

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

Работа допущена к защите  
Руководитель образовательной  
программы «Прикладная математика  
и информатика»

\_\_\_\_\_ К.Н. Козлов  
«\_\_\_\_\_» \_\_\_\_\_ 2024 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  
**РАБОТА БАКАЛАВРА**  
**НЕЙРОСЕТЕВЫЕ И КЛАССИЧЕСКИЕ МЕТОДЫ ДЛЯ ЗАДАЧИ**  
**АВТОФОКУСА ЦИФРОВОГО МИКРОСКОПА**

по направлению подготовки 5030102/00201 Прикладная математика и информатика  
Направленность (профиль) 5030102/00201\_02 Системное программирование

Выполнил  
студент гр. 5030102/00201

М.Д. Анищенко

Руководитель  
старший преподаватель ВШПМиВФ,

В.С. Чуканов

Консультант  
по нормоконтролю

Л.А. Арефьева

Санкт-Петербург  
2024

**САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
ПЕТРА ВЕЛИКОГО  
Физико-механический институт**

УТВЕРЖДАЮ

Руководитель образовательной  
программы «Прикладная математика  
и информатика»

\_\_\_\_\_ К.Н. Козлов

« \_\_\_\_\_ » \_\_\_\_\_ 2024г.

**ЗАДАНИЕ  
на выполнение выпускной квалификационной работы**

студенту Анищенко Михаилу Денисовичу гр. 5030102/00201

1. Тема работы: Нейросетевые и классические методы для задачи автофокуса цифрового микроскопа.
2. Срок сдачи студентом законченной работы: июнь 2024.
3. Исходные данные по работе: Набор данных для обучения и тестирования, представленный парами объектов следующих типов: изображение с микроскопа разной степени четкости и вещественное число со знаком, означающее направление и расстояние до фокальной плоскости.

Инструментальные средства:

- Язык программирования python
- Среда разработки Visual Studio Code
- Система контроля версий git

Ключевые источники литературы:

1. Ho C. J., Chan C. C., Chen H. H. AF-Net: A convolutional neural network approach to phase detection autofocus //IEEE Transactions on Image Processing. – 2020. – Т. 29. – С. 6386-6395.
2. Liao J. et al. Deep learning-based single-shot autofocus method for digital microscopy //Biomedical Optics Express. – 2022. – Т. 13. – №. 1. – С. 314-327.
4. Содержание работы (перечень подлежащих разработке вопросов):

- 4.1. Введение. Обоснование актуальности проблемы.
  - 4.2. Постановка задачи.
    - 4.2.1. Основные уравнения, критерии качества.
  - 4.3. Обзор существующих решений.
    - 4.3.1. Классические методы.
    - 4.3.2. Нейросетевые методы.
  - 4.4. Разработка нового нейросетевого решения.
  - 4.5. Эксперименты и тесты.
  - 4.6. Результаты и сравнения с существующими решениями.
  - 4.7. Выводы
  - 4.8. Заключение
5. Дата выдачи задания: 25.01.2024.

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

Задание принял к исполнению

Студент \_\_\_\_\_ М.Д. Анищенко

## **РЕФЕРАТ**

На 38 с., 10 рисунков, 9 таблиц, 2 приложения

**КЛЮЧЕВЫЕ СЛОВА:** АВТОФОКУСИРОВКА, ЭЛЕКТРОННАЯ МИКРОСКОПИЯ, СВЁРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ, АРХИТЕКТУРА СЕТИ.

Тема выпускной квалификационной работы: «Нейросетевые и классические методы для задачи автофокуса цифрового микроскопа».

Данная работа посвящена различным методом автофокусировки электронных микроскопов, а также разработке и реализации нового подхода к этой задаче с применением нейронных сетей и глубокого обучения.

В результате был разработан алгоритм автофокусировки электронного микроскопа, который не зависит от конкретного оборудования и не требует интеграции в программное обеспечение микроскопа или камеры. Алгоритм основан на архитектуре сверточной нейронной сети MobileNetV2. Сеть обучалась на наборе самостоятельно собранных данных, состоящем из снимков с микроскопа и вещественных чисел, обозначающих расстояние до фокальной плоскости с учетом направления.

По результатам тестирования был сделан следующий вывод: представленный алгоритм может быть применим на практике, так как он имеет высокую скорость работы и отличное качество, не уступающее классическим методам, но при этом более устойчив к шуму.

## **ABSTRACT**

38 pages, 10 figures, 9 tables, 2 appendices

**KEYWORDS:** AUTOFOCUS, ELECTRON MICROSCOPY, CONVOLUTIONAL NEURAL NETWORKS, NETWORK ARCHITECTURE.

The subject of the graduate qualification work is «Title of the thesis».

This work is devoted to various methods of autofocusing electron microscopes, as well as the development and implementation of a new approach to this task using neural networks and deep learning.

As a result, an autofocus algorithm for an electron microscope has been developed, which does not depend on specific equipment and does not require integration into the software of the microscope or camera. The algorithm is based on the architecture of

the convolutional neural network MobileNetV2. The network was trained on a set of independently collected data consisting of images from a microscope and real numbers indicating the distance to the focal plane, taking into account the direction.

Based on the test results, the following conclusion was made: the presented algorithm can be applied in practice, since it has a high speed and excellent quality, not inferior to classical methods, but at the same time more resistant to noise.

## СОДЕРЖАНИЕ

|  |    |
|--|----|
| Введение .....   | 8  |
| Глава 1. Постановка задачи .....   | 9  |
| 1.1. Существующие классические методы автофокусировки .....  | 9  |
| 1.1.1. Аналитический метод автофокусировки.....  | 9  |
| 1.1.2. Фазовый автофокус.....  | 11 |
| 1.1.3. Фазовый сдвиг .....   | 13 |
| 1.2. Нейросетевой подход.....  | 14 |
| 1.2.1. Нейросеть AF-Net .....  | 15 |
| 1.3. Сравнение статистического и нейросетевого метода .....  | 15 |
| 1.4. Выводы .....  | 17 |
| Глава 2. Разработка нового нейросетевого подхода .....   | 17 |
| 2.1. Основные требования к разрабатываемому подходу .....  | 17 |
| 2.2. Элементы нейронных сетей .....  | 18 |
| 2.2.1. Сверточный слой.....  | 18 |
| 2.2.2. Функции активации.....  | 19 |
| 2.2.3. SE блоки.....   | 19 |
| 2.3. Архитектура нейросети .....   | 20 |
| 2.4. Определение направления смещения.....   | 25 |
| 2.5. Название параграфа.....   | 25 |
| 2.5.1. Название подпараграфа .....   | 25 |
| 2.6. Название параграфа.....   | 27 |
| 2.7. Выводы .....  | 34 |
| Глава 3. Название третьей главы: разработка программного обеспечения ..  | 35 |
| 3.1. Название параграфа.....   | 35 |
| 3.2. Название параграфа.....   | 35 |
| 3.3. Выводы .....  | 35 |
| Глава 4. Название четвёртой главы. Апробация результатов исследования,<br>а именно: метода, алгоритма, модели исследования ..... | 35 |
| 4.1. Название параграфа.....   | 35 |
| 4.2. Название параграфа.....   | 35 |
| 4.3. Выводы .....  | 35 |
| Заключение .....   | 36 |
| Словарь терминов.....  | 37 |
| Список использованных источников.....  | 38 |

|   |    |
|---|----|
| Приложение 1. Краткие инструкции по настройке издательской системы<br>L <sup>A</sup> T <sub>E</sub> X ..... | 39 |
| Приложение 2. Некоторые дополнительные примеры .....  | 43 |

## ВВЕДЕНИЕ

В цифровой микроскопии применяются различные методы автофокусировки. Однако самый распространенный способ по-прежнему – ручная фокусировка оператором микроскопа. Такой способ очевидно медленный и не самый точный. Большинство автоматических методов неустойчивы к шуму, что сильно влияет на конечный результат.

В последнее время разрабатываются все более качественные и универсальные подходы к автофокусировке микроскопов. Также нельзя не заметить активное использование нейросетей в широком спектре задач. Микроскопия не стала исключением. Но некоторые из них требуют внедрения в программное обеспечение (ПО) камеры.

Задачи, решаемые в рамках данной работы:

- А. Изучение существующих алгоритмов.
- В. Разработка нейросети, решающей задачу автофокусировки микроскопов.
- С. Внедрение разработанного решения в реальную систему.
- Д. Проведение экспериментов и тестов при реальном использовании микроскопа.

Целью этой работы стала разработка нового нейросетевого метода автофокусировки, который бы работал независимо от оборудования.



## ГЛАВА 1. ПОСТАНОВКА ЗАДАЧИ

*Задача фокусировки* – нахождение положения линзы или системы линз относительно матрицы камеры, в котором снимок объекта будет максимально четким, то есть наведение объекта схемки на резкость.

*Автофокусировка* – фокусировка, которая осуществляется алгоритмом без участия человека.

Необходимо разработать новый алгоритм, проводящий автофокусировку, который отличался бы легкостью применения, высокими скоростью работы и метриками качества даже при наличии шума.

### 1.1. Существующие классические методы автофокусировки

Существует множество классических методов автофокусировки. Например, контрастный, фазовый или аналитический метод на основе оценки резкости изображения. Фазовый автофокус (Phase-Detection Autofocus или PDAF) является одним из самых популярных и качественных. Но и он не лишен недостатков, которые будут описаны ниже.

#### 1.1.1. Аналитический метод автофокусировки

Аналитический метод является наиболее простым и примитивным, но из-за этого наименее эффективным. Данный алгоритм работает на основе расчета резкости изображения. Сама же резкость рассчитывается путем нахождения дисперсии лапласиана в каждой точке снимка.

Лапласиан является мерой второй производной и используется для поиска областей с быстрой сменой яркости. Чем выше дисперсия Лапласиана по всему изображению, тем выше «четкость» этого изображения. Аналитически лапласиан определяется формулой

$$L = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (1.1)$$

Однако при работе с изображением, то есть с матрицей чисел, возникает трудность в применении данной формулы, так как нет явной функции, которая описывала бы сделанный снимок. Чтобы решить эту проблему, необходимо более детально рассмотреть, что из себя представляет лапласиан. Лапласиан есть

дивергенция градиента заданной функции:

$$\nabla F = \text{div grad} F \quad (1.2)$$

Приблизительно рассчитать градиент в точке можно с помощью оператора Собеля [*Потом нормально пихнуть в источники* Sobel I. et al. A 3x3 isotropic gradient operator for image processing]. Оператор Собеля представляет собой дискретный дифференциальный оператор, вычисляющий приближённое значение градиента яркости изображения. Данный оператор производит свертку исходного изображения с некоторыми ядрами размера  $3 \times 3$ .

Пусть  $\mathbf{A}$  – исходное изображение,  $\mathbf{G}_x$  и  $\mathbf{G}_y$  – изображения, каждая точка которых является приближенным значением производной по  $x$  и  $y$  соответственно.

$$\mathbf{G}_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A}, \quad \mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{A} \quad (1.3)$$

$$\mathbf{G} = \sqrt{\mathbf{G}_x^2 + \mathbf{G}_y^2} \quad (1.4)$$

Матрица  $\mathbf{G}$  описывает значение градиента в каждом пикселе. Однако для дальнейшего шага потребуются матрицы  $\mathbf{G}_x$ ,  $\mathbf{G}_y$ . Далее нужно посчитать дивергенцию полученного градиента. Для каждой матрицы градиента необходимо посчитать приближенное значение дивергенции по каждой координате. Для каждого пикселя в матрице вычислим его частную производную. Это можно сделать, вычислив разницу между значением этого пикселя и средним значением его соседей.

$$\text{div} f_{i,j} = f_{i,j} - \frac{\sum_{k=-1}^1 \sum_{s=-1}^1 f_{i+k,j+s}}{8}, \quad (k,s) \neq (i,j) \quad (1.5)$$

Получив две матрицы дивергенции, необходимо их сложить, чтобы получить матрицу лапласиана.

$$L = \nabla F = \text{div} \mathbf{G} = \frac{\partial \mathbf{G}_x}{\partial x} + \frac{\partial \mathbf{G}_y}{\partial y} \quad (1.6)$$

Следующим этапом после вычисления матрицы лапласиана является нахождение выборочной дисперсии значений этой матрицы. Именно это значение и будет мерой четкости/резкости изображения.

Для того, чтобы определить положение фокуса, необходимо проделать упомянутые выше шаги для набора снимков и выбрать тот, который имеет наибольшее

по абсолютной величине значение дисперсии лапласиана, так как это будет означать максимальные перепады интенсивности, а значит, наиболее выраженные края и границы объекта съемки.

Очевидно, этот подход имеет много недостатков:

- Малая скорость работы. Алгоритм требует целого набора изображений на вход, на съемку которого затрачивается сравнительно много времени.
- Индивидуальная настройка для каждого объектива/камеры. Необходимо сделать серию снимков, выбрать самый качественный из них и вернуть камеру в это положение. Очевидно, что для объективов с большим коэффициентом увеличения разница положений камеры между двумя соседними снимками должна быть меньше, чем для объективов с малым увеличением. При этом диапазон расстояний для любой конфигурации должен быть примерно одинаковым.
- Невысокая точность. У данного алгоритма она ограничивается шагом сдвига камеры по оси  $z$ , так как истинное положение фокальной плоскости может оказаться между двумя зарегистрированными положениями камеры. Тогда в худшем случае ошибка вычисления составит  $\frac{s}{2}$ , где  $s$  – величина шага по вертикали.

### 1.1.2. Фазовый автофокус

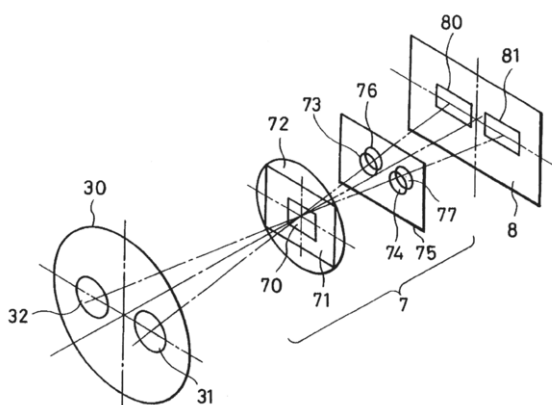


Рис.1.1. Устройство фазового автофокуса. 72 - конденсор; 8 - датчик; 80, 81 - ПЗС-линейки; 30 - зрачок объектива; 31, 32 - области зрачка; 75 - маска; 76, 77 - микрообъективы для датчика; 70 - окно, ограничивающее размер изображения; 71 - рамка конденсора.

Принцип работы фазового автофокуса следующий: в матрицу встроен фазовый датчик (8) с двумя линейками (80, 81), в которые приходят световые потоки с противоположных областей (31 и 32) зрачка (30) объектива через микрообъективы

(76 и 77). Если объект в фокусе, изображения объекта находятся в центрах соседних ПЗС-линеек. Таким образом, сигналы, получаемые процессором с разных линеек, совпадают (находятся «в фазе»). Если же объектив не сфокусирован на объекте съемки, то изображения смещаются внутрь или наружу линеек. Сигналы перестают совпадать и имеют разность фаз (рис.1.2). На основе этой разности фаз считается фазовый сдвиг, а далее – необходимое смещение для системы линз.

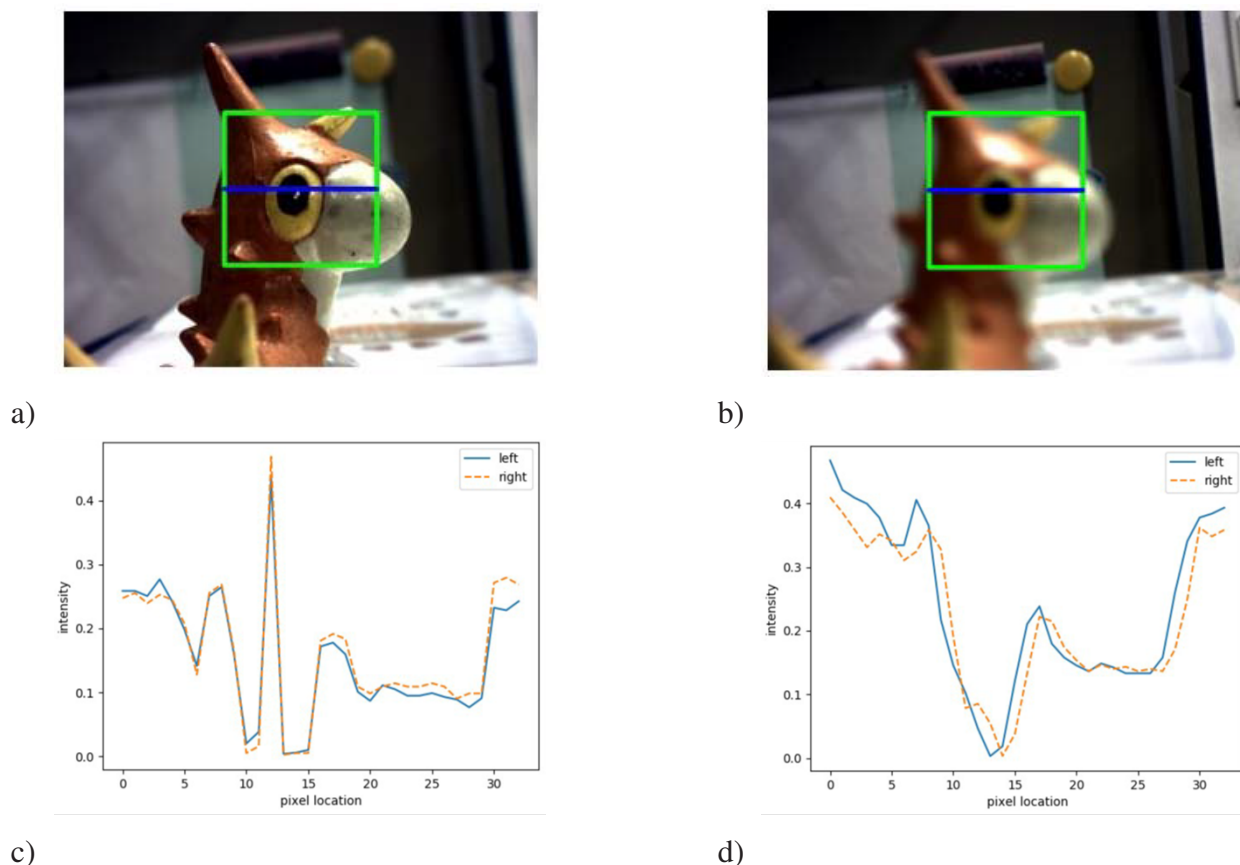


Рис.1.2. Снимки и срезы их фазовых изображений: *a* — снимок объекта в фокусе; *b* — снимок объекта не в фокусе; *c* — срез фазового изображения, когда объект в фокусе; *d* — срез фазового изображения, когда объект не в фокусе

*ПЗС-линейка* – специализированная аналоговая интегральная микросхема, состоящая из светочувствительных фотодиодов, выполненная на основе кремния, использующая технологию ПЗС – приборов с зарядовой связью (полупроводниковых приборов, в которых применяется технология управляемого переноса заряда в объёме полупроводника).

Для PDAF знак фазового сдвига определяет направление движения линзы, а величина фазового сдвига определяет расстояние перемещения линзы. Фазовый сдвиг положителен, если фокальная плоскость находится перед объектом, и отрицателен, если фокальная плоскость находится позади объекта. Чем дальше объект от фокальной плоскости, тем больше абсолютная величина фазового сдвига.

Однако оценка фазового сдвига редко бывает безошибочной. Шум датчика, размытие изображения и низкая текстура часто влияют на точность и надежность оценки фазового сдвига. Хотя ошибку оценки фазового сдвига можно уменьшить, применив фильтр Гаусса к фазовой корреляции, по-прежнему трудно получить точную оценку фазового сдвига, когда объектив находится далеко от сфокусированного положения линзы. Другая заметная проблема PDAF заключается в том, что ошибка оценки фазового сдвига может накапливаться в процессе принятия решения о движении линзы.

Для оценки фазового сдвига в частотной области обычно используется фазовая корреляция. Он применяется для поиска соответствия между смещенными, повернутыми и разномасштабными изображениями. Но этот метод может легко потерпеть неудачу при работе с размытыми или зашумленными изображениями. Чтобы решить эту проблему, было решено применять фильтр Гаусса и использовать разность Гауссианов для извлечения признаков.

В параксиальном и тонколинзовом приближениях формирования изображения глубина объекта  $D$  связана с фазовым сдвигом  $s$  соотношением

$$s = A \left( \frac{1}{z} - \frac{1}{D} \right), \quad (1.7)$$

где  $A$  – некоторая константа,  $z$  – расстояние от фокальной плоскости до линзы. Когда объект находится в фокусе,  $z = D$ . Однако уравнение (1.7) может не выполняться при наличии ошибки оценки фазового сдвига. К этому может приводить ситуация, когда левый и правый фазовые датчики расположены слишком далеко друг от друга.

В качестве решения данной проблемы предлагается использовать статистический метод, который сначала получает распределение вероятностей оптимального расстояния хорда линзы для заданного фазового сдвига на этапе калибровки, а затем использует его для определения расстояния хода линзы при тестировании.

### ***1.1.3. Фазовый сдвиг***

Для идеальной пары левого и правого фазового изображения с фазовым сдвигом  $\Delta x$  выполняется следующее:

$$r(x, y) = l(x + \Delta x, y), \quad (1.8)$$

где  $(x, y)$  – координаты пикселя,  $r(\cdot, \cdot)$  и  $l(\cdot, \cdot)$  – правые и левое фазовые изображения соответственно.

Фазовый сдвиг между левым и правым изображениями можно получить с помощью фазовой корреляции [Павельева Е. А. **Обработка и анализ изображений на основе использования информации о фазе //Компьютерная оптика. – 2018. – Т. 42. – №. 6. – С. 1022-1034.**]. Пусть  $L$  и  $R$  – двумерное преобразование Фурье для левого и правого фазовых изображений соответственно. Для этого найдем корреляционную матрицу  $p(x, y)$  и ее пик.

$$p(x, y) = F^{-1} \left\{ \frac{L \circ \bar{R}}{|L \circ R|} \right\} \quad (1.9)$$

где  $F^{-1}$  – обратное двумерное преобразование Фурье, « $\circ$ » – поэлементное умножение матриц, « $\bar{\cdot}$ » – комплексное сопряжение.

Однако шум от датчиков может приводить к нескольким пикам в корреляционной матрице. Чтобы этого избежать, применим сглаживание корреляционной кривой  $p(x, 0)$  Гауссовым ядром  $g(x)$ :

$$p_f(x) = g(x) * p(x, 0), \quad (1.10)$$

где  $p_f(x)$  – исправленная корреляционная кривая. Тогда фазовый сдвиг  $\Delta x$  определяется как

$$\Delta x = \arg \max_x p_f(x) \quad (1.11)$$

Если построить график фазового сдвига в зависимости от положения линзы, полученная кривая будет называться профилем фазового сдвига. Положение линзы, соответствующее нулевому фазовому сдвигу, является положением линзы в фокусе. Аналогично, если мы построим график контрастности изображения в зависимости от положения линзы, получим профиль фокуса (рис.1.3). На нем положение линзы в фокусе соответствует максимальному значению контрастности.

## 1.2. Нейросетевой подход

Фазовый автофокус показывал отличные результаты по сравнению с более старыми методами автофокусировки. Но упомянутые выше недостатки, в частности неустойчивость к шуму, побуждали искать новые методы работы с оптикой и камерами. Кроме того с ростом интереса к нейросетям и машинному обучению были предприняты попытки использовать их и в этой области. Была разработана сверточная нейросеть на основе алгоритма фазового автофокуса AF-Net.



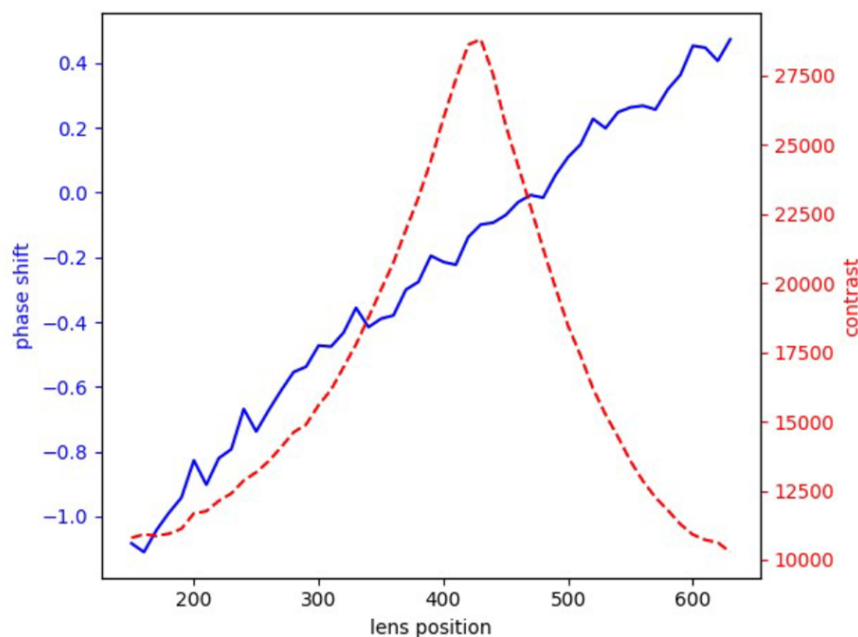


Рис.1.3. Фазовый профиль и профиль фокуса

### 1.2.1. Нейросеть AF-Net

Подход на основе глубокого обучения позволяет уменьшить влияние шума и ошибку, которая может накапливаться при выполнении большого числа операций с фазовыми изображениями.

На вход нейросети поступают те же фазовые изображения (левое и правое), на выходе получается вещественное число со знаком, которое обозначает направление и величину смещения линзы для достижения состояния максимальной четкости. Архитектура сети выглядит следующим образом:

После каждого сверточного слоя следует функция активации ReLU и пакетная нормализация (batch normalization). Используется оптимизатор Adamax, размер батча равен 128, параметры оптимизатора  $\beta_1 = 0.5$ ,  $\beta_2 = 0.999$ , learning rate равен 0.001. Также процесс обучения было принято останавливать, когда потери не уменьшаются в течение 20 эпох. В среднем процесс обучения задействует 80 эпох.

## 1.3. Сравнение статистического и нейросетевого метода

Сравним AF-Net с классическим методом фазового автофокуса, используя следующие метрики: коэффициент успеха (success rate), количество перемещений линзы до остановки процесса и ошибка конечного положения линзы. Сравним на разных начальных положениях линзы. Обозначим нулем положение линзы в

| Выделение признаков |                          |              |                         |                               |
|---------------------|--------------------------|--------------|-------------------------|-------------------------------|
| Названия слоя       | Ядро                     | Шаг ядра     | входные/выходные каналы | входной/выходной размер       |
| conv0               | $5 \times 5$             | $2 \times 2$ | 2/64                    | $33 \times 33 / 16 \times 16$ |
| conv1               | $3 \times 3$             | $1 \times 1$ | 64/128                  | $16 \times 16 / 16 \times 16$ |
| conv2               | $3 \times 3$             | $2 \times 2$ | 128/128                 | $16 \times 16 / 8 \times 8$   |
| conv2               | $3 \times 3$             | $2 \times 2$ | 128/256                 | $8 \times 8 / 4 \times 4$     |
| Полносвязные слои   |                          |              |                         |                               |
| Название слоя       | Входные/выходные размеры |              | Функция активации       |                               |
| fc0                 | 4096/256                 |              | ReLU                    |                               |
| fc1                 | 256/64                   |              | ReLU                    |                               |
| fc2                 | 64/1                     |              | Линейная                |                               |

Таблица 1.1

### Архитектура нейросети

фокусе, положительное смещение - это перемещение линзы ближе к объекту, отрицательное - дальше от объекта, чем положение в фокусе. Результаты:

| Метод          | Метрика                   | начальное положение линзы |       |       |       |       |       |
|----------------|---------------------------|---------------------------|-------|-------|-------|-------|-------|
|                |                           | -30                       | -20   | -10   | 10    | 20    | 30    |
| AF-Net         | Success rate              | 90.48                     | 94.83 | 95.06 | 97.26 | 100.0 | 100.0 |
|                | final lens position error | 1.381                     | 1.207 | 1.185 | 1.027 | 0.981 | 0.826 |
|                | number of lens movements  | 2.26                      | 2.20  | 2.03  | 2.06  | 2.06  | 2.00  |
| Статистический | Success rate              | 23.81                     | 46.55 | 51.85 | 63.01 | 57.41 | 43.48 |
|                | final lens position error | 10.57                     | 5.741 | 4.000 | 3.342 | 3.426 | 3.261 |
|                | number of lens movements  | 3.40                      | 3.00  | 2.21  | 2.21  | 2.54  | 2.60  |

Таблица 1.2

### Сравнение метрик AF-Net и статистического метода

Также сравним их производительность:

| Метод          | Максимальная ошибка положения линзы | Максимальное количество перемещений линзы | Среднее время расчета одного перемещения (мс) |
|----------------|-------------------------------------|---|---|
| AF-Net         | 5                                   | 4   | 10.8  |
| Статистический | 12                                  | 7   | 28.3  |

Таблица 1.3

### Оценка производительности



## 1.4. Выводы

Из сводных сравнительных таблиц хорошо видно, что нейросетевой подход превосходит классический как в метриках качества, так и по производительности. Это означает, что применение AF-Net оправдано, и благодаря ему, фокусировка будет проходить быстрее и точнее. Но автофокусировка, основанная на фазовых изображениях и фазовом сдвиге, имеет существенный недостаток – она требует интеграции непосредственно в камеру, так как предполагается, что такой алгоритм является частью ПО камеры. Он использует показания фазовых датчиков. Но зачастую подключиться к ним извне нет возможности, или же эти датчики отсутствуют вовсе.

## ГЛАВА 2. РАЗРАБОТКА НОВОГО НЕЙРОСЕТЕВОГО ПОДХОДА

Существующие решения, решающие задачу автофокусировки микроскопа, показывают хорошие результаты, но их главный минус в том, что они не являются универсальными. В этой главе будет рассказано о новом подходе на основе машинного обучения, который не требует никаких особых датчиков и интеграции в ПО камеры.

### 2.1. Основные требования к разрабатываемому подходу

Сложность развертывания и применения решений на основе фазовых изображений побудила к разработке нового алгоритма, который не имеет такого количества ограничений и недостатков. Таким образом получаем следующие основные пункты, которые нужно учесть при разработке:

1. На вход должно подаваться изображение с камеры. Алгоритм не должен задействовать какие-либо датчики системы регистрации, которые могут отсутствовать на большинстве видов аппаратуры. Поэтому самым удобным и очевидным входным параметром является изображения.
2. Алгоритм должен принимать на вход ровно один снимок. Если для расчета фокальной плоскости будет использоваться набор изображений, то алгоритм будет работать недостаточно быстро, так как скорее всего потребуется многократное смещение камеры в некоторые промежуточные позиции.

3. Алгоритм должен определять не только величину смещения линзы/камеры, но и его направление. Это также необходимо для достижения максимальной скорости работы.

## 2.2. Элементы нейронных сетей

В данной работе ключевое место имеет подход к задаче автофокуса на основе глубокого обучения. Нейронные сети, которые позволяют работать с изображениями и извлекать различные важные признаки из них, имеют ряд конструктивных особенностей. Далее подробно будут рассмотрены элементы, которые будут применены в разработанном нейросетевом методе автофокусировки и требуют особого внимания.

### 2.2.1. Сверточный слой

*Сверточный слой* – это слой нейронной сети, который выполняет одноименную математическую операцию. Свертка двух функций можно интерпретировать как меру схожести или корреляции этих двух функций. Формально операция свертки определяется следующим образом:

$$s(x) = (f * g)(x) \stackrel{\text{def}}{=} \int_{\mathbb{R}^n} f(y)g(x - y)dy = \int_{\mathbb{R}^n} f(x - y)g(y)dy, \quad (2.1)$$

где  $f, g : \mathbb{R}^n \rightarrow \mathbb{R}$  – функции, интегрируемые в смысле Лебега,  $s : \mathbb{R}^n \rightarrow \mathbb{R}$ .

В случае работы с изображениями вместо функций  $f$  и  $g$  используются многомерные массивы, как правило, двумерные целочисленные. Исходя из этого, стоит переопределить операцию свертки для дискретизированного набора данных следующим образом:

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i - m, j - n)K(m, n), \quad (2.2)$$

где  $I$  – исходное изображение,  $K$  – ядро свертки,  $S$  – карта признаков или карта активации,  $(i, j)$  – её пиксель,  $(m, n)$  – индексы, перебирающие пиксели ядра свертки.

В контексте нейросетей и глубокого обучения вместо операции свертки обычно используется родственная ей операция, которая называется *кросс-корреляцией*. Она симметрична функции свертки и имеет тот же смысл, однако перекрестная

корреляция проще и быстрее реализуется, так как не требует отражения ядра свертки по обеим осям. Кросс-корреляция имеет следующий вид:

$$(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i + m, j + n) K(m, n), \quad (2.3)$$

Если изображение, подаваемое на вход, является многоканальным (например, формата RGB). В таком случае для каждого канала изображения используется свое ядро свертки. Набор таких ядер называется *фильтром свертки*. При этом таких фильтров может быть несколько. Каждое ядро свертки является обучаемым.

*Свертка по глубине* или *depthwise свертка*. [**MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications**]. Этот тип свертки был предложен в 2016 году, и основная его цель – это значительное снижение вычислительной нагрузки, а выполняется он в два этапа. На первом этапе, в отличие от стандартной свертки, в свертке по глубине для каждого канала входного тензора используется свое ядро свертки. Таким образом, фильтр depthwise свертки для цветного трехканального изображения будет состоять из трех ядер и на выходе после первого шага будет также трехканальный тензор. На втором этапе применяется привычная свертка с ядром размера  $1 \times 1$  и с указанным числом фильтров, чтобы собрать полученную информацию по признакам в единое целое.

Благодаря такому подходу, вычислительная сложность алгоритма и количество обучаемых параметров значительно уменьшаются, что очень важно, когда алгоритм применяется в условиях ограниченности времени или мощностей [**тот же источник, стр 2-3**].

*expansion/dilated*

### 2.2.2. Функции активации

В этой работе используется несколько

### 2.2.3. SE блоки

*Squeeze-and-Excitation (SE) блок* – это архитектурное дополнение к сверточным слоям, главная цель которого – улучшение производительности путем отбора признаков, наиболее сильно влияющих на результат. SE блоки перенастраивают веса, усиливая значимые признаки и подавляя менее важные. В предлагаемой нейросети SE блок имеет структуру, изображенную на рис.2.1. Такой блок обрабатывает данные в три шага:

1. Сжатие (Squeeze). На это этапе происходит глобальное объединение по среднему (global average pooling) на каждом канале входного изображения, сжимая пространственные показатели признаков в одно число. Это позволяет извлечь глобальную информацию с о канале.
2. Возбуждение (Excitation). Этот шаг объединяет в себе несколько слоев. Сначала используется полносвязный слой с функцией активации ReLU, зачет еще один полносвязный слой с логистической функцией активации (сигмойдой). Первый слой уменьшает количество каналов с  $C$  до  $C/r$ , где  $r$  – некоторый коэффициент. В большинстве случаев его принято устанавливать равным 16. Второй слой восстанавливает исходное количество каналов. Такой процесс позволяет улавливать и моделировать взаимосвязи между каналами. Кроме того, механизм понижения и восстановления размерности действует как регуляризация и уменьшает возможность переобучения. Стоит отметить, что на практике (например, в библиотеке Pytorch) чаще встречаются реализации SE блоков с использованием свертки с ядром  $1 \times 1$  вместо полносвязных слоев. Это обусловлено тем, что в таком случае сохраняется пространственная структура, что позволяет более естественно и удобно работать с многомерными данными, коими являются изображения и результаты применения сверточных слоев. То есть не приходится выравнивать многомерный массив (такая операция называется *flattening*). А также применения свертки более эффективно с точки зрения вычислений и памяти, так как графические и тензорные процессоры оптимизированы для выполнения именно этой операции, и их ресурсы будут расходоваться экономичнее.
3. Перенастройка весов (Reweight). Весовые коэффициенты, полученные на прошлом этапе, поэлементно умножаются на исходные признаки.

### 2.3. Архитектура нейросети

В основе разработанного решения лежит сверточная нейросеть MobileNet от Google, которая была разработана для классификации объектов на изображении на смартфонах. Существует несколько модификаций данной нейросети. Для решения задачи автофокуса была выбрана версия MobileNetV3\_Small. Одним из ключевых факторов в принятии решения была скорость работы и легковесность, так как

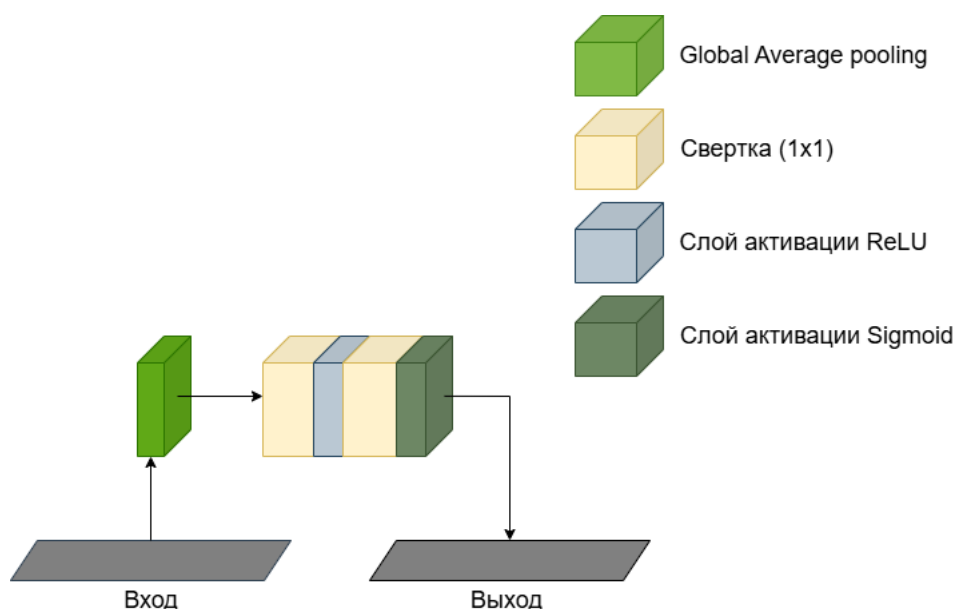


Рис.2.1. Squeeze-and-Excitation блок

принимать решение о смене позиции камеры нужно максимально быстро, не затрачивая больших вычислительных ресурсов.

Главная идея данной сети – так называемые bottleneck-блоки. Они используют depthwise свертку, преимущество которой заключается в снижении вычислительной нагрузки благодаря меньшему количеству операций и меньшему числу обучаемых параметров. В следствие этого уменьшается время обработки входных данных нейросетью, а сам алгоритм становится менее требователен к оборудованию в условиях ограниченных мощности и памяти.

Далее были внесены некоторые изменения в архитектуру MobileNet, чтобы адаптировать ее для решения поставленной задачи. Изменилась сама цель алгоритма: если оригинальная сеть предназначалась для классификации, то модифицированная версия для автофокуса должна решать задачу регрессии, так как ее цель – предсказание непрерывной величины, а именно смещения камеры. В данном случае набором признаков (или факторов) для предсказания данной величины является само входное изображение.

Для того, чтобы адаптировать алгоритм под задачу регрессии, необходимо было внести изменения в последние слои, заменив классификатор на последовательность слоев, выполняющих регрессию. Классификатор состоял из двух полносвязных слоев, функции активации hardswish, слоя исключения (далее dropout или дропаут). Регрессор же состоит из двух полносвязных слоев и функции активации ReLU.

В качестве функции потерь была выбрана smooth L1-loss, которая является комбинацией L1-loss, также известной как Mean Absolute Error (MAE) или средняя абсолютная ошибка, и L2-loss – Mean Squared Error (MSE) или средней-квадратичной ошибкой. Smooth L1-loss задается следующей формулой:

$$L1_{smooth} = \begin{cases} \frac{1}{2\beta}(y - y')^2, & |y - y'| < \beta \\ |y - y'| - \frac{\beta}{2}, & \text{иначе} \end{cases}, \quad (2.4)$$

где  $y$  – истинное искомое значение,  $y'$  – предсказанное значение,  $\beta$  – некоторая константа, которая задается при настройке сети и в данном случае равная единице. Выбранная функция потерь имеет следующие преимущества:

1. Стабильность градиента. Smooth L1-loss малых ошибок ведет себя как MSE, а для больших – как MAE. Константа  $\beta$  задает величину ошибки, на которой происходит это разделение. Но как известно, градиент функции L1-loss не определен в нуле, из-за чего могут возникать трудности при обучении во время обратного распространения ошибки. Вероятность совпадение истинного и предсказанного значений мала, но все же такое случается. Градиент функции средне-квадратичной ошибки лишен этого недостатка, но эта функция потерь вносит слишком большое влияние при больших ошибках. Градиент функции smooth L1-loss выглядит следующим образом:

$$\nabla L1_{smooth} = \begin{cases} \frac{y - y'}{\beta}, & |y - y'| < \beta \\ \text{sign}(y - y'), & \text{иначе} \end{cases} \quad (2.5)$$

Это способствует лучшей сходимости алгоритма обучения.

2. Устойчивость к выбросам. Функция потерь MSE чувствительна к выбросам из-за квадратичной зависимости от ошибки. Smooth L1 при больших отклонениях ведет себя линейно, что позволяет избежать слишком большого влияния ошибок.

Таким образом, Smooth L1-loss является некоторым симбиозом функций абсолютной и среднеквадратичной ошибки, вобрав в себя лучшее от каждой из них.

Описание архитектуры нейросети представлено в табл.2.1. Более детальную схему слоев можно увидеть на рис.2.2.

Также на рис.2.3 представлена структура Bottleneck блока. Данный тип элемента можно считать ключевой инновацией, поскольку в нем используются инвер-

| Вход              | Слой                    | exp size | Выход | SE | AF        | stride |
|-------------------|-------------------------|----------|-------|----|-----------|--------|
| $672^2 \times 3$  | conv2d, $3 \times 3$    | -        | 16    | -  | Hardswish | 2      |
| $336^2 \times 16$ | bneck, $3 \times 3$     | 16       | 16    | ✓  | ReLU      | 2      |
| $168^2 \times 16$ | bneck, $3 \times 3$     | 72       | 24    | -  | ReLU      | 2      |
| $84^2 \times 24$  | bneck, $3 \times 3$     | 88       | 24    | -  | ReLU      | 1      |
| $84^2 \times 24$  | bneck, $5 \times 5$     | 96       | 40    | ✓  | Hardswish | 2      |
| $42^2 \times 40$  | bneck, $5 \times 5$     | 240      | 40    | ✓  | Hardswish | 1      |
| $42^2 \times 40$  | bneck, $5 \times 5$     | 240      | 40    | ✓  | Hardswish | 1      |
| $42^2 \times 40$  | bneck, $5 \times 5$     | 120      | 48    | ✓  | Hardswish | 1      |
| $42^2 \times 48$  | bneck, $5 \times 5$     | 144      | 48    | ✓  | Hardswish | 1      |
| $42^2 \times 48$  | bneck, $5 \times 5$     | 288      | 96    | ✓  | Hardswish | 2      |
| $21^2 \times 96$  | bneck, $5 \times 5$     | 576      | 96    | ✓  | Hardswish | 1      |
| $21^2 \times 96$  | bneck, $5 \times 5$     | 576      | 96    | ✓  | Hardswish | 1      |
| $21^2 \times 96$  | conv2d, $1 \times 1$    | -        | 576   | -  | Hardswish | 1      |
| $21^2 \times 576$ | avgpool2d, $7 \times 7$ | -        | 576   | -  | -         | 1      |
| 576               | fully-connected         | -        | 256   | -  | ReLU      | -      |
| 256               | fully-connected         | -        | 1     | -  | -         | -      |

Таблица 2.1

Структура слоев предлагаемой нейросети

тированные остаточные блоки. Особенность таких bottleneck блоков заключается в следующем:

- Инвертированная структура. В традиционных остаточных блоках используется уменьшение размерности данных перед обработкой. В инвертированных блоках сначала происходит увеличение размерности, затем обработка данных, а потом сжатие обратно к исходной размерности. Это позволяет выявлять высокоуровневые признаки в условиях ограниченности вычислительных ресурсов.
- Свертка по глубине. Как уже говорилось в параграфе 2.2.1, depthwise свертка значительно уменьшает вычислительную сложность и количество обучаемых параметров.
- Squeeze-and-Excitation (SE) блоки. Эти блоки помогают адаптивно перенастраивать весовые коэффициенты каналов, усиливая важные признаки и подавляя менее значимые, что улучшает представление данных и общую производительность модели.
- Функция активации Hardswish. Эта функция проста в вычислительном смысле, помогает нейросети справляться с ленильными зависимостями,



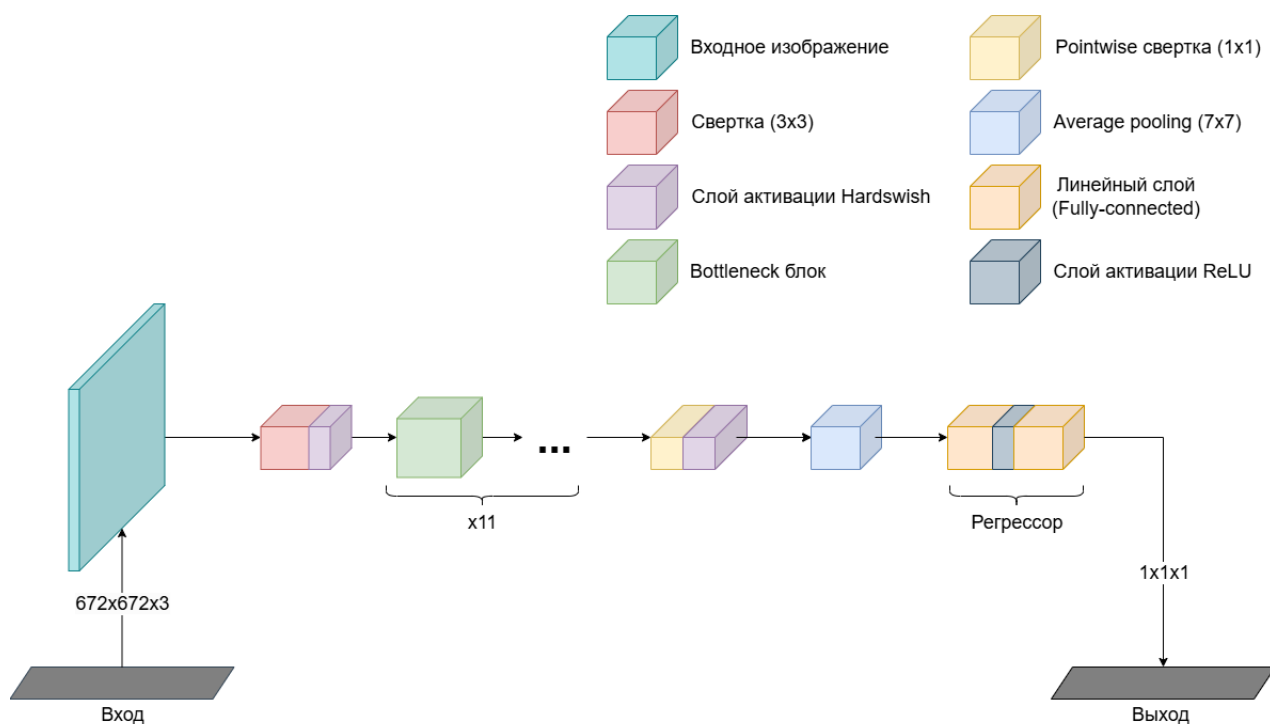


Рис.2.2. Общая схема нейросети

хорошо сохраняет информацию и градиенты при отрицательных значениях ВХОДНЫХ данных.

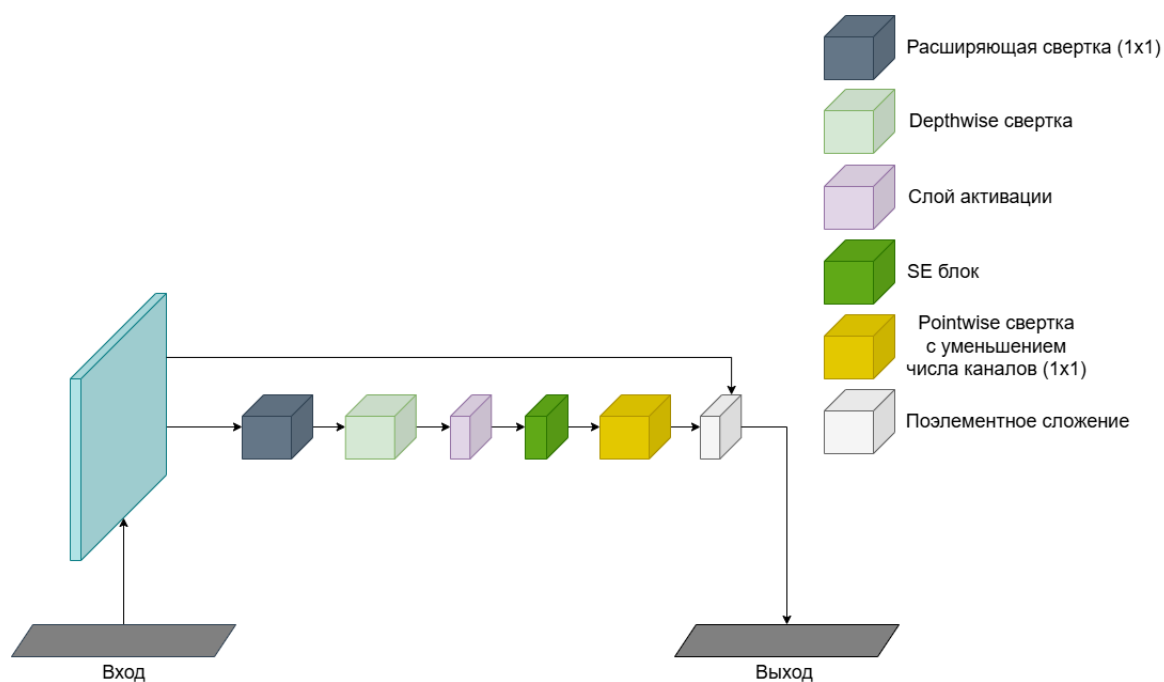


Рис.2.3. Структура bottleneck блока



## 2.4. Определение направления смещения

Попытка смоделировать размытое изображение (например, для обучения нейросети) с помощью простых инструментов таких, как размытие по Гауссу, приводит к тому, что между изображениями по обе стороны от фокальной плоскости, но на равном удалении нет разницы. Соответственно, в таком случае нельзя определить направление смещения камеры от объекта съемки. Однако реальная дефокусировка устроена сложнее, она содержит асимметричные хроматические и сферические aberrации. Изображение, получаемое с микроскопа в силу неидеальности систем регистрации и оптики, а также физики света является сверткой «чистого» изображения и функции рассеяния точки (ФРТ) [Sibarita J. B. *Deconvolution microscopy* // *Microscopy Techniques: -/-*. – 2005. – С. 201-243.]. Помимо этого камера вносит свой шум в получаемое изображение. Таким образом, итоговое изображение можно описать формулой:

$$I' = I \otimes H + N, \quad (2.6)$$

где  $I$  – исходное изображение,  $H$  – ФРТ,  $N$  – шум от системы регистрации,  $I'$  – результирующее изображение.

Благодаря такому механизму, реальные снимки объекта по разные стороны фокальной плоскости будут отличаться. На рис.2.4 приведена оценка асимметрии функции рассеяния точки, смоделированной на разных длинах волн и на разном расстоянии от фокальной плоскости. Хорошо видно, что противоположные друг другу изображения неодинаковые. Именно это позволяет определять направление движения камеры.

## 2.5. Название параграфа

Название параграфа оформляется с помощью команды `\section{...}`, название главы — `\chapter{...}`.

### 2.5.1. Название подпараграфа

Название подпараграфа оформляется с помощью команды `\subsection{...}`.

Использование подпараграфов в основной части крайне не рекомендуется. В случае использования, необходимо вынести данный номер в содержание. Название подпараграфа оформляется с помощью команды `\subsubsection{...}`.

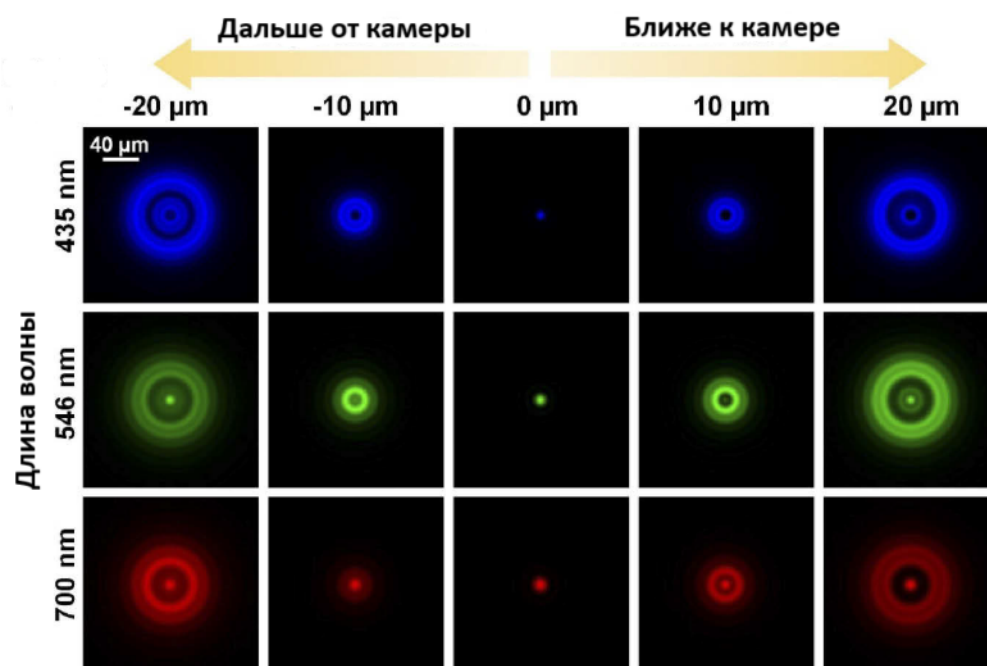


Рис.2.4. Смоделированные ФРТ

Вместо подпараграфов рекомендовано использовать перечисления.

Перечисления могут быть с нумерационной частью и без неё и использоваться с иерархией и без иерархии. Нумерационная часть при этом формируется следующим способом:

1. в перечислениях *без иерархии* оформляется арабскими цифрами с точкой (или длинным тире).
2. В перечислениях *с иерархией* — в последовательности сначала прописных латинских букв с точкой, затем арабских цифр с точкой и далее — строчных латинских букв со скобкой.

Далее приведён пример перечислений с иерархией.

- A. Первый пункт.
- B. Второй пункт.
- C. Третий пункт.
- D. По ГОСТ 2.105–95 [**gost-russian-text-documents**] первый уровень нумерации идёт буквами русского или латинского алфавитов (*для определенности выбираем английский алфавит*), а второй — цифрами.
  1. В данном пункте лежит следующий нумерованный список:
    - a) первый пункт;
    - b) третий уровень нумерации не нормирован ГОСТ 2.105–95 (*для определенности выбираем английский алфавит*);

с) обращаем внимание на строчность букв в этом нумерованном и следующем маркированном списке:

- первый пункт маркированного списка.

Е. Пятый пункт верхнего уровня перечисления.

Маркированный список (без нумерационной части) используется, если нет необходимости ссылки на определенное положение в списке:

- первый пункт с *маленькой буквы* по правилам русского языка;
- второй пункт с *маленькой буквы* по правилам русского языка.

Оформление псевдокода необходимо осуществлять с помощью пакета `algorithm2e` в окружении `algorithm`. Данное окружение интерпретируется в шаблоне как рисунок. Пример оформления псевдокода алгоритма приведён на рис.2.5.

Обратим внимание, что можно сослаться на строку 1 псевдокода из рис.2.5.

## 2.6. Название параграфа

Одиночные формулы также, как и отдельные формулы в составе группы, могут быть размещены в несколько строк. Чтобы выставить номер формулы напротив средней строки, используйте окружение `multlined` из пакета `mathtools` следующим образом [Ganter1999]:

$$\begin{aligned} (A_1, B_1) &\leq (A_2, B_2) \Leftrightarrow \\ &\Leftrightarrow A_1 \subseteq A_2 \Leftrightarrow \\ &\Leftrightarrow B_2 \subseteq B_1. \end{aligned} \tag{2.7}$$

Используя команду `\labelcref{...}` из пакета `cleveref`, допустимо оформить ссылку на несколько формул, например, (???? ??).

Пример оформления четырёх иллюстраций в одном текстово-графическом объекте приведён на рис.2.6. Это возможно благодаря использованию пакета `subcaption`.

Далее можно сослаться на составные части данного рисунка как на самостоятельные объекты: рис.2.6а, рис.2.6b, рис.2.6с, рис.2.6d или на три из четырёх изображений одновременно: рис.2.6а–2.6с.

Приведём пример табличного представления данных с записью продолжения на следующей странице на табл.2.2.

**Algorithm**

**Input:** the many-valued context  $\mathbb{M} \stackrel{\text{def}}{=} (G, M, W, J)$ , the class membership  $\varepsilon : G \rightarrow K$

**Output:** positive and negative binary contexts  $\overline{\mathbb{K}}_+ \stackrel{\text{def}}{=} (\overline{G}_+, M, I_+)$ ,  $\overline{\mathbb{K}}_- \stackrel{\text{def}}{=} (\overline{G}_-, M, I_-)$  such that i-tests found in  $\overline{\mathbb{K}}_+$  are diagnostic tests in  $\mathbb{M}$ , and objects from  $\overline{\mathbb{K}}_-$  are counter-examples

1. **for**  $\forall g_i, g_j \in G$  **do**
2.     **if**  $i < j$  **then**
3.          $\overline{G} \leftarrow (g_i, g_j);$
4.     **for**  $\forall (g_i, g_j) \in \overline{G}$  **do**
5.         **if**  $m(g_i) = m(g_j)$  **then**
6.              $(g_i, g_j)Im;$
7.         **if**  $\varepsilon(g_i) = \varepsilon(g_j)$  **then**
8.              $\overline{G}_+ \leftarrow (g_i, g_j);$
9.         **else**  $\overline{G}_- \leftarrow (g_i, g_j);$
10.      $I_+ = I \cap (\overline{G}_+ \times M), I_- = I \cap (\overline{G}_- \times M);$
11.     **for**  $\forall \overline{g}_+ \in \overline{G}_+, \forall \overline{g}_- \in \overline{G}_-$  **do**
12.         **if**  $\overline{g}_+^\uparrow \subseteq \overline{g}_-^\uparrow$  **then**
13.              $\overline{G}_+ \leftarrow \overline{G}_+ \setminus \overline{g}_+;$

Рис.2.5. Псевдокод алгоритма DiagnosticTestsScalingAndInferring  
[Naidenova2017]

Таблица 2.2

Пример задания данных из [Peskov2004] (с повтором для переноса таблицы на новую страницу)

| $G$   | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $K$ |
|-------|-------|-------|-------|-------|-----|
| 1     | 2     | 3     | 4     | 5     | 6   |
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |
| $g_4$ | 1     | 2     | 1     | 0     | 2   |
| $g_5$ | 1     | 1     | 0     | 1     | 2   |
| $g_6$ | 1     | 1     | 1     | 2     | 2   |
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |

|       |   |   |   |   |   |
|-------|---|---|---|---|---|
| 1     | 2 | 3 | 4 | 5 | 6 |
| $g_4$ | 1 | 2 | 1 | 0 | 2 |
| $g_5$ | 1 | 1 | 0 | 1 | 2 |
| $g_6$ | 1 | 1 | 1 | 2 | 2 |
| $g_1$ | 0 | 1 | 1 | 0 | 1 |
| $g_2$ | 1 | 2 | 0 | 1 | 1 |
| $g_3$ | 0 | 1 | 0 | 1 | 1 |
| $g_4$ | 1 | 2 | 1 | 0 | 2 |
| $g_5$ | 1 | 1 | 0 | 1 | 2 |
| $g_6$ | 1 | 1 | 1 | 2 | 2 |
| $g_1$ | 0 | 1 | 1 | 0 | 1 |
| $g_2$ | 1 | 2 | 0 | 1 | 1 |
| $g_3$ | 0 | 1 | 0 | 1 | 1 |
| $g_4$ | 1 | 2 | 1 | 0 | 2 |
| $g_5$ | 1 | 1 | 0 | 1 | 2 |
| $g_6$ | 1 | 1 | 1 | 2 | 2 |
| $g_1$ | 0 | 1 | 1 | 0 | 1 |
| $g_2$ | 1 | 2 | 0 | 1 | 1 |
| $g_3$ | 0 | 1 | 0 | 1 | 1 |
| $g_4$ | 1 | 2 | 1 | 0 | 2 |
| $g_5$ | 1 | 1 | 0 | 1 | 2 |
| $g_6$ | 1 | 1 | 1 | 2 | 2 |
| $g_1$ | 0 | 1 | 1 | 0 | 1 |
| $g_2$ | 1 | 2 | 0 | 1 | 1 |
| $g_3$ | 0 | 1 | 0 | 1 | 1 |
| $g_4$ | 1 | 2 | 1 | 0 | 2 |
| $g_5$ | 1 | 1 | 0 | 1 | 2 |
| $g_6$ | 1 | 1 | 1 | 2 | 2 |
| $g_1$ | 0 | 1 | 1 | 0 | 1 |
| $g_2$ | 1 | 2 | 0 | 1 | 1 |
| $g_3$ | 0 | 1 | 0 | 1 | 1 |
| $g_4$ | 1 | 2 | 1 | 0 | 2 |
| $g_5$ | 1 | 1 | 0 | 1 | 2 |
| $g_6$ | 1 | 1 | 1 | 2 | 2 |

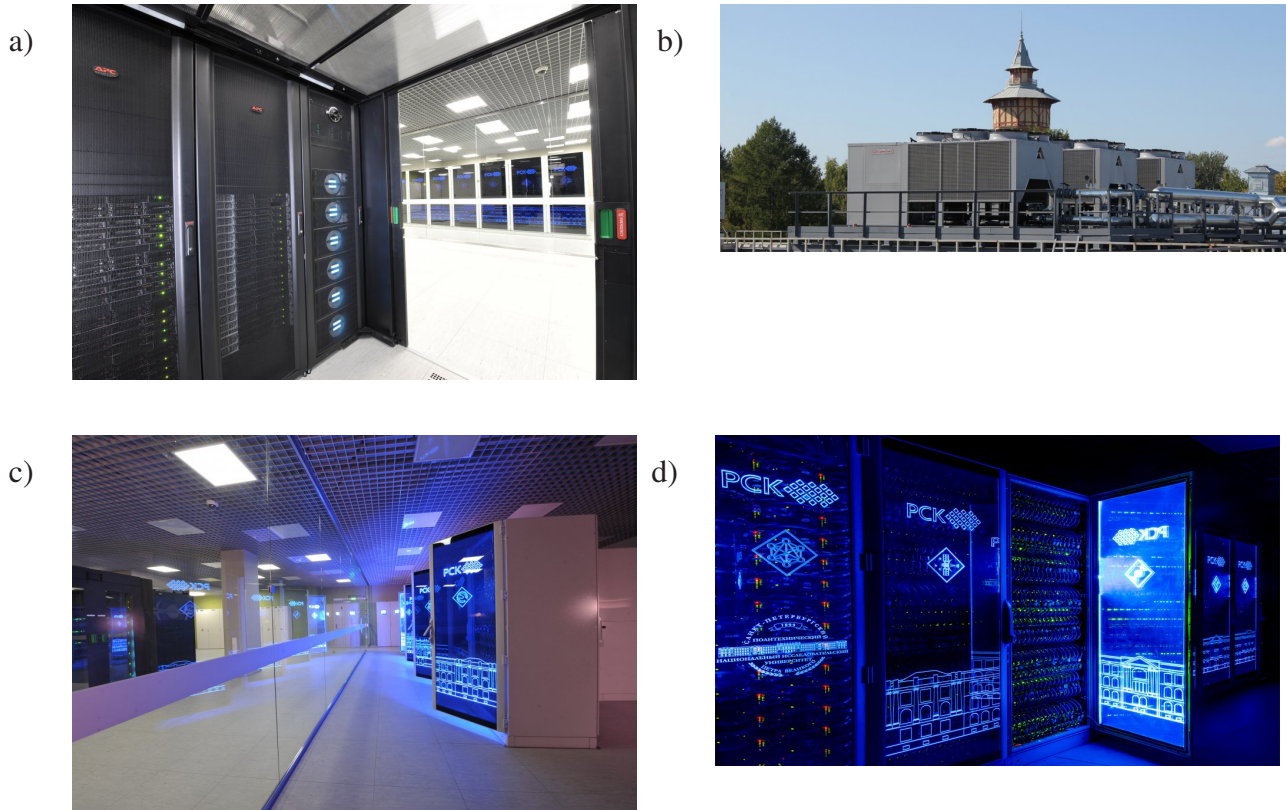


Рис.2.6. Фотографии суперкомпьютерного центра СПбПУ [spbpu-gallery]: *a* — система хранения данных и узлы NUMA-вычислителя; *b* — холодильные машины на крыше научно-исследовательского корпуса; *c* — машинный зал; *d* — элементы вычислительных устройств

Таблица 2.3

Пример представления данных для сквозного примера по ВКР [Peskov2004]

| $G$   | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $K$ |
|-------|-------|-------|-------|-------|-----|
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |
| $g_4$ | 1     | 2     | 1     | 0     | 2   |
| $g_5$ | 1     | 1     | 0     | 1     | 2   |
| $g_6$ | 1     | 1     | 1     | 2     | 2   |

Таблица 2.4

Пример задания данных в табличном виде из [Peskov2004] (с помощью окружения minipage)

| $G$   | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $K$ |
|-------|-------|-------|-------|-------|-----|
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |
| $g_4$ | 1     | 2     | 1     | 0     | 2   |
| $g_5$ | 1     | 1     | 0     | 1     | 2   |
| $g_6$ | 1     | 1     | 1     | 2     | 2   |





Рис.2.7. Новый научно-исследовательский корпус СПбПУ [spbpu-gallery] (с помощью окружения minipage)

Вопросы форматирования текстово-графических объектов (окружений) не регламентированы в известных нам ГОСТах, поэтому предлагаем придерживаться следующих правил:

- **полужирный текст** рекомендуем использовать только для названий стандартных окружений с нумерационной частью, например, для представления *впервые*: **определение 1.1**, **теорема 2.2**, **пример 2.3**, **лемма 4.5**;
- *курсив* рекомендуем использовать только для выделения переменных в формулах, служебной информации об авторах главы (статьи), важных терминов, представляемых по тексту, а также для всего тела окружений, связанных с получением *новых существенных результатов и их доказательством*: теорема, лемма, следствие, утверждение и другие.

По аналогии с нумерацией формул, рисунков и таблиц нумеруются и иные текстово-графические объекты, то есть включаем в нумерацию номер главы, например: теорема 3.1. для первой теоремы третьей главы монографии. Команды  $\text{\LaTeX}$  выставляют нумерацию и форматирование автоматически. Полный перечень команд для подготовки текстово-графических и иных объектов находится в подробных методических рекомендациях [spbpu-bci-template-author-guide].

Для удобства авторов названия стандартных окружений, рекомендованных к использованию, приведены в табл.2.5, а в табл.2.6 перечислены имена специально разработанных окружений для шаблонов SPbPU.

На базе пакета `tikz` разработано большое количество расширений [ctan-tikz], например, `tikzcd`, которые мы рекомендуем использовать для оформления иллюстраций.

Таблица 2.5

## Стандартные окружения

| Название окружения | Назначение   |
|--------------------|--|
| center             | центрирование, аналог команды <code>\centering</code> , но с добавлением нежелательного пробела, поэтому лучше избегать применения <code>center</code>   |
| itemize            | перечисления, в которых нет необходимости нумеровать пункты (немаркированные списки)   |
| enumerate          | перечисления с нумерацией (немаркированные списки)   |
| refsection         | создание отдельных библиографических списков для глав  |
| tabular            | оформление таблиц  |
| table              | автоматическое перемещение по тексту таблиц, оформленных, например, с помощью <code>tabular</code> , для минимизации пустых пространств  |
| longtable          | оформление многостраничных таблиц  |
| tikzpicture        | создание иллюстраций с помощью пакета <code>tikz</code> [ <b>ctan-tikz</b> ]   |
| figure             | автоматическое перемещение по тексту рисунков, оформленных например, с помощью <code>tikz</code> или подключенных с помощью команды <code>\includegraphics</code> , для минимизации пустых пространств |
| subfigure          | оформление вложенных рисунков в составе <code>figure</code>  |
| algorithm          | оформление псевдокода на основе пакета <code>algorithm2e</code> [ <b>ctan-algorithm2e</b> ]  |
| minipage           | оформление рисунков и таблиц без функций автоматического перемещения по тексту для минимизации пустых пространств  |
| equation           | оформление выключенных (не встроенных в текст с помощью <code>\$. . . \$</code> ) одиночных формул на одной строке   |
| multilined         | оформление выключенных (не встроенных в текст с помощью <code>\$. . . \$</code> ) одиночных формул в несколько строк   |
| aligned            | оформление нескольких формул с выравниванием по символу <code>&amp;</code> .   |

В случае, если авторам потребовалось новое окружение, то создать его можно в файле `my_folder/my_settings.tex` согласно правилам, приведённым ниже.

- Для перехода в режим создания окружений следует указать:
  - `\theoremstyle{myplain}` — окружения с доказательствами или аксиомами
  - `\theoremstyle{mydefinition}` — окружения, не связанные с доказательствами или аксиомами.
- В команде создания окружения следует ввести краткий псевдоним (`m-new-env`) и отображаемое в pdf имя окружения (Название\_окружения):
  - `\newtheorem{m-new-env-second}{Название_окружения} - [chapter]`.



Таблица 2.6

## Специальные окружения

| Название окружения | Текстово-графический объект |
|--------------------|-----------------------------|
| abstr              | реферат (abstract)          |
| m-theorem          | теорема                     |
| m-corollary        | следствие                   |
| m-proposition      | утверждение                 |
| m-lemma            | лемма                       |
| m-axiom            | аксиома                     |
| m-example          | пример                      |
| m-definition       | определение                 |
| m-condition        | условие                     |
| m-problem          | проблема                    |
| m-exercise         | упражнение                  |
| m-question         | вопрос                      |
| m-hypothesis       | гипотеза                    |

**Теорема 2.1** (о чем-то конкретном). *Текст теоремы полностью выделен курсивом. Допустимо математические символы не выделять курсивом, если это искажает их значения. Используется абзацный отступ, так как “Абзацы в тексте начинают отступом” в соответствии с ГОСТ 2.105–95. Название теоремы допустимо убрать. Доказательство окончено.*

Доказательство теоремы 2.1, леммы, утверждений, следствий и других подобных окружений (в последнем абзаце) завершаем предложением в котором сказано, что доказательство окончено. Например, доказательство теоремы 2.1 окончено.

Тело доказательства не выделяется курсивом. Тело следующих окружений также не выделяется сплошным курсивом: определение, условие, проблема, пример, упражнение, вопрос, гипотеза и другие.

**Определение 2.1** (термин). В тексте определения только *важные термины* выделяются курсивом. Если определение носит лишь вспомогательный характер, то допустимо не использовать окружение m-definition, представляя текст определения в обычном абзаце. Ключевые термины при этом обязательно выделяются курсивом.

Вместо теоремо-подобных окружений для вставки небольших текстово-графических объектов иногда используются команды. Типичным примером такого

подхода является команда `\footnote{text}`<sup>1</sup>, где в аргументе `text` указывают текст *подстрочной ссылки (сноски)*. В них *нельзя добавлять веб-ссылки или цитировать литературу*. Для этих целей используется список литературы. Нумерация сносок сквозная по ВКР без точки на конце выставляется в шаблоне автоматически, однако в каждом приложении к ВКР нумерация, зависящая от номера приложения, выставляется префикс «П», например «П1.1» — первая сноска первого приложения.

## 2.7. Выводы

Текст заключения ко второй главе. Пример ссылок [**Article; Book; Booklet; Conference; Inbook; Incollection; Manual; Mastersthesis; Misc; Phdthesis; Proceedings; Techreport; Unpublished; badiou:briefings**], а также ссылок с указанием страниц, на котором отображены те или иные текстово-графические объекты [**Naidenova2017**] или в виде мультицитаты на несколько источников [**Naidenova2017; Ganter1999**]. Часть библиографических записей носит иллюстративный характер и не имеет отношения к реальной литературе.

Короткое имя каждого библиографического источника содержится в специальном файле `my_biblio.bib`, расположенном в папке `my_folder`. Там же находятся исходные данные, которые с помощью программы `Viber` и стилевого файла `Biblatex-GOST` [**ctan-biblatex-gost**] приведены в списке использованных источников согласно ГОСТ 7.0.5-2008. Многообразные реальные примеры исходных библиографических данных можно посмотреть по ссылке [**ctan-biblatex-gost-examples**].

Как правило, ВКР должна состоять из четырех глав. Оставшиеся главы можно создать по образцу первых двух и подключить с помощью команды `\input` к исходному коду ВКР. Далее в приложении 1 приведены краткие инструкции запуска исходного кода ВКР [**latex-miktex; latex-texstudio**].

В приложении 2 приведено подключение некоторых текстово-графических объектов. Они оформляются по приведенным ранее правилам. В качестве номера структурного элемента вместо номера главы используется «П» с номером главы. Текстово-графические объекты из приложений не учитываются в реферате.

---

<sup>1</sup>Внимание! Команда вставляется непосредственно после слова, куда вставляется сноска (без пробела). Лишние пробелы также не указываются внутри команды перед и после фигурных скобок.

### **ГЛАВА 3. НАЗВАНИЕ ТРЕТЬЕЙ ГЛАВЫ: РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Хорошим стилем является наличие введения к главе. Во введении может быть описана цель написания главы, а также приведена краткая структура главы.

#### **3.1. Название параграфа**

#### **3.2. Название параграфа**

#### **3.3. Выводы**

Текст выводов по главе 3.

### **ГЛАВА 4. НАЗВАНИЕ ЧЕТВЁРТОЙ ГЛАВЫ. АПРОБАЦИЯ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ, А ИМЕННО: МЕТОДА, АЛГОРИТМА, МОДЕЛИ ИССЛЕДОВАНИЯ**

Хорошим стилем является наличие введения к главе. Во введении может быть описана цель написания главы, а также приведена краткая структура главы.

#### **4.1. Название параграфа**

#### **4.2. Название параграфа**

Пример ссылки на литературу [avtonomova:fya; Peskov2004-ru; Kotelnikov2004-ru; Kotelnikov2004].

#### **4.3. Выводы**

Текст выводов по главе 4.

## ЗАКЛЮЧЕНИЕ

Заключение (2 – 5 страниц) обязательно содержит выводы по теме работы, *конкретные предложения и рекомендации* по исследуемым вопросам. Количество общих выводов должно вытекать из количества задач, сформулированных во введении выпускной квалификационной работы.

Предложения и рекомендации должны быть органически увязаны с выводами и направлены на улучшение функционирования исследуемого объекта. При разработке предложений и рекомендаций обращается внимание на их обоснованность, реальность и практическую приемлемость.

Заключение не должно содержать новой информации, положений, выводов и т. д., которые до этого не рассматривались в выпускной квалификационной работе. Рекомендуются писать заключение в виде тезисов.

Последним абзацем в заключении можно выразить благодарность всем людям, которые помогали автору в написании ВКР.

## СЛОВАРЬ ТЕРМИНОВ

**TeX** — язык вёрстки текста и издательская система, разработанные Дональдом Кнутом.

**LaTeX** — язык вёрстки текста и издательская система, разработанные Лэсли Лампортом как надстройка над TeX.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

## Приложение 1

### Краткие инструкции по настройке издательской системы L<sup>A</sup>T<sub>E</sub>X

В SPbPU-BCI-template автоматически выставляются необходимые настройки и в исходном тексте шаблона приведены примеры оформления текстово-графических объектов, поэтому авторам достаточно заполнить имеющийся шаблон текстом главы (статьи), не вдаваясь в детали оформления, описанные далее. Возможный «быстрый старт» оформления главы (статьи) под Windows следующий<sup>П1.1</sup>:

- A. Установка полной версии MikTeX [latex-miktex]. В процессе установки лучше выставить параметр доустановки пакетов «на лету».
- B. Установка TexStudio [latex-texstudio].
- C. Запуск TexStudio и компиляция my\_chapter.tex с помощью команды «Build&View» (например, с помощью двойной зелёной стрелки в верхней панели). Иногда, для достижения нужного результата необходимо несколько раз скомпилировать документ.
- D. В случае, если не отобразилась библиография, можно
  - воспользоваться командой Tools → Commands → Biber, затем запустив Build&View;
  - настроить автоматическое включение библиографии в настройках Options → Configure TexStudio → Build → Build&View (оставить по умолчанию, если сборка происходит слишком долго): txs:///pdflatex | txs:///biber | txs:///pdflatex | txs:///pdflatex | txs:///view-pdf.

В случае возникновения ошибок, попробуйте скомпилировать документ до последних действий или внимательно ознакомьтесь с описанием проблемы в log-файле. Бывает полезным переход (по подсказке TexStudio) в нужную строку в pdf-файле или запрос с текстом ошибки в поисковиках. Наиболее вероятной проблемой при первой компиляции может быть отсутствие какого-либо установленного пакета L<sup>A</sup>T<sub>E</sub>X.

В случае корректной работы настройки «установка на лету» все дополнительные пакеты будут скачиваться и устанавливаться в автоматическом режиме. Если доустановка пакетов осуществляется медленно (несколько пакетов за один запуск

<sup>П1.1</sup> Вниманию! Пример оформления подстрочной ссылки (сноски).

компилятора), то можно попробовать установить их в ручном режиме следующим образом:

1. Запустите программу: меню → все программы → MikTeX → Maintenance (Admin) → MiKTeX Package Manager (Admin).
2. Пользуясь поиском, убедитесь, что нужный пакет присутствует, но не установлен (если пакет отсутствует воспользуйтесь сначала MiKTeX Update (Admin)).
3. Выделив строку с пакетом (возможно выбрать несколько или вообще все неустановленные пакеты), выполните установку Tools → Install или с помощью контекстного меню.
4. После завершения установки запустите программу MiKTeX Settings (Admin).
5. Обновите базу данных имен файлов Refresh FNDB.

Для проверки текста статьи на русском языке полезно также воспользоваться настройками Options → Configure TexStudio → Language Checking → Default Language. Если русский язык «ru\_RU» не будет доступен в меню выбора, то необходимо вначале выполнить Import Dictionary, скачав из интернета любой русскоязычный словарь.

Далее приведены формулы (П1.2), (П1.1), рис.П1.2, рис.П1.1, табл.П1.2, табл.П1.1.

$$\pi \approx 3,141. \quad (\text{П1.1})$$



Рис.П1.1. Вид на гидробашню СПбПУ [spbpu-gallery]



Представление данных для сквозного примера по ВКР [Peskov2004]

| $G$   | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $K$ |
|-------|-------|-------|-------|-------|-----|
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |
| $g_4$ | 1     | 2     | 1     | 0     | 2   |
| $g_5$ | 1     | 1     | 0     | 1     | 2   |
| $g_6$ | 1     | 1     | 1     | 2     | 2   |

## П1.1. Параграф приложения

### П1.1.1. Название подпараграфа

Название подпараграфа оформляется с помощью команды `\subsection{...}`.  
Использование подпараграфов в основной части крайне не рекомендуется.

#### П1.1.1.1. Название подподпараграфа

$$\pi \approx 3,141. \quad (\text{П1.2})$$



Рис.П1.2. Вид на гидробашню СПбПУ [spbpu-gallery]

Представление данных для сквозного примера по ВКР [Peskov2004]

| $G$   | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $K$ |
|-------|-------|-------|-------|-------|-----|
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |
| $g_4$ | 1     | 2     | 1     | 0     | 2   |
| $g_5$ | 1     | 1     | 0     | 1     | 2   |
| $g_6$ | 1     | 1     | 1     | 2     | 2   |

Приложение 2

Некоторые дополнительные примеры

В приложении<sup>П2.1</sup> приведены формулы (П2.2), (П2.1), рис.П2.2, рис.П2.1, табл.П2.2, табл.П2.1

$$\pi \approx 3,141.$$

(П2.1)



Рис.П2.1. Вид на гидробашню СПбПУ [spbpu-gallery]

Таблица П2.1

Представление данных для сквозного примера по ВКР [Peskov2004]

| $G$   | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $K$ |
|-------|-------|-------|-------|-------|-----|
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |
| $g_4$ | 1     | 2     | 1     | 0     | 2   |
| $g_5$ | 1     | 1     | 0     | 1     | 2   |
| $g_6$ | 1     | 1     | 1     | 2     | 2   |

<sup>П2.1</sup>Внимание! Пример оформления подстрочной ссылки (сноски).

## П2.1. Подраздел приложения

$$\pi \approx 3,141. \quad (\text{П2.2})$$



Рис.П2.2. Вид на гидробашню СПбПУ [spbpu-gallery]

Таблица П2.2

Представление данных для сквозного примера по ВКР [Peskov2004]

| $G$   | $m_1$ | $m_2$ | $m_3$ | $m_4$ | $K$ |
|-------|-------|-------|-------|-------|-----|
| $g_1$ | 0     | 1     | 1     | 0     | 1   |
| $g_2$ | 1     | 2     | 0     | 1     | 1   |
| $g_3$ | 0     | 1     | 0     | 1     | 1   |
| $g_4$ | 1     | 2     | 1     | 0     | 2   |
| $g_5$ | 1     | 1     | 0     | 1     | 2   |
| $g_6$ | 1     | 1     | 1     | 2     | 2   |