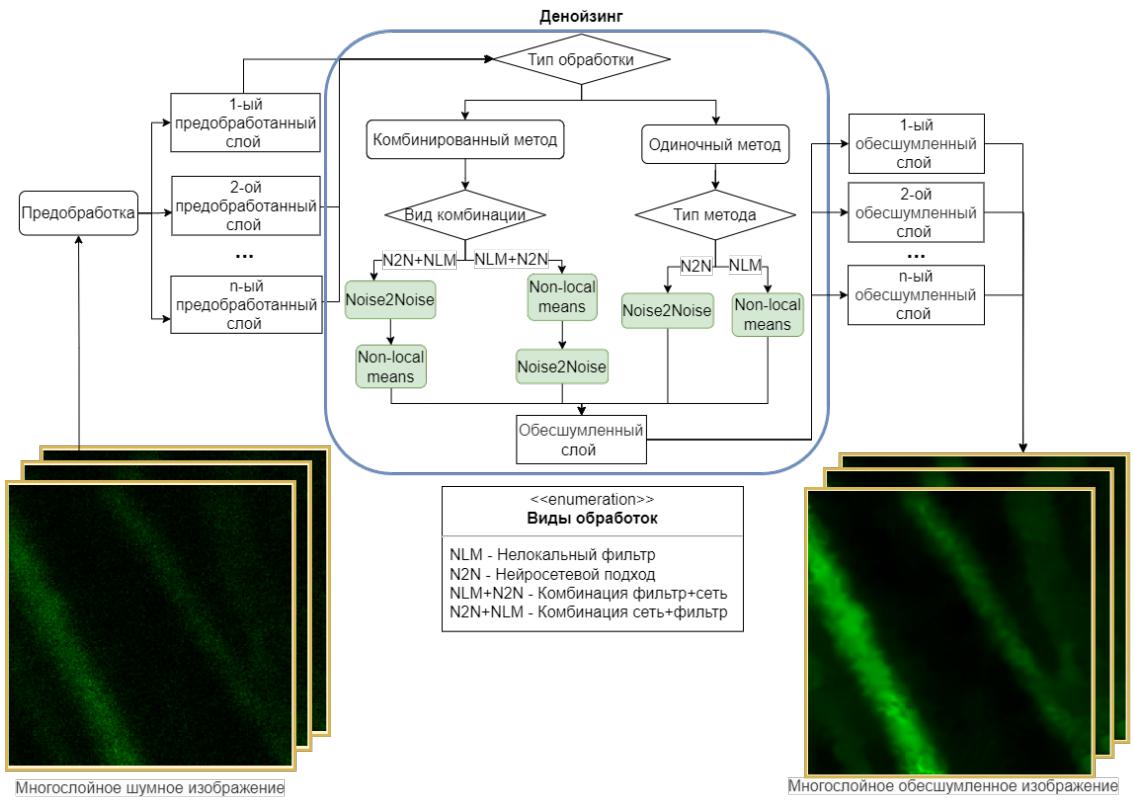


Рис.2.3. Алгоритм 3D денозинга

## 2.2. Описание алгоритма автоматической сегментации флуоресцентных сфер

Для ускорения процесса ручной сегментации был предложен автоматический подход с использованием методов компьютерного зрения. Для наглядности работы алгоритма продемонстрирована диаграмма деятельности на рис.2.5.

### Алгоритм автоматической сегментации сфер

#### A. Проведение предварительной обработки

1. Перевод изображения в монохром
2. Применение Гауссова размытия с использованием гиперпараметра радиуса размытия

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2.13)$$

где  $x, y$  - координаты точки,  $\sigma$  - радиус размытия (стандартное отклонение Гауссового распределения).

Изображение сворачивается с Гауссовым ядром - каждый пиксель изображения заменяется взвешенной суммой соседних пикселей, где веса определяются значениями Гауссового ядра.

### 3. Бинаризация с гиперпараметром порога бинаризации

- B. Анализ связанных компонент - этап, на котором происходит детекция всех сфер (и слипшихся, и одиночных; пример видов сфер на рис.2.4). Результат анализа - вывод следующих параметров: количество компонент связности (количество сфер), статистика компонент (координаты границ сфер, объёмы сфер), центроиды (центры сфер).



Рис.2.4. Виды сфер в результате анализа связанных компонент: *a* — одиночная сфера; *b* — слипшиеся сферы

- C. Фильтрация сфер на основе предложенного объёма правильной (одиночной) сферы. Если объём конкретной сферы меньше заданного, то она распознаётся как правильная и сохраняется.
- D. Извлечение трёхмерных сфер с заданным гиперпараметром ограничивающего прямоугольника.
- E. Усреднение флуоресцентных сфер. Считается выборочное среднее по всем извлечённым.
- F. Вывод результатов (усреднённой сферы и снимка с сегментированными сферами).

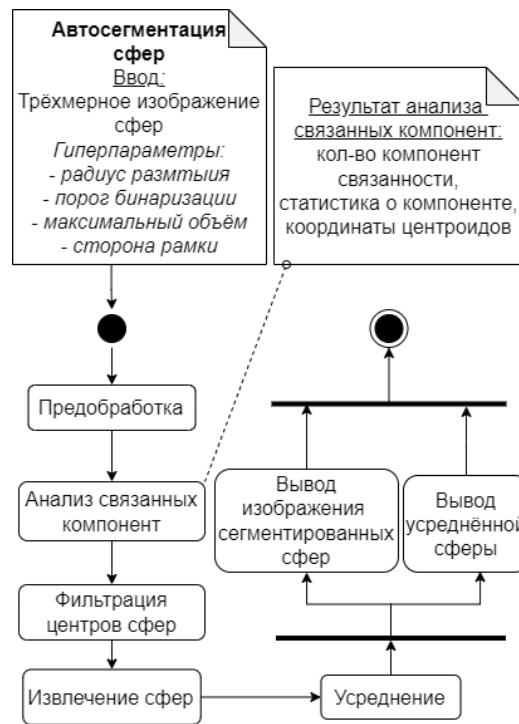


Рис.2.5. Диаграмма деятельности автоматической сегментации сфер

### Замечание

Значения гиперпараметров подбирались экспериментальным способом. Благодаря быстрой работе методов и заданным значениям по умолчанию, у человека есть возможность менять показатели в реальном времени, отслеживая динамику работы.

## ГЛАВА 3. РЕЗУЛЬТАТЫ

Данная глава посвящена анализу результатов экспериментов по денойзингу с использованием различных методов, а также автоматической сегментации трёхмерных сфер. Особое внимание уделяется результатам интеграции и внедрения реализованных методов в онлайн-сервис.

### 3.1. Результаты денойзинга

В этом разделе особое внимание уделяется процессу обучения модели Noise2Noise, а также экспериментам, проводившимся на реальных и синтетических снимках различными методами денойзинга.

### 3.1.1. Используемые метрики

Для оценки качества работы методов денойзинга использовались следующие метрики [\*]:

- A. Время работы - время, затраченное на выполнение процесса денойзинга.
- B. Корень из средней квадратичной ошибки(RMSE)

Среднеквадратичная ошибка(СКО):

$$MSE(I_1, I_2) = \frac{1}{N} \sum_{i,j,k} (I_{1(i,j,k)} - I_{2(i,j,k)})^2 \quad (3.1)$$

Доля ошибки RMSE:

$$RMSE(I_1, I_2) = \sqrt{\frac{MSE(I_1, I_2)}{N}} \quad (3.2)$$

где  $I_1, I_2$  - изображения одинаковой размерности,  $i, j, k$  - переменные итерирования по всем точкам слоя изображения,  $N = l \cdot r \cdot c$  - число всех точек в изображениях,  $l$  - число слоёв изображений,  $r$  - высота каждого слоя,  $c$  - ширина каждого слоя.

- C. Отношение пикового сигнала к шуму (PSNR)

$$PSNR(I_1, I_2) = 10 \log_{10} \left( \frac{R^2}{MSE} \right) \quad (3.3)$$

где  $R$  - максимальное возможное значение пикселя в изображении.

Если значения в каждой точке изображения кодируются, например, тремя цветами, каждый из которых хранится в 8 битах, то максимальное значение R будет следующим:

$$R = 2^8 = 256 \quad (3.4)$$

### 3.1.2. Результаты тренировки нейронной сети Noise2Noise

В данной задаче использовался метод обучения с учителем (supervised learning) для оценки параметров шума Пуассона на изображениях. В качестве минимизируемой функции применялась метрика MSE. Обучение осуществлялось с использованием нейронной сети Noise2Noise, которая позволяла эффективно обучаться без необходимости иметь точное изображение в паре.

Для обучения использовался FMD (Fluorescence Microscopy Denoising) датасет [\*], состоящий из 12 000 изображений с различным уровнем шума [1, 2, 4,

8, 16] (итого 60 000 изображений). Тренировочный набор данных был разделён на 2 выборки: тренировочную и валидационную в отношении 4:1.

Процесс обучения был сопровождён следующими параметрами:

- Оптимизатор Адама (Adam - Adaptive Moment Estimation)[\*], выбранный за его эффективность и способность адаптивно изменять скорость обучения.
- Размер batch = 10, что позволило эффективно использовать память GPU и стабилизировать процесс обучения.
- Установлена скорость обучения на 0.0001 с последующим применением линейно-косинусной функции изменения[\*].
- Число эпох: 250, что оказалось оптимальным для предотвращения переобучения.

Обучение проводилось на вычислительном кластере из 4 GPU V100 64Гб, предоставленном Yandex DataSphere. Процесс занял около 45 минут, включая загрузку данных, обучение модели и оценку результатов на валидационных данных.

Каждые 5 эпох проводился тест на различных изображениях. Это позволяло отслеживать динамику обучения. Выводился визуальный анализ из 8 разных изображений рис.3.1.

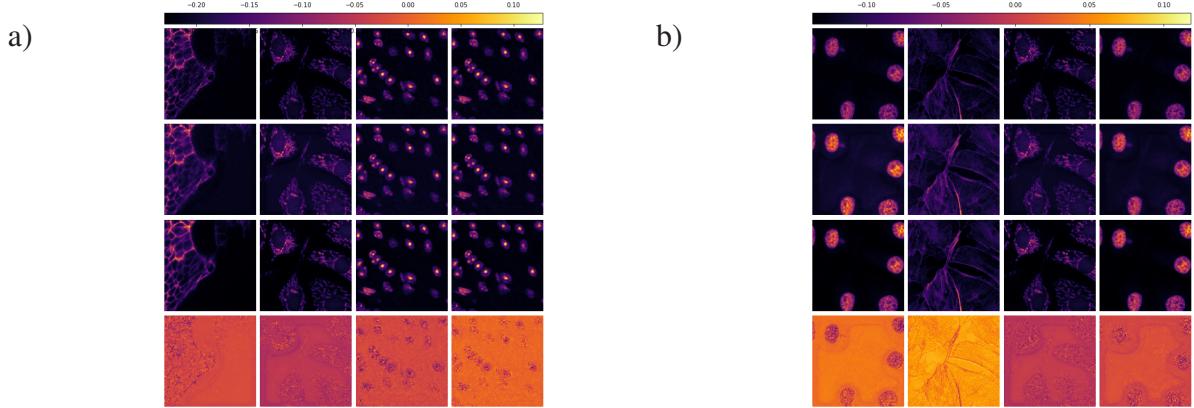


Рис.3.1. Визуальный анализ на 50 эпохе, содержащий несколько тестовых изображений (слева-направо), а также этапы денойзинга (сверху-вниз): чистое изображение, зашумлённое, обесшумленное, разница между обесшумленным и чистым: *a* — тест из 1-ых 4-ёх изображений; *b* — тест из 2-ых 4-ёх изображений

**Итоговые результаты.** В конце обучения были построены графики сходимости метрики PSNR и RMSE. Оценки показали высокую точность предсказаний рис.3.2.

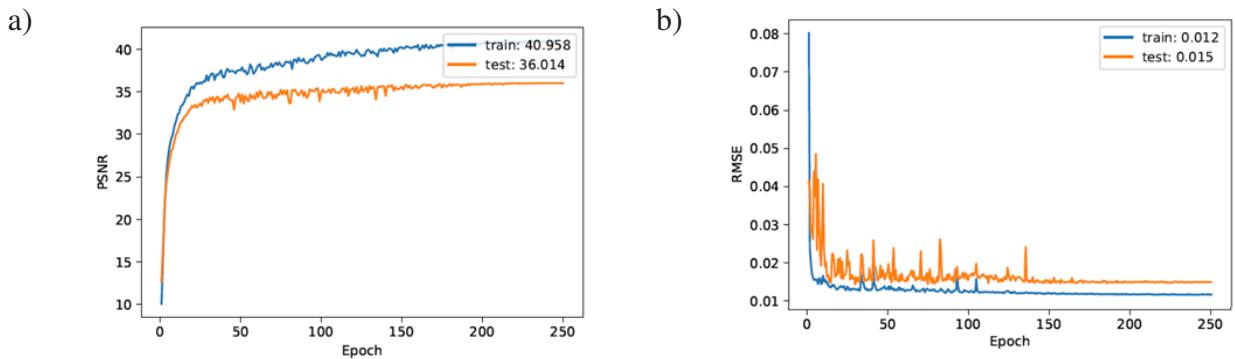


Рис.3.2. Графики, отображающие метрики PSNR и RMSE в процессе обучения на протяжении всех 200 эпох. Синий график - результаты на тренировочной выборке; оранжевый - результаты на валидационной выборке: a — метрика PSNR; b — метрика RMSE

### 3.1.3. Эксперименты денойзинга

В этом подпараграфе будет рассмотрен поробный анализ работы алгоритма 3D денойзинга с использованием различных методов, как нейросети Noise2Noise, нелокального фильтра Non-local means, так и их комбинаций. Оценка будет производится с помощью описанных ранее метрик.

Измерение времени работы методов денойзинга проводилось на центральном процессоре (CPU) Intel Xeon Gold 6230 и 64 Gb RAM, предоставленном также от Yandex DataSphere.

#### 3.1.3.1. Оценка метрик алгоритма 3D денойзинга

Первым экспериментом работы методов стало обесшумливание синтетического изображения трубочек размером 7x2048x2048, на который накладывался различный шум Пуассона с параметром  $\lambda \in [1, 2, 4, 8, 16]$ .

Сначала рассмотрим результаты работы методов для уровня шума  $\lambda = 16$ . На рис.3.3 можно увидеть, что все методы справляются с задачей денойзинга, особенно касаемо областей, где нет объекта(фон). Также можно заметить, что нелокальный фильтр и соответствующая комбинация фильтр+сеть удаляет части объекта с низкой интенсивностью. Напротив методы Noise2Noise и комбинация сеть+фильтр лучше сохраняют детали изображения.

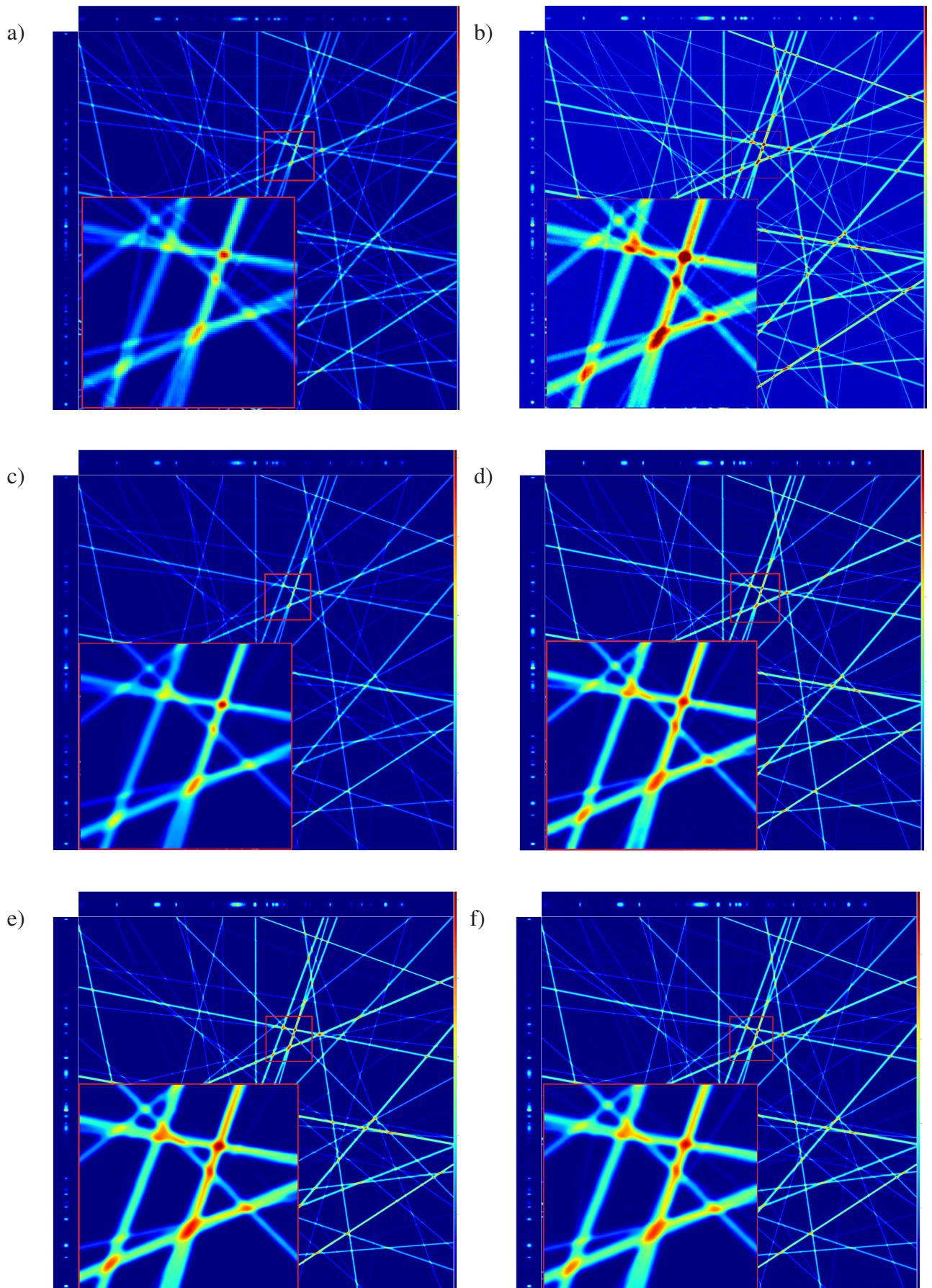


Рис.3.3. Визуальный анализ на синтетических данных для  $\lambda = 16$ : a — Результат нелокального фильтра; b — Результат нейросети; c — Результат комбинации фильтр+сеть; d — Результат комбинации сеть+фильтр

Однако более информативными будут значения, приведённые в таблице 3.1. Из них можно сделать следующие выводы:

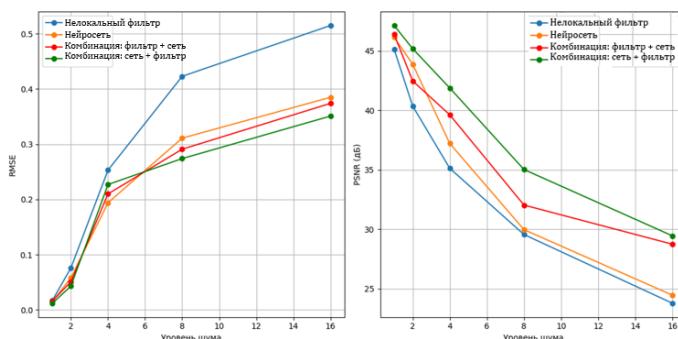
- A. Нелокальный фильтр (NLM) демонстрирует наименьшее время работы, но при этом имеет наибольшие значения RMSE и наименьшие значения PSNR среди всех рассмотренных методов.
- B. Нейросеть (N2N) требует больше времени на обработку, но достигает лучших результатов по сравнению с нелокальным фильтром как по RMSE, так и по PSNR.
- C. Комбинированные методы (фильтр+сеть и сеть+фильтр) показывают лучшие показатели по метрикам RMSE и PSNR в сравнении с нейросетевым подходом. При этом комбинация "сеть+фильтр" демонстрирует наилучшие результаты, чем "фильтр+сеть" как и в случае визуального анализа.

Таблица 3.1

Численный анализ на синтетических данных для  $\lambda = 16$ 

Методы	Время работы, с	RMSE	PSNR, дБ
Нелокальный фильтр, NLM	15	0.51	23.8
Нейросеть, N2N	72	0.38	24.5
Комбинация фильтр+сеть	87	0.37	28.7
Комбинация сеть+фильтр	88	0.35	29.5

Также был проведён анализ работы методов в зависимости от уровня шума  $\lambda \in [1, 2, 4, 8, 16]$ . На рис.3.4 можно увидеть, что лучшими по обоим показателям оказались комбинированные методы, а также что с увеличением уровня шума RMSE всех методов увеличивается, а PSNR соответственно уменьшается. Наименьшие значения RMSE и наибольшие значения PSNR достигаются при комбинации методов "сеть+фильтр" что свидетельствует о большей устойчивости этого подхода к высоким уровням шума.

Рис.3.4. Численный анализ на синтетических данных в зависимости от уровня шума  $\lambda$

Таким образом, комбинированные подходы обеспечивают наилучшее качество денойзинга, несмотря на большее время обработки, что делает их предпочтительными для задач, где качество результата критично.

### *3.1.3.2. Денойзинг реальных данных*

Для проверки работоспособности методов были проведены эксперименты по денойзингу на реальных данных. В этих экспериментах не будут представлены результаты метрик PSNR и RMSE, так как отсутствует точное изображение, с которым можно было бы сравнить обработанные данные.

Помимо этого, был проведен анализ сегмента реального изображения, на котором отсутствует объект. Этот анализ включает вычисление основных статистических характеристик распределения, таких как среднее, дисперсия и среднеквадратическое отклонение (СКО). Проведение статистического анализа для области, содержащей фон, необходимо по нескольким причинам:

- Оценка работоспособности методов денойзинга по удалению шумов на реальных снимках.
- Исследование природы шума и подтверждение его распределения по Пуассону.

Начнём с эксперимента на изображении трубочек размером 40x200x200. На рис.3.5 приведены результаты денойзинга. На основании данных результатов можно сделать следующие выводы:

- A. Нелокальный фильтр справляется хорошо с шумами низкой интенсивности, однако сам объект остался в запущённом состоянии.
- B. Нейросеть успешно обесшумила весь снимок, сделав объект более чётким и очерченным.
- C. Комбинированные методы совместили достоинства обоих подходов. Интенсивности шумов на фоне изображения (где нет объекта) значительно уменьшились, что свидетельствует о высокой эффективности методов в удалении шумов. При этом комбинация "сеть + фильтр" лучше сглаживает шумы и сохраняет детали объекта по сравнению с комбинацией "фильтр + сеть".

Также были посчитаны основные случайные величины в фоновом сегменте на данном снимке. Из полученных результатов в таблице 3.2 можно сделать следующие выводы:

- A. Добавочный шум на реальных снимках распределён по Пуассону, так как среднее и дисперсия практически равны ( $M[x] \approx D[x] = \lambda$ , где  $\lambda$  - возможный уровень шума)
- B. Нелокальный фильтр и нейросетьправляются с задачей, так как все измеряемые статистические величины уменьшаются.
- C. Комбинированные методы показывают улучшенные результаты: среднее значение интенсивности шума уменьшается более чем в 2 раза, а дисперсия уменьшается в 1.5 раза.

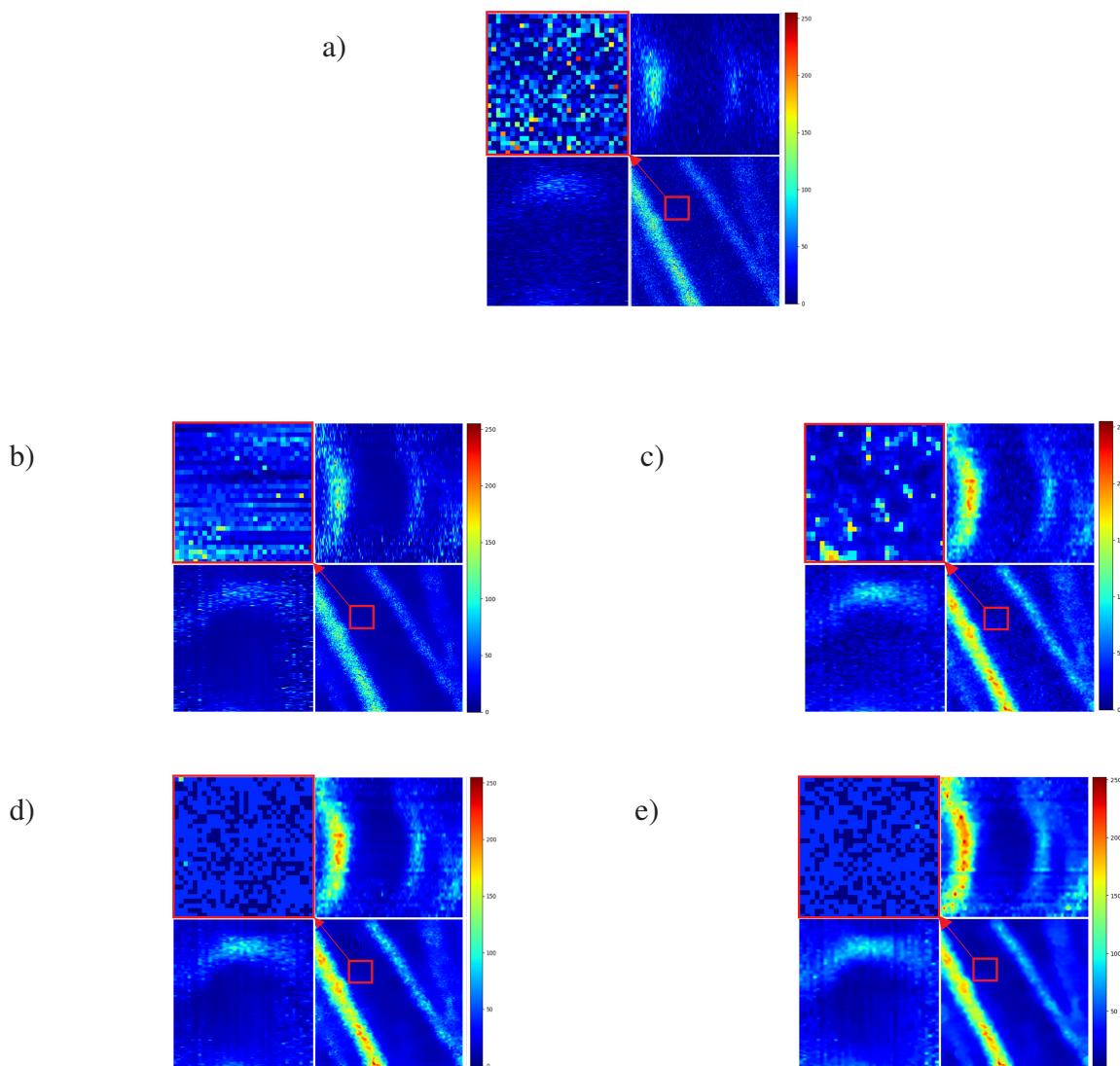


Рис.3.5. Денойзинг реального снимка; Сегмент, не содержащий объекта : *a* — Результат нелокального фильтра; *b* — Результат нейросети; *c* — Результат комбинации фильтр+сеть; *d* —Результат комбинации сеть+фильтр;

Таблица 3.2

## Статистика интенсивности шума на реальном снимке

Методы	Среднее	Дисперсия	СКО
Шумный снимок	8.3	8.4	2.9
Нелокальный фильтр, NLM	6	3.1	1.6
Нейросеть, N2N	5.5	3.3	1.8
Фильтр+сеть	2.3	2.7	1.6
Сеть+фильтр	2.2	2.8	1.7

Также для реальных снимков разных размеров были построены графики интенсивностей вдоль линии, отмеченной красной линией. На основе рис.3.6а и табл. 3.3 можно сказать, что:

- A. Время работы нелокального фильтра и нейросети практически не отличается из-за небольшого размера изображения.
- B. Нейросеть лучше подавляет пики интенсивностей, чем нелокальный фильтр.
- C. Комбинация "сеть+фильтр" даёт наиболее гладкий график интенсивностей, что указывает на более эффективное устранение шумов и сохранение деталей.

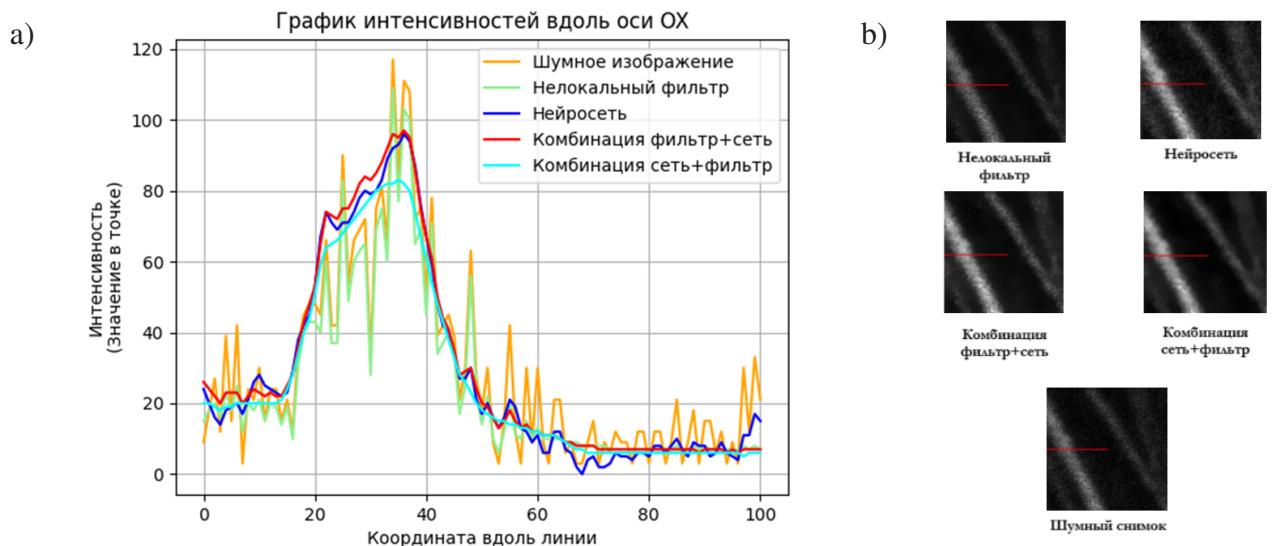


Рис.3.6. График интенсивностей вдоль оси ОХ для реального изображения размером 40x200x200: а — График интенсивностей вдоль оси ОХ; б — Результаты денойзинга

Таблица 3.3

## Статистика интенсивности шума на реальном снимке

	Нелокальный фильтр	Нейросеть	Фильтр+сеть	Сеть+фильтр
Время, сек.	1.5	1.5	3.1	3.2

Для снимка большой размерности ( $29 \times 2048 \times 2048$ ) были также применены различные методы денойзинга. Аналогично, для наглядности, построен график интенсивностей рис.3.7а, проходящий сквозь объект снимка. Также было замерено время и записано в 3.4. Итак можно заметить следующее:

- A. Нелокальный фильтр (NLM) работает значительно быстрее, чем нейросеть и комбинированные методы, но результат денойзинга значительно хуже.
- B. Нейросеть работает дольше, но даёт график интенсивностей с меньшим числом перепадов интенсивностей.
- C. График интенсивностей комбинированных методов показывает, что комбинации дают более гладкий результат, чем отдельно взятые методы, что свидетельствует о лучшем сохранении деталей и снижении шумов.
- D. Из двух комбинаций метод сеть+фильтр дает наиболее гладкий график интенсивностей и лучше сохраняет детали изображения.

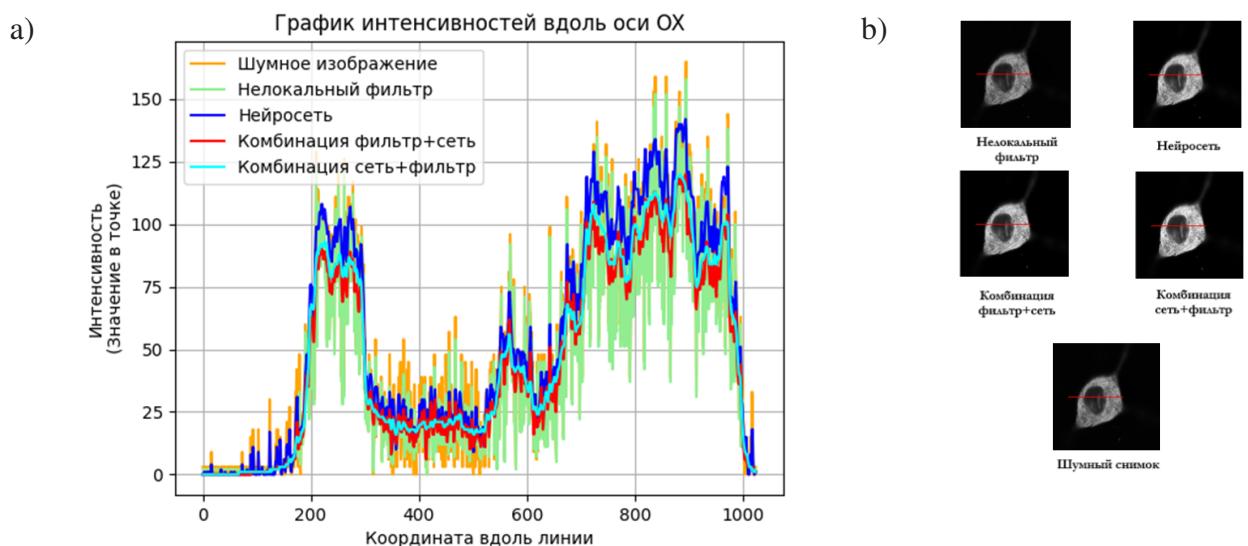


Рис.3.7. График интенсивностей вдоль оси ОХ для реального изображения размером  $29 \times 2048 \times 2048$ : а — График интенсивностей вдоль оси ОХ; б — Результаты денойзинга

Таблица 3.4

Статистика интенсивности шума на реальном снимке

	Нелокальный фильтр	Нейросеть	Фильтр+сеть	Сеть+фильтр
Время	51 сек.	4.6 мин.	5.1 мин.	5.2 мин.

### 3.2. Результаты автоматической сегментации

Результаты показывают, что алгоритм автоматической сегментации сфер имеет значительное преимущество перед ручной сегментацией. На примере изображения размером **36x2048x2048** время выполнения для автосегментации составило всего **0.83 секунды** с усреднением и 0.78 секунды без усреднения, в то время как **ручная сегментация** занимает значительно больше времени, около **10-15 минут**. Эксперименты для 2-ух сегментаций проводились на CPU: **AMD Ryzen 7 5700U with Radeon Graphics 1.80 GHz**.

На рис.3.8 приведено визуальное представление результата автосегментации:

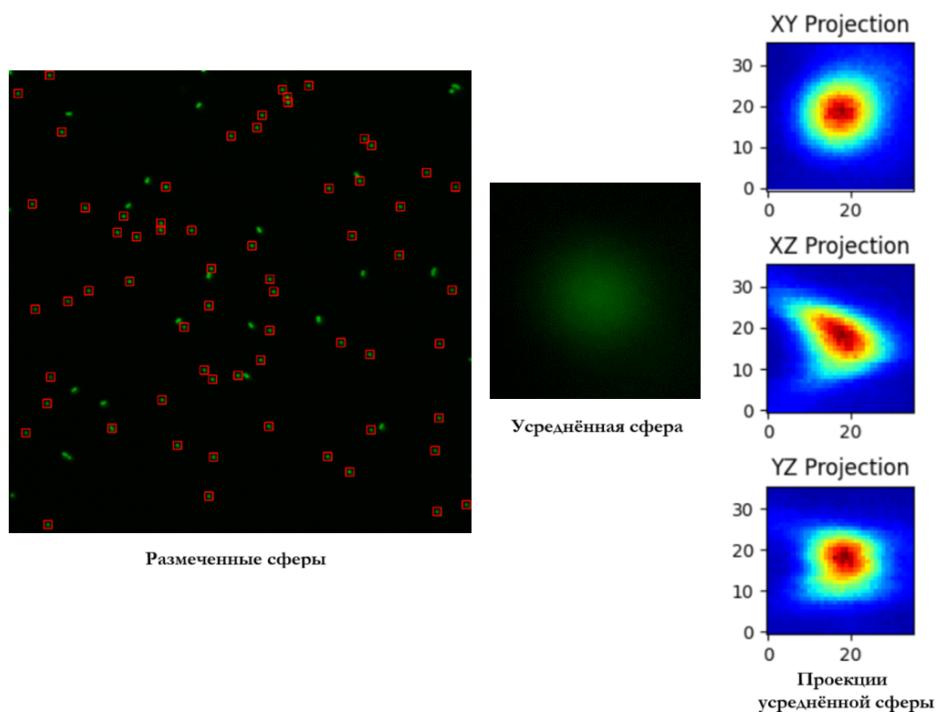


Рис.3.8. Результат автоматической сегментации сфер

Применение денойзинга значительно улучшает качество изображений. На рис.3.9 можно увидеть, что предварительно обработанное изображение со сферами нейронной сетью Noise2Noise, после процесса сегмен-

ции и усреднения даёт более округлую и отчётливую усреднённую сферу.

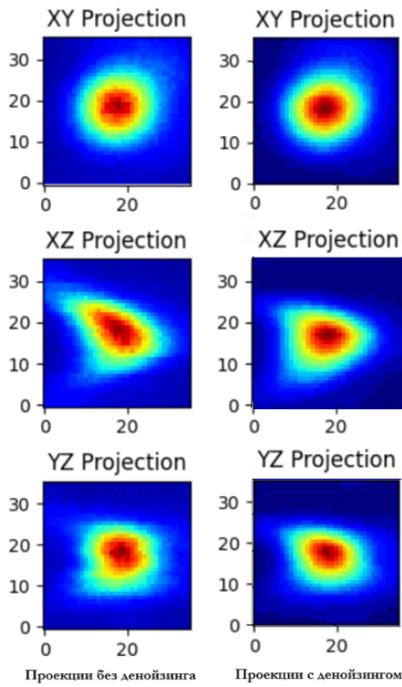


Рис.3.9. Сравнение результатов сегментации сфер с и без денойзинга

Всё выше перечисленное говорит о том, что использование автоматической сегментации сфер оправдано, особенно учитывая сложность и время, затрачиваемое на ручную сегментацию при большом количестве сфер на изображении. Применение предварительного денойзинга дополнительно улучшает точность и качество сегментации, что делает этот подход еще более полезным для обработки изображений с высоким уровнем шума.

### 3.3. Описание архитектуры онлайн-сервиса деконволюции

Результат проделанной работы был оформлен в виде онлайн-сервиса с встроенными процессами обработки изображений с помощью языков программирования Python и JavaScript. Архитектура сервиса включает следующие технологии и инструменты: Django, ReactJS, Docker, Redis, Nginx, Celery, Flower.

Все процессы улучшения изображений могут быть представлены в виде идущими друг за другом шагами обработки: например, загрузка изображений, настройка параметров, обработка, сохранение и визуализация результатов. Потому веб-интерфейс каждой отдельной функциональности представляемого ПО реализован в виде степпера (от англ. step - шаги). Данный степпер даёт возможность пользователю наблюдать промежуточные результаты на каждом шаге обработки

данных, а также на этапах до и после обработки. Кроме того в переходах между разными этапами (степперами) результаты на предыдущем шаге сохраняются в кэше (высокоскоростной памяти небольшого размера), чтобы минимизировать задержки при доступе к данным.

Архитектура серверной части поддерживает масштабируемость, гибкость, высокую производительность, мониторинг и логгирование, а также сервис является надёжным и простым в использовании благодаря контейнеризации Docker. Также сервис обладает модульностью - возможностью интеграции с ПО лаборатории и загрузки изображений в базу данных (БД).

Итак, перейдём к рассмотрению конкретных этапов обработки изображений:

#### A. Сегментация (извлечение) флуоресцентных сфер:

На данном этапе (см. рис.3.10) пользователь может гибко комбинировать два подхода к сегментации: ручной и автоматический. Настройка параметров и использование слайдеров значительно упрощает процесс извлечения образцов для пользователя.

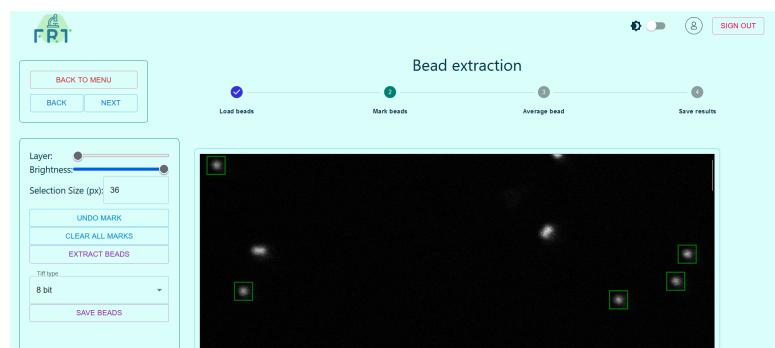


Рис.3.10. Процесс сегментации сфер

#### B. Усреднение сфер:

На данном этапе (см. рис.3.11) есть возможность просмотра выбранных сфер, их усреднение с выбором метода предобработки.



Рис.3.11. Процесс усреднения трёхмерных сфер

### C. Вычисление Функции рассеяния точки (ФРТ):

На 2-ой ступпер (см. рис.3.12) предзагружается усреднённая сфера и пользователь может получить ФРТ экспериментальным способом, применяя метод деконволюции, меняя различные гиперпараметры: размер сферы, число иттераций, параметр регуляризации и вид метода Ричардсона-Люси (РЛ).

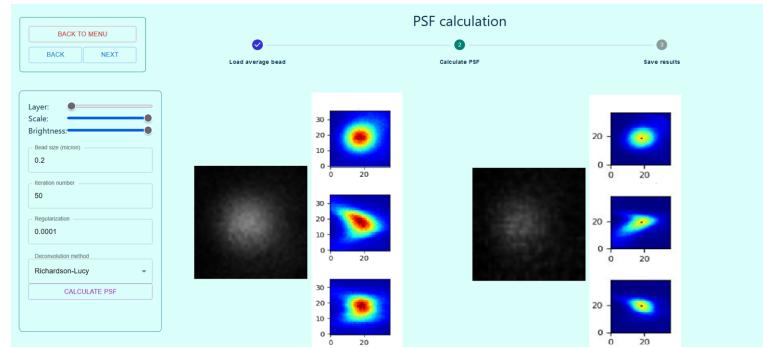


Рис.3.12. Процесс вычисления ФРТ

### D. Деконволюция трёхмерного изображения:

В последнем ступпере (см. ) загружается эФРТ и в интерактивном формате, аналогично вычислению ФРТ, пользователь может обработать трёхмерное изображение в реальном времени. Как можно увидеть на рис.3.13, что результат деконволюции сделал сферы более отчётливыми и отделимыми друг от друга.

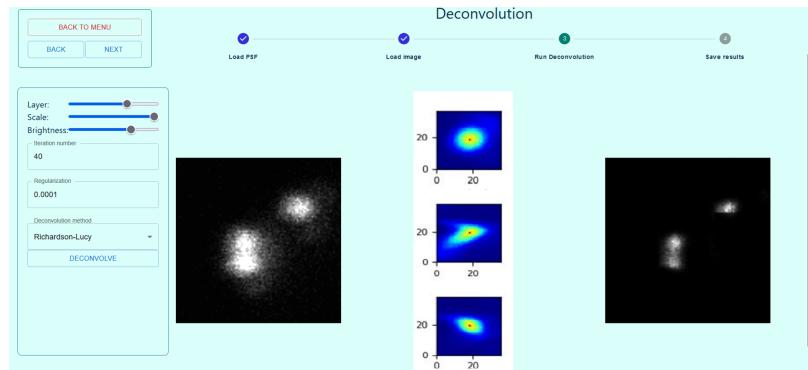


Рис.3.13. Процесс деконволюции изображений

## ЗАКЛЮЧЕНИЕ

В данной работе были рассмотрены существующие методы обесшумливания двухмерных изображений, такие как нелокальный фильтр среднего (Non-local means), а также метод глубокого обучения Noise2Noise. На основе изученного материала был разработан алгоритм 3D денойзинга, включающий в себя эти 2 подхода, а также комбинирующий их. На основе полученных результатов денойзинга можно сделать следующие выводы:

- Нелокальный фильтр демонстрирует высокую скорость работы и эффективно удаляет шумы с низкой интенсивностью.
- Нейросетевой метод позволяет удалять шумы на всём изображении. Метрики качества, такие как PSNR и RMSE, превосходят показатели нелокального фильтра примерно на 15%.
- Комбинирование вышеупомянутых методов позволило добиться улучшения метрик PSNR и RMSE на 10% по сравнению с нейросетевым подходом, а также обеспечивают наиболее гладкий график интенсивностей на выходе.
- Шум, наблюдаемый на реальных снимках приближен к распределению Пуассона.
- Результаты методов на реальных снимках показывают уменьшение дисперсии шума, что также доказывает работоспособность методов шумоподавления.
- Комбинированные методы имеют меньшие значения средних и дисперсий, что также подтверждает улучшения качества работы при комбинации методов.

Также был разработан алгоритм автоматической сегментации флуоресцентных сфер, который позволил ускорить процесс ручной сегментации на 2 порядка.

Все разработанные алгоритмы были успешно интегрированы и развернуты в веб-сервисе, созданном в сотрудничестве с Лабораторией молекулярной нейродегенерации.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**