

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ «МИФИ»
(НИЯУ МИФИ)

ИНСТИТУТ ЯДЕРНОЙ ФИЗИКИ И ТЕХНОЛОГИЙ
КАФЕДРА №40 «ФИЗИКА ЭЛЕМЕНТАРНЫХ ЧАСТИЦ»

На правах рукописи

ИЛЬЯСОВ АЙДАР ИРШАТОВИЧ

**ПРИМЕНЕНИЕ СВЁРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ
РЕКОНСТРУКЦИИ ПОЛОЖЕНИЯ СОБЫТИЯ В
ЭКСПЕРИМЕНТЕ DEAR-3600**

Направление подготовки 14.04.02 «Ядерная физика и технологии»
Диссертация на соискание степени магистра

Научный руководитель,
к.ф.-м.н.

_____ А. В. Гробов

Москва 2021

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТРА

**ПРИМЕНЕНИЕ СВЁРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ
РЕКОНСТРУКЦИИ ПОЛОЖЕНИЯ СОБЫТИЯ В
ЭКСПЕРИМЕНТЕ DEAR-3600**

Студент	_____ А. И. Ильясов
Научный руководитель, к.ф.-м.н.	_____ А. В. Гробов
Рецензент, к.ф.-м.н.	_____ А. С. Чепурнов
Секретарь ГЭК, к.ф.-м.н.	_____ А. А. Кириллов
Зав. каф. №40, д.ф.-м.н., проф.	_____ М. Д. Скорохватов
Рук. учеб. прог., д.ф.-м.н., проф.	_____ М. Д. Скорохватов

ОГЛАВЛЕНИЕ

Введение	4
Скрытая масса Вселенной	4
Детекторы по поиску темной материи	7
Машинное обучение в физике частиц	9
1 DEAP-3600	12
1.1 Описание детектора	12
1.2 Алгоритмы реконструкции положения события	16
1.2.1 Метод MBLikelihood	16
1.2.2 Метод TimeFit2	17
1.2.3 Преимущества и недостатки алгоритмов, цель работы . . .	18
2 Исследовательская работа	21
2.1 Используемые алгоритмы реконструкции	21
2.1.1 Полносвязная нейронная сеть	21
2.1.2 Нейронная сеть с короткими связями	22
2.1.3 Нейронная сеть со сверточными слоями	23
2.2 Ход работы	24
2.3 Результаты	27
Заключение	39
Список литературы	41
Приложение А Построение моделей нейронной сети с помощью языка программирования Python	46

ВВЕДЕНИЕ

СКРЫТАЯ МАССА ВСЕЛЕННОЙ

Предсказания Стандартной модели (СМ) были подтверждены множеством экспериментов. Однако существуют следующие проблемы вне рамок СМ:

- Масса и осцилляции нейтрино;
- Барионная асимметрия Вселенной;
- Наличие скрытой массы и темной энергии во Вселенной;
- Инфляция.

Проблема скрытой массы рассматривается в данной работе. Существует несколько свидетельств ее существования. Фриц Цвикки в работе [1] на основе наблюдений относительных скоростей галактик в скоплении Волос Вероники получил, что наблюдаемая масса скопления, которая была определена по суммарным светимостям галактик и их красному смещению, оказалась в 500 раз меньше массы скопления, рассчитанной исходя из собственных скоростей отдельных частей скопления в соответствии с теоремой вириала: для всякого гравитационно связанного тела его средняя по времени потенциальная гравитационная энергия U_G связана со средней по времени кинетической энергией движения частиц тела ε_{kin} следующим образом:

$$U_G = -2\varepsilon_{kin}. \quad (1)$$

Поскольку гравитационная энергия системы $U_G \approx \frac{-GM^2}{R}$ (где M – масса системы, R – радиус системы, G – гравитационная постоянная), а средняя кинетическая энергия системы $\varepsilon_{kin} = \frac{M\bar{v}^2}{2}$ (где $\bar{v}^2$ – среднеквадратичная скорость частей системы), то из уравнения 1 получаем:

$$M \approx \frac{\bar{v}^2 R}{G}.$$

Таким образом, зная $\bar{v}^2$ и R можно оценить массу системы [2].

Скорость вращения отдельных частей системы вокруг центра системы определяется стабильностью орбит:

$$\frac{GmM}{r^2} = \frac{mv^2}{r},$$

где r – радиус орбиты, m – масса части системы. Тогда для скорости $v(r)$ получаем следующую зависимость:

$$v(r) = \sqrt{\frac{GM}{r}}.$$

В приближении сферичности центральной области системы получаем, что

$$M = \bar{\rho} \frac{4}{3} \pi r^3,$$

где $\bar{\rho}$ – средняя плотность вещества. Тогда приближенно получаем линейный рост скорости вращения частей системы при увеличении расстояния от центра системы:

$$v(r) \sim \sqrt{\frac{M}{r}} \sim r.$$

Во внешней области массу M можно считать постоянной, тогда:

$$v(r) \sim \frac{1}{\sqrt{r}}.$$

Наблюдения показали, что видимая масса была слишком мала для того, чтобы удерживать части галактики от разлетания. Ротационные кривые во внешней области имеют другое поведение, нежели предсказанное теоретическими расчетами (рисунок 1).

Другим подтверждением существования скрытой массы стало гравитационное линзирование (рисунок 2). В данном методе наблюдения скрытой массы используется тот факт, что она обладает гравитационным полем и является гравитационной линзой — массивным телом, изменяющим своим гравитационным полем направление распространения электромагнитного излучения. Из общей теории относительности известно, что путь лучей света в таком поле искривляется. Исследуя искривленные изображения скоплений галактик, можно восстановить их истинные изображения и оценить распределение массы в

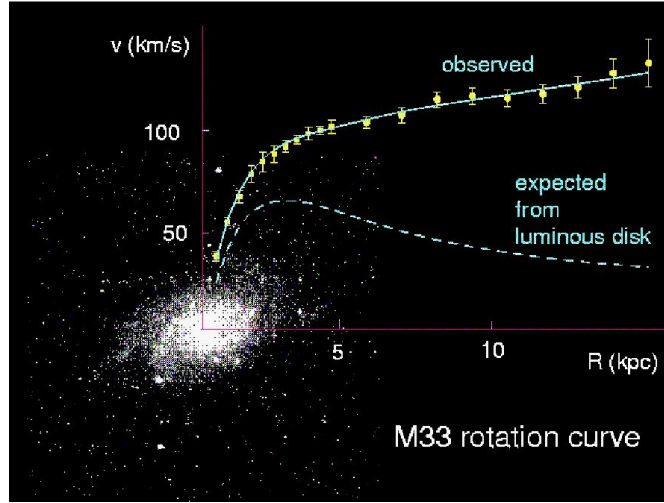


Рисунок 1 — Ожидаемая и наблюдаемая галактическая ротационная кривая для спиральной галактики М33

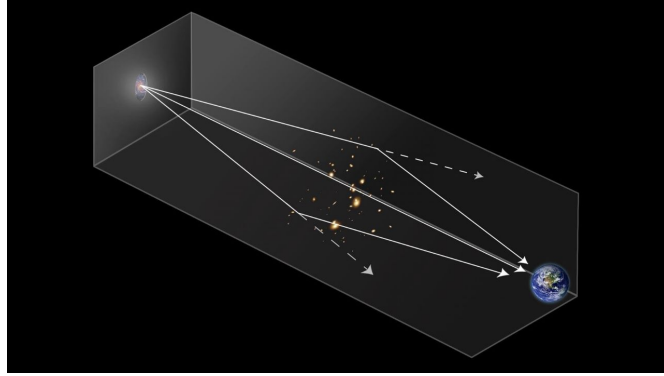


Рисунок 2 — Схема распространения электромагнитного излучения галактики через гравитационную линзу

гравитационной линзе — скоплении галактик. Именно при таких исследованиях получается дополнительный вывод о необходимости существования скрытой массы в скоплениях галактик [3].

Последние результаты эксперимента PLANCK говорят о том, что Вселенная состоит на $\Omega_\Lambda = (68.47 \pm 0.73)\%$ из темной энергии и на $\Omega_m = (31.47 \pm 0.24) \%$ из нерелятивистского вещества, из которых $\Omega_c = (26.42 \pm 0.26) \%$ — темная материя (скрытая масса) и $\Omega_b = (4.92 \pm 0.03) \%$ — барионное вещество [4]. Отметим, что темная материя, по всей видимости, состоит из массивных частиц, которые отсутствуют в Стандартной модели физики частиц. Эти частицы — нерелятивистские в настоящий момент времени, почти не взаимодействуют между собой (иначе это привело бы к формированию в основном шарообразных гало, а это противоречит наблюдениям) и не взаимодействуют с фотонами [5].

Сделав предположение о том, что частицы темной материи X находились

в термодинамическом равновесии с обычным веществом в ранней Вселенной, можно принять, что эти частицы вышли из равновесия в какой-то момент времени и с тех пор распространяются свободно. Сравнивая температуру заковки $T_{\text{freeze-out}}$ с массой темной материи m_χ , выделяют три модели [5]:

- холодная темная материя: $T_f < m_\chi$, основной кандидат — слабовзаимодействующие массивные частицы - *WIMP*;
- горячая темная материя: $T_f > m_\chi$, $m_\chi < 1$ эВ, основной кандидат — правые нейтрино;
- теплая темная материя: $T_f > m_\chi$, $m_\chi > 1$ эВ, основной кандидат — стерильное нейтрино.

Наиболее приемлемым вариантом считается холодная темная материя (cold dark matter — CDM), наиболее перспективными кандидатами являются слабовзаимодействующие массивные частицы — WIMP, а соответствующая такой материи пространственно-плоская модель Вселенной называется Λ CDM модель [5].

ЭКСПЕРИМЕНТЫ ПО ПОИСКУ СКРЫТОЙ МАССЫ

Поиски WIMP ведутся на многих экспериментах в течении последних 20 лет, их можно разделить на три группы: коллайдерные, косвенные и прямые.

Если считать, что тепловое "отцепление" (thermal decoupling) является механизмом, который обеспечивает изобилие темной материи в ранней Вселенной, то ожидается, что WIMP-частицы тоже аннигилируют и производят частицы Стандартной модели, включая гамма-кванты, нейтроны, электроны, позитроны, протоны, антипротоны, дейтроны и антидейтроны. Измерения доли этих частиц СМ в космических лучах используется для ограничения или определения свойств темной материи, что является конечной целью косвенного поиска.

Например, в экспериментах AMS [6], PAMELA [7] измеряется доля античастиц в космических лучах, в экспериментах WMAP [8], PLANCK [9] изучается реликтовое излучение, а в эксперименте FERMI-LAT [10] — γ - излучение от разных объектов галактики.

Эксперименты по прямому обнаружению основаны на взаимодействии

WIMP из локального гало темной материи с ядрами-мишенями экспериментов на Земле. Например, эксперименты CRESST [11], CoGeNT [12], EDELWEISS [13] используют кристаллы для обнаружения WIMP-частиц, а эксперименты DarkSide [14], DEAP [15], WARP [16], XENON [17] – благородные газы.

Простая модель распределения WIMP в нашей галактике предполагает, что они должны собираться в сферическом изотермическом гало. Солнечная система движется через это гало со скоростью 220 км/с [18]. Для WIMP с массой m и начальной кинетической энергией $E_0 = \frac{1}{2}m_\chi v_0^2$, ядро с массой m_N будет рассеиваться на угол θ , с энергией:

$$E_R = \frac{\mu_\chi^2 v_0^2 (1 - \cos\theta)}{m_N}, \quad (2)$$

где $\mu_\chi = \frac{m_\chi m_N}{m_\chi + m_N}$ — приведённая масса системы WIMP-ядро, v_0 — скорость WIMP. Для WIMP с массой 100 ГэВ/с² и $v_0 = 200$ км/с и аргоном в качестве ядра-мишени ($m_N = 40$ ГэВ/с²) ожидаемая энергия отдачи очень мала, порядка десятков кэВ. По этой причине в экспериментах с темной материей необходимы низкие энергетические пороги для увеличения чувствительности для поиска WIMP. Также в этой же области энергий регистрируются события, вызванные рассеянием нейтронов, что создает большое количество фоновых событий и делает поиск WIMP затруднительным.

Ожидаемая частота событий на 1 кэВ, на килограмм мишени и за 1 день как функция энергии отдачи в предположении, что сечение вимп-нуклонов составляет 10^{-45} см² и масса WIMP 1 ТэВ/с² изображена на рисунке 3. Ожидаемая скорость прямого обнаружения темной материи в жидком аргоне составляет порядка 10^{-7} - 10^{-10} событий/день/кг/кэВ. Следовательно, для обнаружения вимпов необходимо создавать детекторы объемом не менее 1 тонны.

Эксперименты могут быть чувствительными как к ядерным спин - независимым (spin-independent – SI) взаимодействиям, так и к спин - зависимым (spin-dependent – SD). Для интересующего диапазона обмена импульсами ожидается, что взаимодействие SI будет приблизительно когерентным по всему ядру, поэтому для WIMP-частиц с равной связью с протонами и нейтронами поперечное сечение масштабируется с квадратом атомной массы ядра-мишени [19]. Поэтому текущие эксперименты более чувствительны к темной материи SI, чем к темной материи SD.

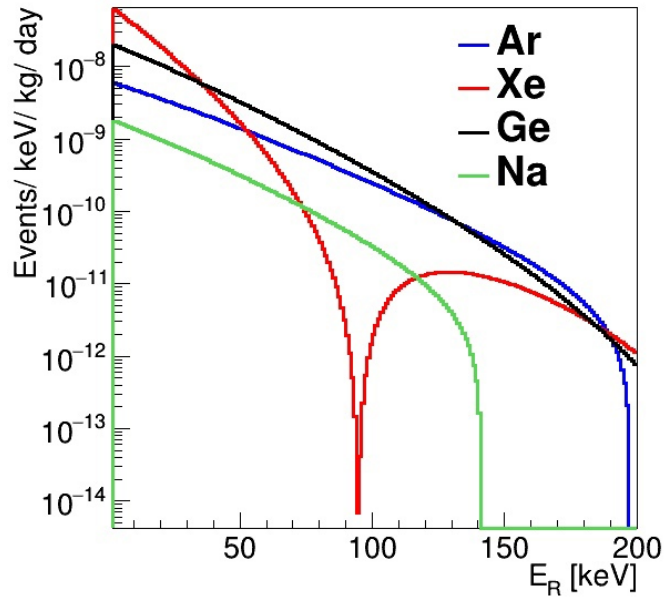


Рисунок 3 — Ожидаемая частота событий/день/кг/кэВ как функция энергии E_R для различных материалов мишени с учетом сечения рассеяния WIMP-нуклон 10^{-45} см^2 и массы WIMP $1 \text{ ТэВ}/c^2$

Различные типы экспериментов конкурируют за достижение наивысшей чувствительности и минимально возможных детектируемых сечений взаимодействия WIMP-нуклон. Обычно чувствительность экспериментов по прямому обнаружению выражается в терминах поперечного сечения взаимодействия WIMP-нуклон (см^2) в зависимости от массы WIMP ($\text{ГэВ}/c^2$). Чтобы иметь возможность сравнивать результаты различных экспериментов с разными материалами мишени, сечение определяется как рассеяние WIMP на нуклоне мишени (а не на ядре). Результаты различных экспериментов представлены на рисунке 4.

МАШИННОЕ ОБУЧЕНИЕ В ФИЗИКЕ ЧАСТИЦ

Наши знания о природе элементарных частицах и их взаимодействиях резюмируются стандартной моделью физики элементарных частиц. Улучшение нашего понимания в этой области потребовало экспериментов, которые работают при все более высоких энергиях и интенсивностях, которые производят чрезвычайно большие и богатые информацией выборки данных. Использование

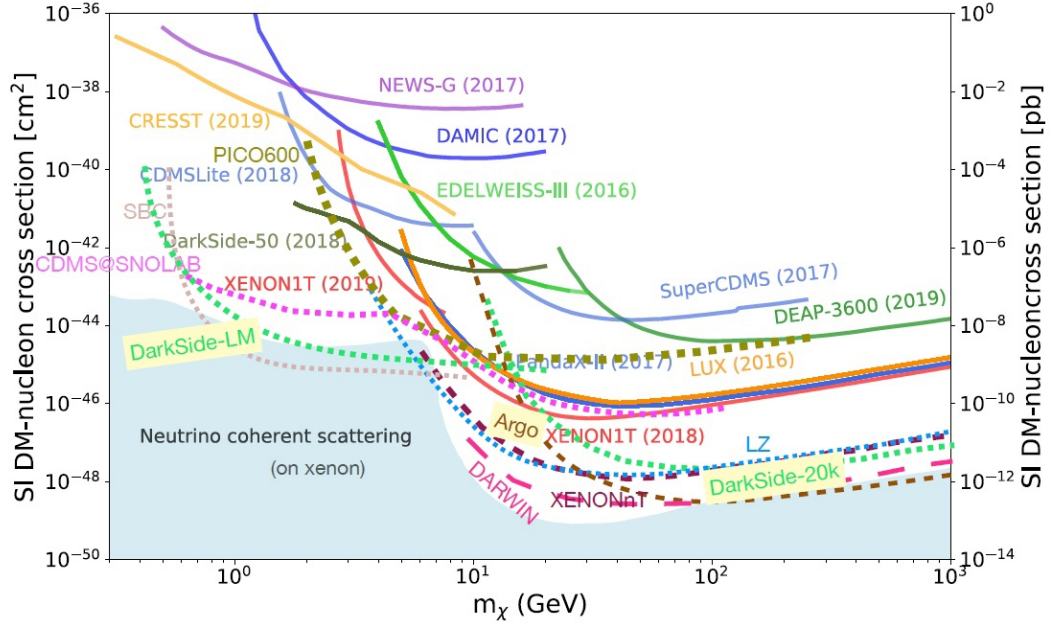


Рисунок 4 — Ограничения на спин-независимые сечения WIMP-нуклон как функция массы WIMP, полученные или планируемые в различных экспериментах[20]

методов машинного обучения революционизирует то, как мы интерпретируем эти образцы данных, значительно увеличивая потенциал нынешних и будущих экспериментов [21].

Многие известные частицы были обнаружены с помощью детекторов, которые сделали следы субатомных частиц видимыми человеческому глазу. Например, пузырьковые камеры [22], заполненные перегретыми жидкостями, которые вскипают при прохождении через них заряженных частиц, преобразуют траектории частиц в видимые следы пузырьков, которые затем можно сфотографировать и проанализировать. Детекторы на Большом адронном коллайдере (LHC) [23] намного сложнее и регистрируют данные с гораздо большей скоростью, чем это возможно с использованием пузырьковых камер. Например, в эксперименте LHCb [24] каждые шесть секунд анализируется столько событий, сколько Большая европейская пузырьковая камера зафиксировала за все 11 лет своей работы (1973–1983), а наборы данных, собранные экспериментами ATLAS [25] и CMS [26] на LHC, сопоставимы с крупнейшими промышленными выборками данных. Люди не могут визуально изучить такие большие объемы данных; алгоритмы, работающие на больших вычислительных фермах, давно взяли на себя эту задачу [21].

За последние два десятилетия физика элементарных частиц перешла на

использование методов машинного обучения при сборе и анализе больших выборок данных [27]. Новаторские исследования, в которых использовались нейронные сети [28; 29] и усиленные деревья решений (BDT) [30; 31] в экспериментах предыдущего поколения [32–43], заложили основу для появления машинного обучения как важного инструмента на LHC. Алгоритмы машинного обучения внесли важный вклад в открытие бозона Хиггса [25; 26], и большинство задач анализа данных теперь выигрывают от использования машинного обучения. Параллельно с этим, область машинного обучения развивалась быстрыми темпами, и, в частности, раздел глубокого обучения обеспечил сверхчеловеческую производительность в нескольких областях [44–46]. Включение этих инструментов при сохранении научной строгости, необходимой для анализа физики элементарных частиц, расширяет горизонты развития науки [21].

1. DEAP-3600

1.1. ОПИСАНИЕ ДЕТЕКТОРА

DEAP-3600 — это однофазный детектор по поиску темной материи, в котором в качестве мишени используется жидкий аргон (LAr) [15]. Схематически он изображен на рисунке 1.1 и имеет следующие характеристики и особенности [47]:

- а) расположен в лаборатории SNOLAB (Онтарио, Канада) на глубине ≈ 2 км;
- б) предел на спин независимое сечение рассеяния частиц темной материи на аргоновой мишени – $3.9 \times 10^{-45} \text{ см}^2$ ($1.5 \times 10^{-44} \text{ см}^2$) для массы WIMP $100 \text{ ГэВ}/c^2$ ($1 \text{ ТэВ}/c^2$) на уровне достоверности 90% [47];
- в) мишенью является (3279 ± 96) кг жидкого аргона ^{40}Ar (LAr). Также имеется 30 см прослойка газообразного аргона (GAr) сверху детектора (рисунк 1.1);
- г) аргон содержится в акриловом сосуде (acrylic vessel, AV) толщиной 5 см с внутренним диаметром 1.7 м;
- д) Внутренняя поверхность AV покрыта слоем 1, 1, 4, 4 - тетрафенил - 1, 3 - бутадиена (TPB) толщиной 3 мкм, который преобразует 128-нм сцинтилляционный свет, создаваемый LAr, в волны видимого света с пиковым значением в 420 нм [48];
- е) при пролете частицы через рабочий объем, она рассеивается на аргоне, производя сцинтилляционный свет. Сигнал внутри детектора регистрируется 255 8-дюймовыми фотоэлектронными умножителями (ФЭУ) Hamamatsu R5912 HQE с низкой радиоактивностью. Пример распределения фотоэлектронов для одного события 1.2;
- ж) сверху детектора имеется отдел горловины. Она содержит охлаждающий змеевик из нержавеющей стали, заполненный жидким азотом (LN_2), который конденсирует GAr во время заполнения и работы. Конденсированный LAr попадает в AV с помощью акриловых направляющих (flowguides, FG),

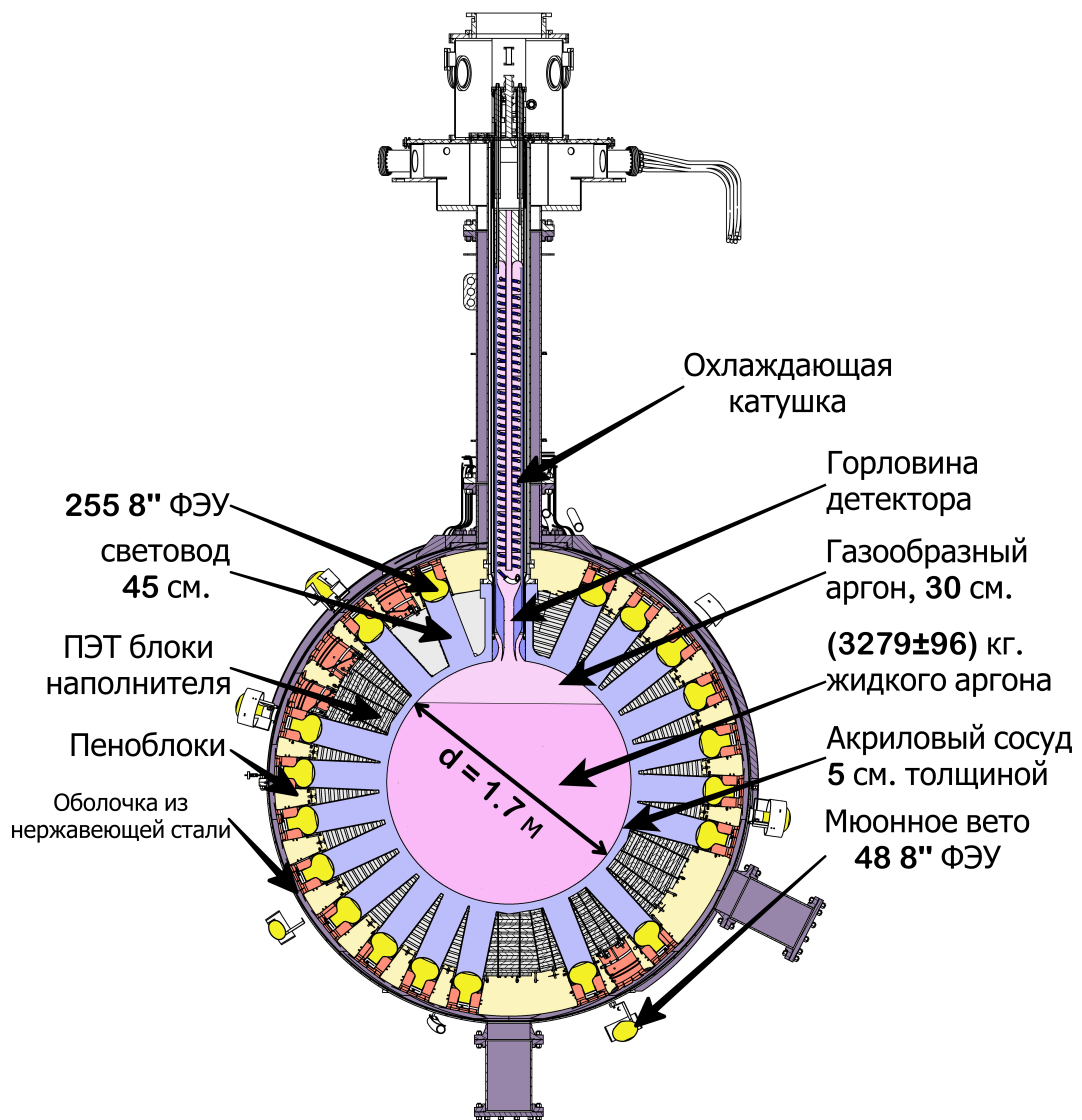


Рисунок 1.1 — Внутреннее строение детектора DEAP-3600

расположенных в отверстии шейки. Эти FG направляют поток аргона к охлаждающему змеевику и от него во время работы детектора;

- з) основание горловины обернуто в оптические волокна, и в сочетании с 4 дополнительными ФЭУ образуют шейное вето — дополнительную защиту детектора от фоновых событий, которые могут появляться в этой области из-за наличия горловины;

и) нейтронная защита:

- (а) ФЭУ соединены с поглощающим ультрафиолет акриловыми световодами (light guide, LG) длиной 45 см, которые транспортируют видимые фотоны от акрилового сосуда к ФЭУ;

- (б) пространство между световодами заполняется чередующимися слоями полиэтилена высокой плотности и блоками наполнения из пенополистирола, которые обеспечивают пассивную защиту компонентов детектора от нейтронов. Блоки наполнителя также обеспечивают теплоизоляцию, поэтому ФЭУ работают при температуре 240-290 К.
- к) черенковское мюонное veto: конструкция заключена в сферическую оболочку из нержавеющей стали. Эта сфера погружена в резервуар с водой высотой 7.8 м и шириной 7.8 м с 48 обращенными наружу ФЭУ Hamamatsu R1408 диаметром 8 дюймов, установленными на внешней поверхности сферы. Вместе эти ФЭУ и резервуар с водой составляют черенковское мюонное veto, используемое для маркировки космогенно - индуцированного фона, в то время как защитная вода обеспечивает подавление нейтронного и гамма-фона из пещеры.

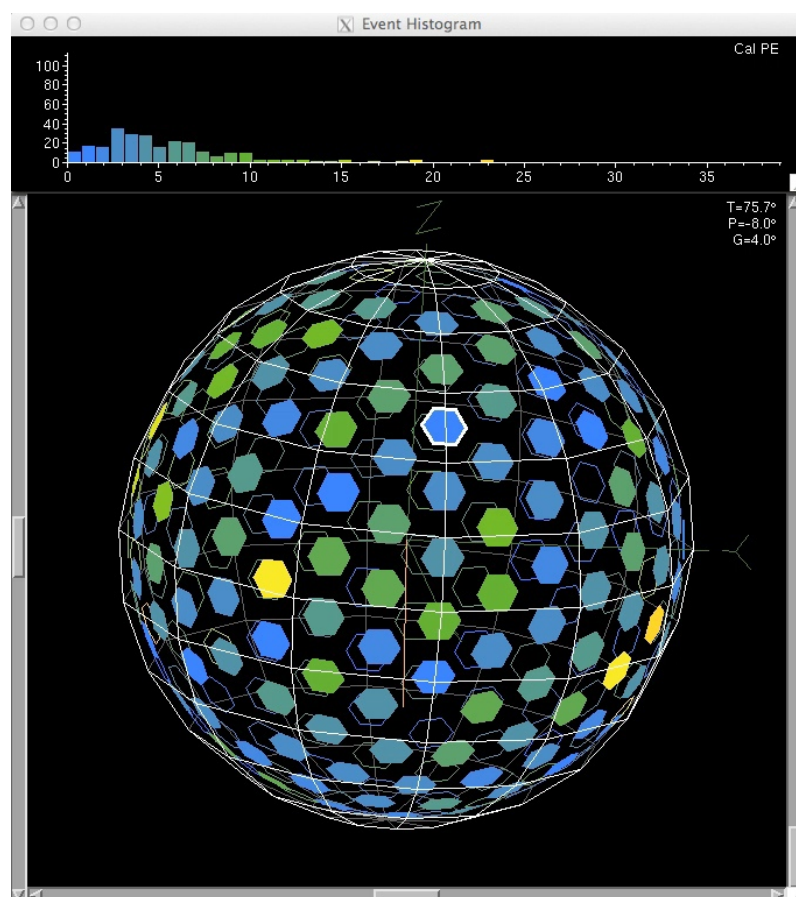


Рисунок 1.2 — Пример регистрации фотоэлектронов для одного события. Все ФЭУ схематически изображены в виде гексагонов. Шкала наверху и различный цвет ФЭУ означает распределение количества зарегистрированных фотоэлектронов

^{39}Ar — это источник β - излучения, который естественным образом встре-

чается в атмосферном аргоне, используемом в детекторе DEAP-3600. β - распады ^{39}Ar обеспечивают высокую статистику сцинтилляционных сигналов в LAr в ответ на электроны с энергией между порогом срабатывания детектора и конечной точкой ^{39}Ar при 565 кэВ_{ее} [49]. В DEAP-3600 запускается триггер и собираются данные, когда в скользящем окне 177 нс обнаруживается общий заряд, равный среднему заряду приблизительно 19 фотоэлектронов. После запуска система сбора данных (DAQ) записывает напряжение на каждом ФЭУ каждые 4 нс. Оцифровка устанавливается таким образом, что пик события приходится примерно на 2,5 мкс на оцифрованных графиках ФЭУ. Нормальные данные поиска темной материи имеют окно событий длиной 16 мкс. Алгоритм поиска импульсов применяется к оцифрованным сигналам ФЭУ для определения заряда и времени каждого импульса. Импульсный заряд преобразуется в фотоэлектроны путем деления на средний однофотоэлектронный заряд ФЭУ. Результирующая переменная, называемая qPE, содержит истинные фотоэлектроны, а также темновой шум ФЭУ и постимпульсы. Время пика события определяется на основе времени, когда обнаруживается наибольшее количество qPE. Затем времена прихода qPE корректируются так, чтобы пик события приходился на $t = 0$ нс. Пример результирующего калиброванного сигнала от события с электронной отдачей с энергией приблизительно 20 кэВ_{ее} показан на рисунке 1.3.

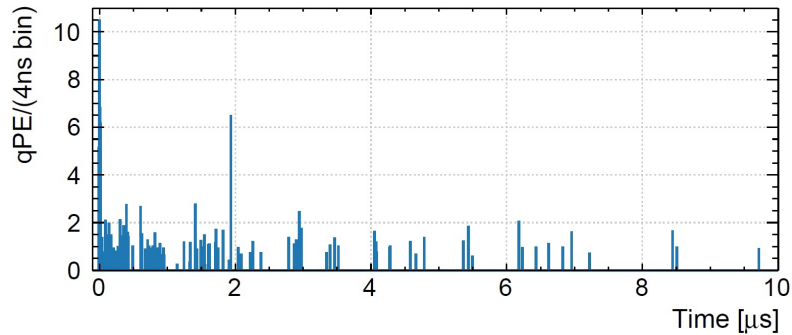


Рисунок 1.3 — Пример результирующего калиброванного сигнала от события с электронной отдачей с энергией приблизительно 20 кэВ_{ее}

DEAP-3600 — это один из немногих в мире детекторов по прямому обнаружению темной материи, использующих жидкий благородный газ в качестве мишени. Благодаря сцинтилляции жидкого аргона детектор обладает мощным свойством разделения по форме импульса, позволяющим эффективно идентифицировать и разделять события с ядерной и электронной отдачей [15].

1.2. АЛГОРИТМЫ РЕКОНСТРУКЦИИ ПОЛОЖЕНИЯ СОБЫТИЯ

В настоящий момент используются два различных алгоритма восстановления положения, разработанные коллаборацией DEAP-3600 [47]. Первый алгоритм, названный “MBLikelihood”, полагается исключительно на пространственное распределение заряда через ФЭУ для восстановления положения. Второй алгоритм, называемый “TimeFit2”, использует время прихода фотонов для восстановления положения.

Оба алгоритма эффективны. В данной работе описывается несколько методов на основе нейронных сетей для улучшения имеющегося результата реконструкции положения события.

1.2.1. МЕТОД MBLIKELIHOOD

MBLikelihood сравнивает наблюдаемое распределение фотоэлектронов (РЕ) в каждом ФЭУ с предсказанным распределением с учетом гипотетического положения события. Эти предполагаемые распределения подготавливаются с использованием данных Монте-Карло моделирования, которое включает в себя полную модель отклика детектора. Восстановленное положение события, возвращенное MBLikelihood, соответствует вершине события, в которой распределение РЕ в каждом ФЭУ наиболее согласуется с предсказанным распределением.

Рассмотрим функцию правдоподобия $\mathcal{L}(\vec{x})$, которая описывает вероятность наблюдения события с учетом предполагаемого положения $\vec{x}$, где $(\vec{x}) = (x, y, z)$ [47]:

$$\ln \mathcal{L}(\vec{x}) = \sum_{i=1}^{N_{\text{PMTs}}} \ln \text{Poisson}(q_i, \lambda_i), \quad \lambda_i = \lambda_i \left(|\vec{x}|, \frac{\vec{x} \cdot \vec{r}_i}{|\vec{x}| |\vec{r}_i|}, q_{\text{total}} \right), \quad (1.1)$$

где $\text{Poisson}(q_i, \lambda_i)$ — пуассоновская вероятность обнаружить q_i фотоэлектронов в ФЭУ с номером i в некоторой точке $\vec{r}_i$ в течение всего окна событий 10 мкс. Ожидаемое количество фотоэлектронов в ФЭУ i определяется как λ_i , которое

является функцией радиуса тестового положения $|\vec{x}|$, угла между тестовым положением и ФЭУ с номером i и общего количества фотоэлектронов q_{total} [47].

Функция правдоподобия обращается к справочным таблицам для поиска значений λ_i , которые генерируются с использованием высокой статистики, Монте-Карло моделирования событий сцинтилляции, созданных в детекторе в дискретных положениях вершин вдоль осей (x, y, z) . Начало системы координат реконструкции $(x, y, z) = (0, 0, 0)$ — это центр детектора. Моделирование выполнено в предположении идеально сферической геометрии и полностью заполненного детектора. Окончательная реконструированная позиция, сохраненная в структуре данных, задается значением $\vec{x}$, которое максимизирует $\mathcal{L}(\vec{x})$.

1.2.2. МЕТОД TIMEFIT2

Подобно MBLikelihood, этот алгоритм определяет восстановленную вершину события путем нахождения гипотетического значения $\vec{x}$, максимизируя функцию правдоподобия, однако этот алгоритм учитывает время, а не заряд. Рассмотрим функцию правдоподобия, которая описывает вероятность наблюдения события с учетом предполагаемого местоположения и времени события t [47]:

$$\ln \mathcal{L}(t_0, \vec{x}_0) = \sum_{i=1}^{N_{PE}} \ln \mathcal{L}^{t, \text{res}}(t_i - t_0; \vec{x}_0, \text{PMT}_i), \quad (1.2)$$

где t_i — время i -го фотоэлектрона, наблюдаемого ФЭУ с номером i , N_{PE} — количество быстро прибывающих фотоэлектронов, обнаруженных в первые 40 нс события, а $\mathcal{L}^{t, \text{res}}$ — остаточное распределение времени. Разница во времени между фактическим временем, в течение которого импульс был зарегистрирован ФЭУ, по сравнению с прогнозируемым временем из расчетов времени пролета, определяется как временная невязка. Для набора дискретных значений положения внутри детектора $\mathcal{L}^{t, \text{res}}$ определяется априори для каждого ФЭУ. Объединенные значения $(t, \vec{x})$, которые максимизируют $\mathcal{L}^{t, \text{res}}$, возвращаются и сохраняются в структуре данных как реконструированные время и позиция события.

$\mathcal{L}^{t, \text{res}}$ прогнозируется на основе оптической модели, которая включает в себя групповые скорости УФ-фотонов, излучаемых LAr (11 см/нс при 128 нм)

и видимых фотонов, излучаемых ТРВ (24 см/нс при 420 нм), а также постоянные времени сцинтилляции LAr и флуоресценции ТРВ, среднее время пробега видимых фотонов в акриле LG и AV и среднее время отклика ФЭУ. Групповые скорости, использованные для этих расчетов, были определены на основе измеренных показателей преломления LAr на различных длинах волн [50; 51]. Для простоты в модель не включены некоторые эффекты, такие как рассеяние фотонов на ТРВ и внутри него и рэлеевское рассеяние фотонов в LAr [47].

1.2.3. ПРЕИМУЩЕСТВА И НЕДОСТАТКИ АЛГОРИТМОВ, ЦЕЛЬ РАБОТЫ

На рисунке 1.4 изображены графики, демонстрирующие согласованность между методами MBLikelihood и TimeFit2.

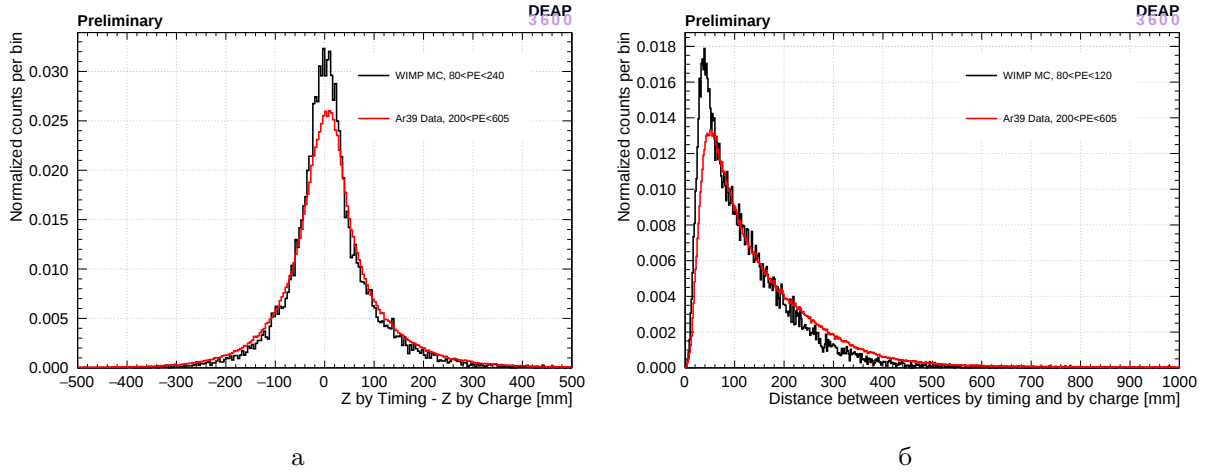


Рисунок 1.4 — Согласованность между методами TimeFit2 и MBLikelihood. **а**: разница в восстановленной координате Z, мм от двух методов. Показано для WIMP Монте-Карло (^{40}Ar) черным цветом, а для данных ^{39}Ar — красным, **б**: расстояние, мм между восстановленными положениями одного события двумя разными методами. Показано для WIMP Монте-Карло (^{40}Ar) черным цветом, а для данных ^{39}Ar — красным [47]

При допущении об одном источнике сцинтилляционного света ожидается, что реконструкция события методами MBLikelihood и TimeFit2 будет согласована для равномерно распределенных событий, таких как WIMP и ^{39}Ar . Однако не ожидается, что она будет согласована для событий с несколькими источниками, событий со значительным вторичным импульсом или событий от горловины детектора. События WIMP Монте-Карло (^{40}Ar) выбираются в диа-

пазоне энергии от (80; 240) РЕ, в то время как события ^{39}Ar выбираются в диапазоне (200; 605) РЕ, чтобы иметь такое же количество быстрых фотоэлектронов (то есть фотоэлектронов, пришедших в течение первых 200 нс) с WIMP, чтобы сделать относительно справедливое сравнение. Несколько худшая согласованность, показанная на ^{39}Ar , связана с тем, что TimeFit2 использует только 40 нс быстрого света. ^{39}Ar имеет меньше импульсов в первые 40 нс, поэтому в TimeFit2 используется немного меньше данных.

Оба алгоритма согласуются по координате Z для событий ^{39}Ar в данных и для моделирования отдачи ^{40}Ar с точностью до 35 мм для 50% таких событий [47].

Среднее значение ошибки а также разрешение алгоритма реконструкции для Монте-Карло данных ^{39}Ar в диапазоне (50; 250) РЕ для конкретных значений Монте-Карло представлено на рисунке 1.5.

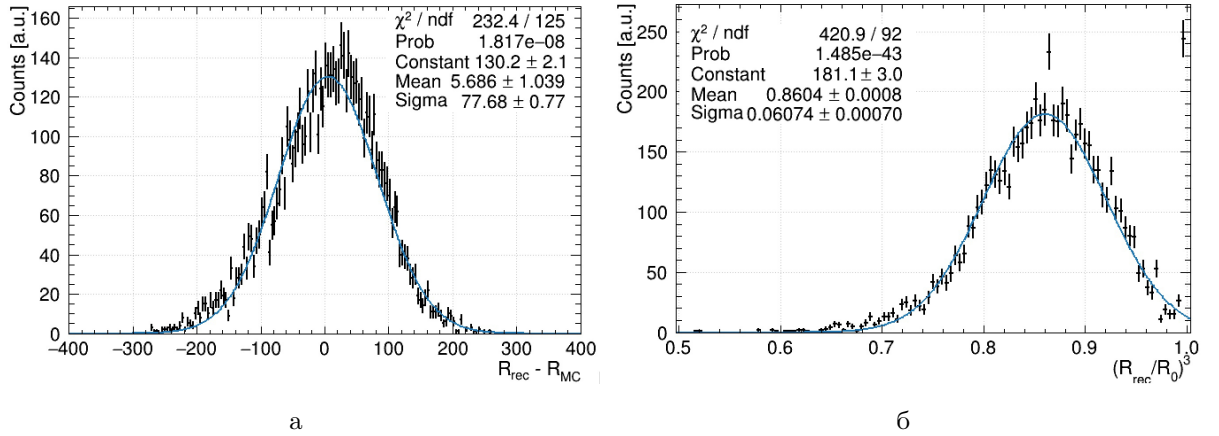


Рисунок 1.5 — Гауссово распределение **а**: $R_{\text{rec}} - R_{\text{MC}}$, мм для Монте-Карло данных β -распадов ^{39}Ar в диапазоне (50; 250) РЕ для $R_{\text{MC}} = 313.51$ мм; **б**: $(R_{\text{rec}}/R_0)^3$ для Монте-Карло данных β -распадов ^{39}Ar в диапазоне (50; 250) РЕ для $(R_{\text{MC}}/R_0)^3 = 0.903$

На данный момент анализ поиска WIMP основывается в первую очередь на алгоритме MBLikelihood с выделением чувствительного объема ($R < 630$ мм), однако также необходимо, чтобы оба алгоритма сходились и согласовывались друг с другом. Это позволяет отвергать события, положение которых неправильно реконструировано, где допущения, лежащие в основе обоих алгоритмов, не реализуются. При выделении чувствительного объема данные и реконструкция, полученная алгоритмом на основе РЕ, согласуются с точностью до 13% [47].

Для оценки точности реконструкции используется метод, в котором события распада ^{39}Ar разделяются на два “псевдособытия”: каждый РЕ из события независимо назначается каждому из двух псевдособытий с вероятностью 50%. Это приводит к тому, что оба псевдособытия имеют примерно половину РЕ от исходного события из одного и того же истинного положения. Точность положения определяется из распределения восстановленных расстояний между псевдособытиями и радиуса восстановления исходного события. В пределах области поиска WIMP вплоть до радиуса $R = 630$ мм псевдособытия обычно реконструируются в пределах 30 - 45 мм друг от друга.

Целью данной работы является создание принципиально нового алгоритма, основанного только на физических параметрах (импульсы в каждом ФЭУ, время прихода фотоэлектронов) самого события, который будет:

- 1) использоваться во всём объеме детектора $R < 850$ мм. Это позволит применять реконструкцию во всем детекторе и увеличить чувствительный объем детектора;
- 2) работать не менее эффективно и точно, чем существующий алгоритм реконструкции;
- 3) дополнять и поддерживать существующий алгоритм.

2. ИССЛЕДОВАТЕЛЬСКАЯ РАБОТА

2.1. ИСПОЛЬЗУЕМЫЕ АЛГОРИТМЫ РЕКОНСТРУКЦИИ

В работе используется три принципиально разных алгоритма машинного обучения.

2.1.1. ПОЛНОСВЯЗНАЯ НЕЙРОННАЯ СЕТЬ

Нейронная сеть [52] — это алгоритм машинного обучения, в котором существует входной слой, в котором количество нейронов эквивалентно количеству параметров события, выходной слой, в котором количество нейронов соответствует количеству определяемых переменных, и любого количества “скрытых” слоев, в котором может быть любое количество нейронов. Полносвязная нейронная сеть — это нейронная сеть, в котором каждый нейрон слоя связан со всеми нейронами следующего и предыдущего слоя. Эти связи являются весовыми значениями и настраиваются в течение обучения. Схема полносвязной нейронной сети представлена на рисунке 2.1.

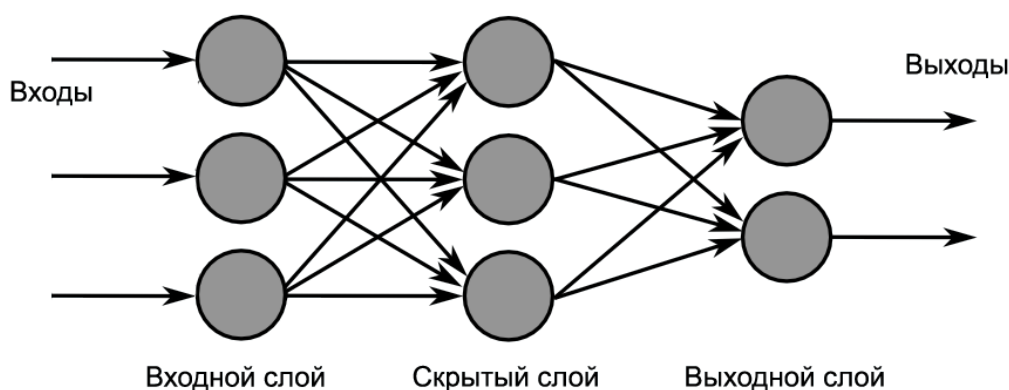


Рисунок 2.1 — Схема полносвязной нейронной сети

Основным слоем в такой структуре является полносвязный (то есть такой слой, в котором каждый нейрон связан со всеми нейронами следующего слоя)

Dense слой. В качестве аргументов он имеет значение количества нейронов в данном слое и функцию активации. Принцип работы полносвязного слоя и функции активации представлен на рисунке 2.2.

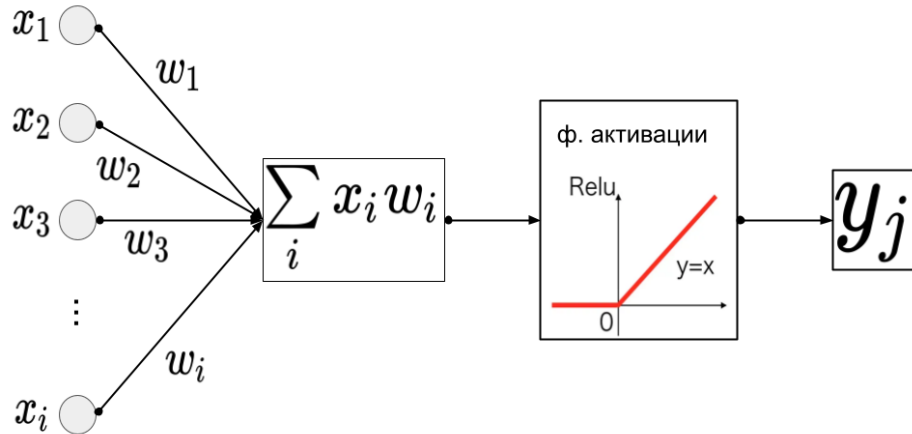


Рисунок 2.2 — Схема работы нейрона и функции активации

В данной работе использовались две функции активации: “reLU” и “linear”, которые выражаются следующим образом:

$$f_{reLU}(x) = \begin{cases} 0, & x < 0, \\ x, & x > 0 \end{cases}, \quad f_{linear}(x) = x. \quad (2.1)$$

2.1.2. НЕЙРОННАЯ СЕТЬ С КОРОТКИМИ СВЯЗЯМИ

Нейронная сеть с короткими (*shortcut*) связями [53; 54] — это нейронная сеть, в котором выходной вектор с какого-либо слоя передается на более чем один последующий слой. Преимуществом такой сети перед полносвязными является тот факт, что в процессе обучения модель на последних слоях не теряет информацию, которая могла быть получена на более ранних слоях. Недостатками такого алгоритма является необходимость соответствия длин векторов начального и конечного слоя *shortcut* соединения, а также более высокое время работы сети. Схема такой сети представлена на рисунке 2.3. Структура и основные элементы сети идентичны полносвязной нейронной сети.

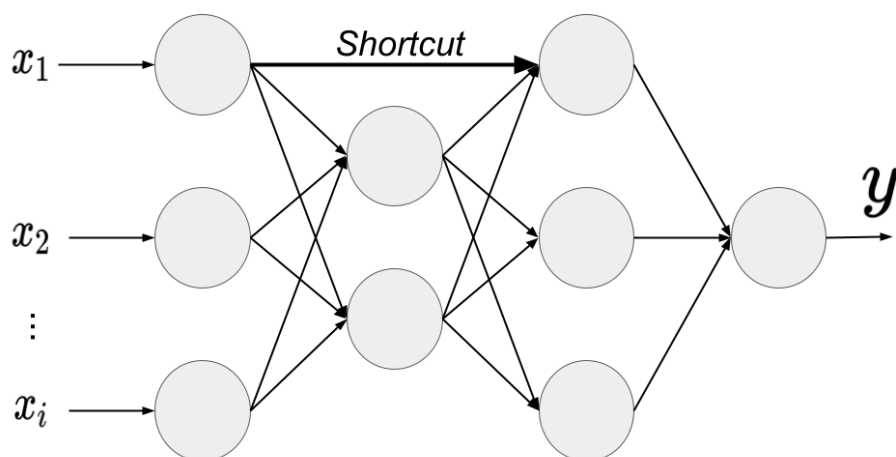


Рисунок 2.3 — Схема нейронной сети с короткими связями

2.1.3. НЕЙРОННАЯ СЕТЬ СО СВЕРТОЧНЫМИ СЛОЯМИ

Сверточная нейронная сеть [55; 56] — это алгоритм глубокого обучения, который используется для нахождения паттернов во входных данных. Эти сети основаны на слоях свертки — они математически сворачивают входную матрицу в матричное представление с определенным набором фильтров. Каждое ядро призвано подчеркнуть определенные особенности входной матрицы. После последовательности этих слоев выходная матрица выравнивается в вектор и распространяется на полносвязную нейронную сеть. Схема работы одномерного свёрточного слоя представлен на рисунке 2.4.

Основными слоями в данной модели служили: *Conv1D* — выполняли свертку входного слоя с разными ядрами для обнаружения основных характеристик входных данных, *Flatten* — для преобразования матрицы в вектор для дальнейшего использования его в полносвязной нейронной сети и *Dense* — для окончательного определения координаты.

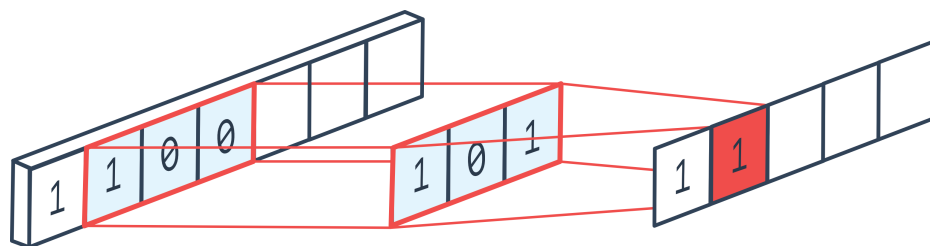


Рисунок 2.4 — Схема работы одномерного ядра свёрточной нейронной сети

Также в работе использовались свёрточные нейронные сети с использо-

ванием карт распределения фотоэлектронов в каждом ФЭУ (heatmap). Схема работы такого алгоритма представлен на рисунке 2.5.

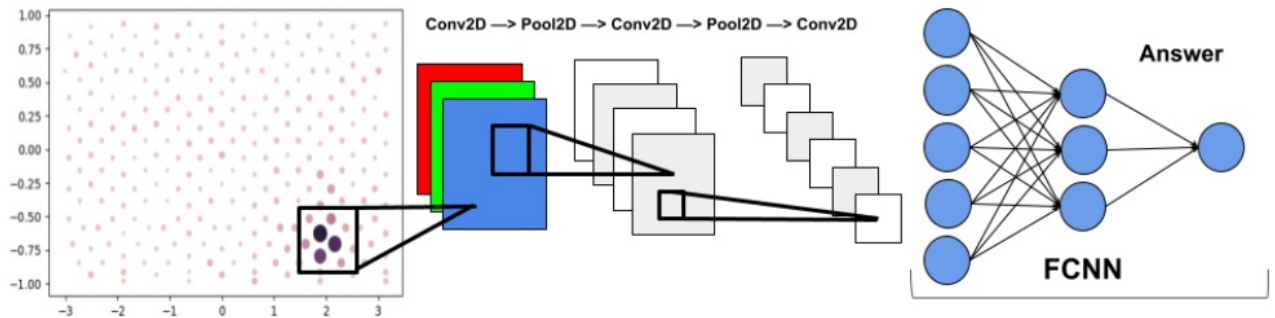


Рисунок 2.5 — Схема работы свёрточной нейронной сети

В первую очередь карты распределения фотоэлектронов преобразуются в матричное представление размером $128 \times 128 \times 3$, где первые два значения — это пиксельный размер изображения, а 3 число означает разложение изображения по трем каналам цветов — красный, зеленый и синий (RGB). Эти матрицы и передаются на свёрточную нейронную сеть. Здесь сверточный слой (*Conv2D*) — выполняет свертку входного слоя с разными ядрами для обнаружения основных характеристик входных данных (рисунок 2.6); слой максимального объединения (*MaxPooling2D*) — понижает размерность матрицы, принимая максимальное значение в окне заданной размерности; модель построена как простой стек слоев, где каждый слой имеет ровно один входной тензор и один выходной тензор; на последнем слое матрица сворачивается в одномерный вектор и он передается на полносвязные слои нейронной сети.

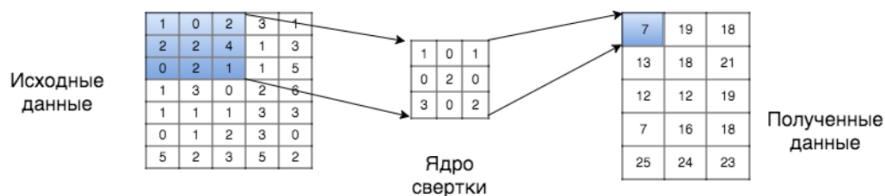


Рисунок 2.6 — Схема работы двумерного ядра свёрточной нейронной сети

2.2. ХОД РАБОТЫ

Ход работы был следующим:

- 1) с помощью данных, полученных в результате Монте-Карло моделирования естественной радиоактивности ^{39}Ar в детекторе DEAP-3600 получены root-файлы, которые содержат в себе информацию о величине, времени прихода и номера ФЭУ для каждого фотоэлектрона;
- 2) с помощью root-файлов была создана база данных, которая содержит 785 столбцов и 900 тысяч строк. Каждый столбец — количество фотоэлектронов, пришедших в каждый ФЭУ, отнормированное на суммарное количество фотоэлектронов, а также время начала и окончания принятия сигнала в ФЭУ, каждая строка — отдельное событие. Распределение смоделированных координат $X - Y - Z$ событий представлено на рисунке 2.7;

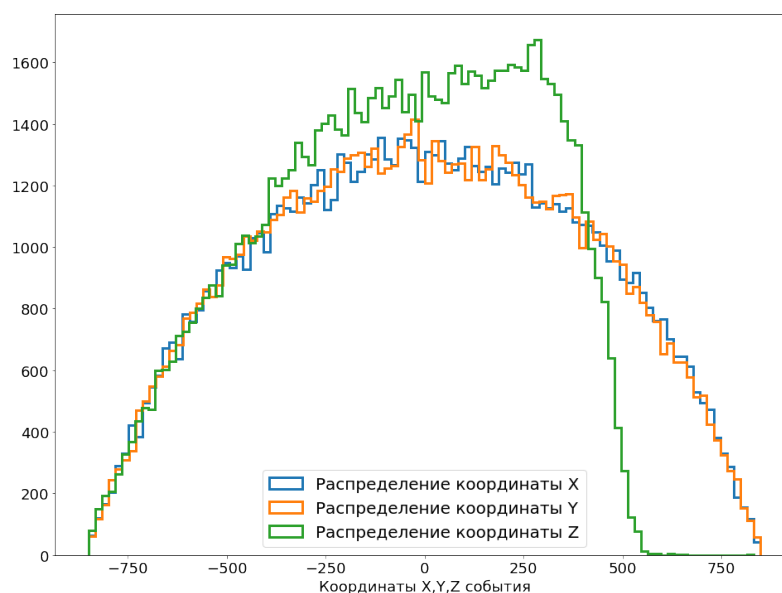


Рисунок 2.7 — Распределение координат X , Y , Z , мм Монте-Карло моделирования. Необычное распределение координаты Z объясняется прослойкой газообразного аргона вверху детектора

- 3) в качестве параметров событий были использованы 765 значений импульсов и времен начала и окончания сигнала в каждом ФЭУ. В процессе работы было обнаружено, что эффективность модели не сильно зависит от наличия времен прихода фотоэлектронов в ФЭУ в качестве параметров события и в дальнейшем использовались только величины импульсов в каждом из 255 ФЭУ;
- 4) для проверки модели нейронной сети на основе карт распределения фотоэлектронов было создано около 9.2×10^4 heatmap-карт. Примеры таких карт представлены на рисунке 2.8.

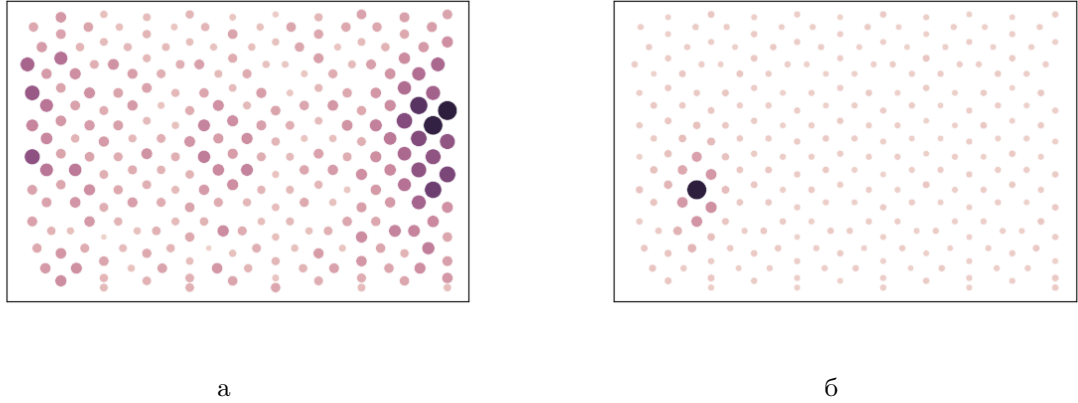


Рисунок 2.8 — Примеры распределения фотоэлектронов в фотоэлектронных умножителях (heatmap-карты)

5) было рассмотрено несколько конфигураций слоёв нейронной сети, а также несколько областей детектора с применяемым физическим ограничением на радиус рабочего объема детектора. Количество событий N в тренировочном наборе в каждой из областей указаны в таблице 2.1. События были разделены в соотношении 70:30 для тренировки и тестирования моделей. Это делается с целью предотвращения перетренированности на одних и тех же данных.

Таблица 2.1 — Количество событий N , оставшееся после применения ограничения на радиус рабочего объема детектора R , мм в тренировочном наборе

	$R < 430$	$R < 630$	$R < 720$	$R < 850$
$N, 10^3$	107	300	426	677
	$200 < R < 430$	$430 < R < 630$	$630 < R < 720$	$720 < R < 850$
$N, 10^3$	96	193	126	251

6) в работе использовались четыре различные структуры нейронных сетей. Описание работы нейронных сетей можно найти в Разделе 2.1, а полное описание используемых алгоритмов в приложении А. Были подобраны наиболее оптимальные параметры моделей, которые одновременно ведут к (а) наибольшей эффективности и (б) наименьшим вычислительным затратам. Вкратце каждая сеть представляет из себя следующую структуру:

- (а) Полносвязная нейронная сеть (FCNN) с входным слоем, четырьмя полносвязными скрытыми слоями и выходным слоем;
- (б) нейронная сеть с короткими связями с тремя одинаковыми блока-

- ми, которые состояли из 3 полносвязных слоев. Выходное значение с каждого блока передавалось не только на следующий блок, но и на блок через один. Для сопоставления двух векторов, пришедших на вход очередного блока использовался аддитивный слой, который суммировал поэлементно значения векторов. Принцип вычислений аналогичен полносвязной нейронной сети;
- (в) структура сети со свёрточными слоями. Здесь использовалась последовательная структура из 1 входного, 5 слоев свёртки, сглаживающего и 1 полносвязного слоя;
 - (г) структура сети со свёрточными слоями для 2D heatmap-карт. В структуре использовалось последовательно по 3 свёрточных слоя Conv2D и MaxPooling2D с различными параметрами, далее выходная матрица разглаживалась в вектор, который передавался на полносвязную нейронную сеть из 4 скрытых слоёв и выходного слоя.
- 7) модели тренировались в течение 30 эпох партиями по 1000 (500 для модели свёрточной сети с применением heatmap-карт) событий. Эти значения также были подобраны опытным путем. Увеличение количества эпох приводило к перетренированности моделей (то есть точность на тренировочном наборе увеличивалась, а на тестовом — не изменялась), а уменьшение партии — к недостаточно низкому значению ошибки.

2.3. РЕЗУЛЬТАТЫ

Прежде чем проверять модель, необходимо было понять, можно ли использовать какую-либо из них на том же регионе $R < 630$ мм, или можно увеличить чувствительный радиус детектора. Для этого первые три модели были натренированы на различных диапазонах чувствительного радиуса детектора (таблица 2.1). В результате для каждого из радиусов была вычислена средняя ошибка каждого алгоритма в реконструкции. Результаты представлены в таблице 2.2.

Наилучшим образом алгоритмы работают при тренировке на всём объеме детектора ($R < 850$ мм) либо на внешней ее части ($720 < R < 850$ мм), а хуже всего — во внутренней части детектора ($R < 430$ мм) выдавая разрешение по всем 3 координатам около 100 мм. Такой результат можно объяснить двумя

Таблица 2.2 — Погрешности для координат X , Y , Z для каждой рассматриваемой области параметра R , мм. $\text{FWHM} = 2.355 \sigma$

	Точность	X_{FWHM} , мм	Y_{FWHM} , мм	Z_{FWHM} , мм
$R < 850$	0.959 ± 0.003	47 ± 4	49 ± 3	51 ± 7
$R < 720$ мм	0.944 ± 0.001	64 ± 3	64 ± 7	66 ± 2
$R < 630$ мм	0.933 ± 0.002	75 ± 3	76 ± 6	78 ± 2
$R < 430$ мм	0.899 ± 0.007	91 ± 7	92 ± 8	105 ± 15
$720 < R < 850$ мм	0.983 ± 0.003	41 ± 9	42 ± 8	42 ± 9
$630 < R < 720$ мм	0.966 ± 0.005	55 ± 5	55 ± 8	42 ± 5
$430 < R < 630$ мм	0.952 ± 0.002	79 ± 10	77 ± 8	58 ± 2
$200 < R < 430$ мм	0.913 ± 0.004	90 ± 5	88 ± 3	98 ± 7

Таблица 2.3 — Средняя ошибка и разрешение для координат X , Y , Z для всех четырех рассматриваемых моделей машинного обучения. Систематические погрешности не включены. $\text{FWHM} = 2.355 \sigma$

	SC	FCNN	CONVNN	2DCONVNN
X_{FWHM} , мм	44.98	43.04	56.76	76.58
X_{μ} , мм	4.58	4.43	0.74	4.64
Y_{FWHM} , мм	52.10	44.18	55.46	81.83
Y_{μ} , мм	7.47	2.92	0.06	3.09
Z_{FWHM} , мм	48.97	43.42	47.98	68.58
Z_{μ} , мм	0.80	3.13	3.07	1.26

причинами: недостаточным количеством событий Монте-Карло генератора в этой области и сложность в различении событий в связи со схожим количеством фотоэлектронов в каждом ФЭУ.

Итак, модели работают наилучшим образом на всём объеме детектора $R < 850$ мм. Теперь необходимо сравнить эти модели друг с другом. На рисунке 2.9 представлено распределение реконструированных координат X , Y , Z и сравнение этой реконструкции с реальным распределением (рисунок 2.7). По гистограммам видно, что все четыре модели правильно выполняют свою задачу в определении координаты. Никаких особых выбросов не наблюдается.

Для более конкретного сравнения эффективностей на рисунках 2.10, 2.11, 2.12, 2.13 представлены распределения ошибок в определении координат X , Y , Z а также в определении положения события для всех четырех методов нейронной сети. В сводной таблице 2.3 представлены значения этих ошибок без учета систематических погрешностей.

На графиках указаны параметры μ и σ которые являются средней ошиб-

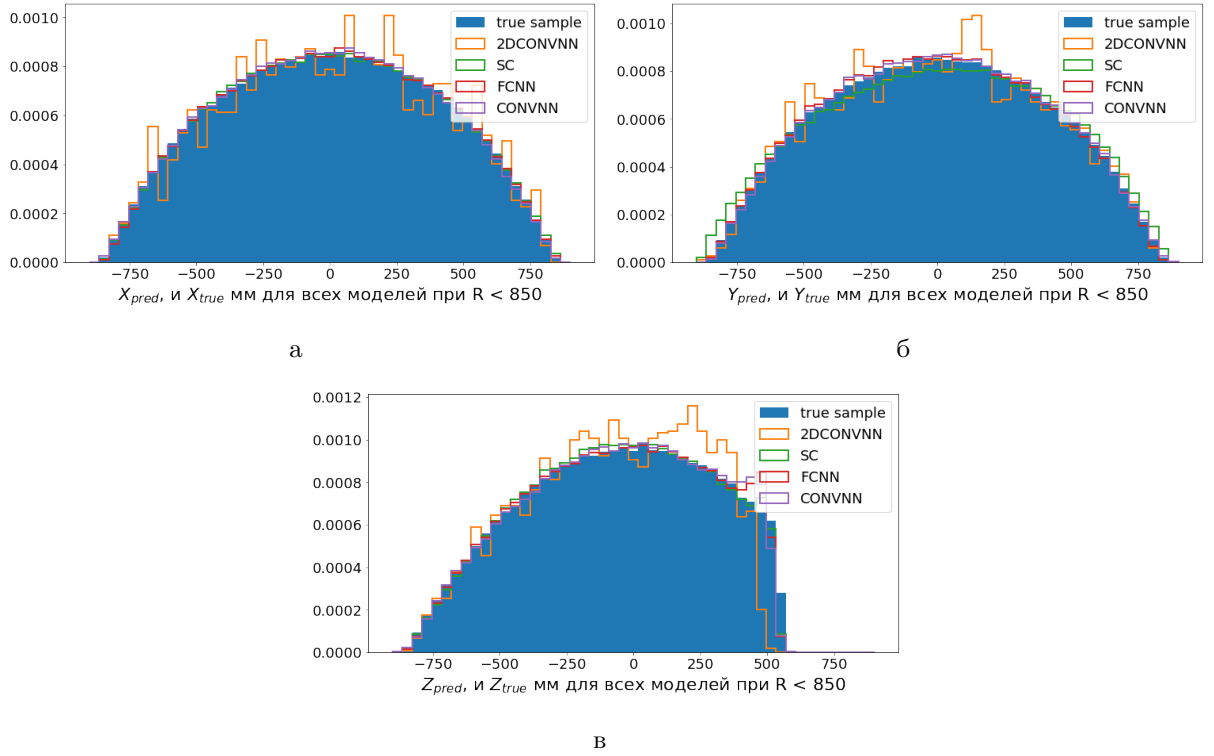


Рисунок 2.9 — Сравнение распределения реконструкции четырех моделей с реальным распределением для координат X (а) Y (б) и Z (в). Необычное распределение реконструированных координат модели 2DCONVNN объясняется намного меньшим статистическим набором данных по сравнению с другими алгоритмами

кой и разрешением той или иной модели. Значения были получены с помощью аппроксимации распределения функция Гаусса и нахождением ее параметров. Наилучшим образом показали себя модели полносвязной нейронной сети FCNN (распределения 2.10б, 2.11б, 2.12б, 2.13б) и одномерной свёрточной сети (распределения 2.10в, 2.11в, 2.12в, 2.13в). 2D свёрточные нейронные сети и сети с короткими связями показали результат хуже. Более того, создание изображений для тренировочного набора и тренировка модели занимает намного больше временных и вычислительных ресурсов.

Далее необходимо сравнить используемые модели с имеющимися алгоритмами. Это было сделано тремя различными способами.

Сначала объем детектора был разделен на оболочки шириной по 50 мм (рисунок 2.16а). Модели тренировались на событиях из всего объема, а тестировались только на событиях из определённой оболочки. Таким образом можно проверить точность реконструкции модели в различных регионах детектора и сравнить с аналитическими моделями.

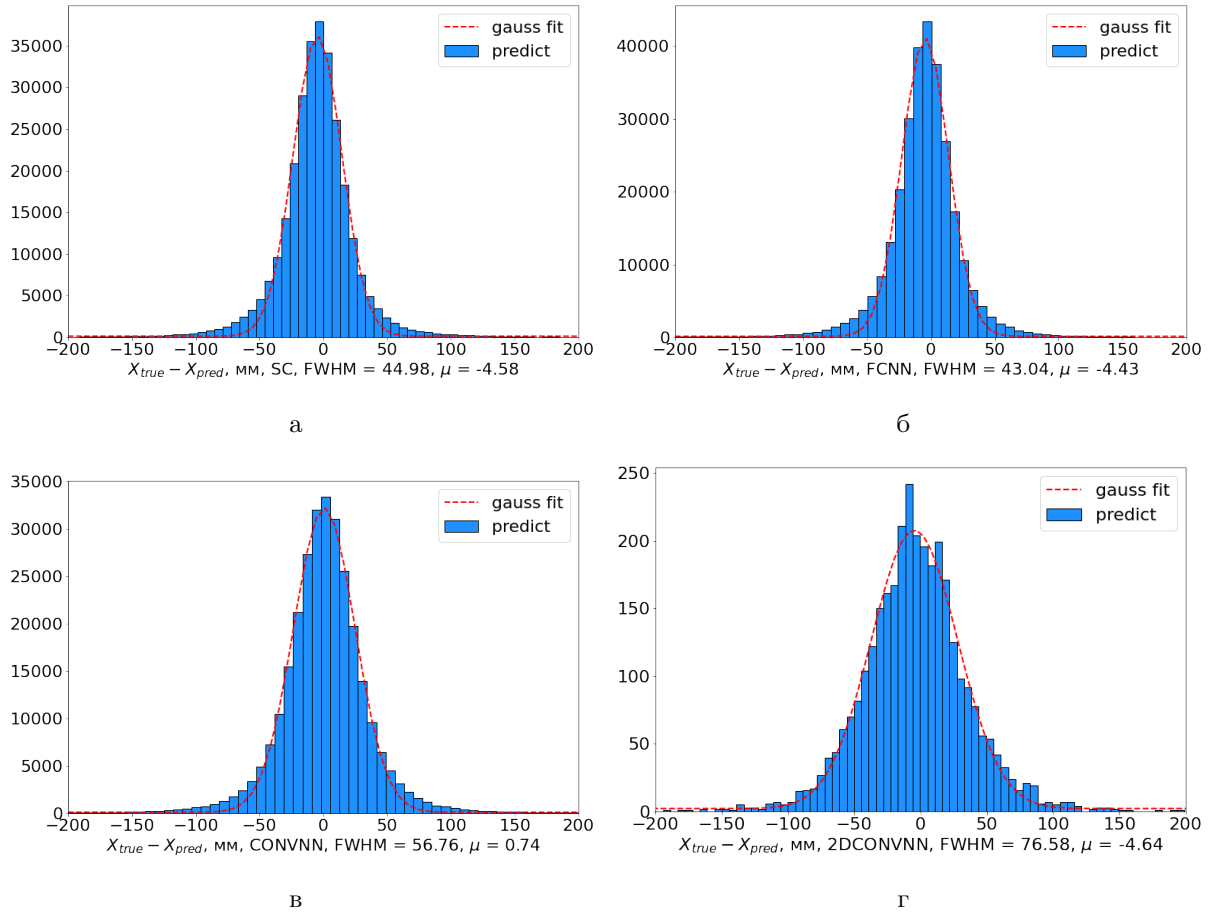


Рисунок 2.10 — Ошибка в определении координаты X для модели на основе (а) — нейронной сети с короткими связями, (б) — полносвязной нейронной сети, (в) — свёрточной нейронной сети и (г) — двумерной свёрточной нейронной сети

На рисунке 2.14 представлено распределение ошибки в реконструкции положения для всех 20 образовавшихся оболочек шириной 50 мм. Близкая к нулю средняя ошибка говорит о хорошей работе алгоритма на всех диапазонах. Различная высота гистограмм связана с различным количеством событий в каждом регионе: равная ширина оболочек образует разный их объем.

На рисунке 2.15 изображено среднее значение расстояния от центра для каждой оболочки и средние значения, предсказанные моделями машинного обучения. Результат хорошо соотносится с истинным значением в большей части детектора. Во внутренней его части имеется расхождение, которое можно объяснить малым количеством тестовых данных в данных областях.

Второй способ — это разделить детектор на оболочки равного объема (рисунок 2.166).

Такое разделение позволит вычислять точность в различных регионах детектора на приблизительно одинаковом количестве тестовых событий. Это

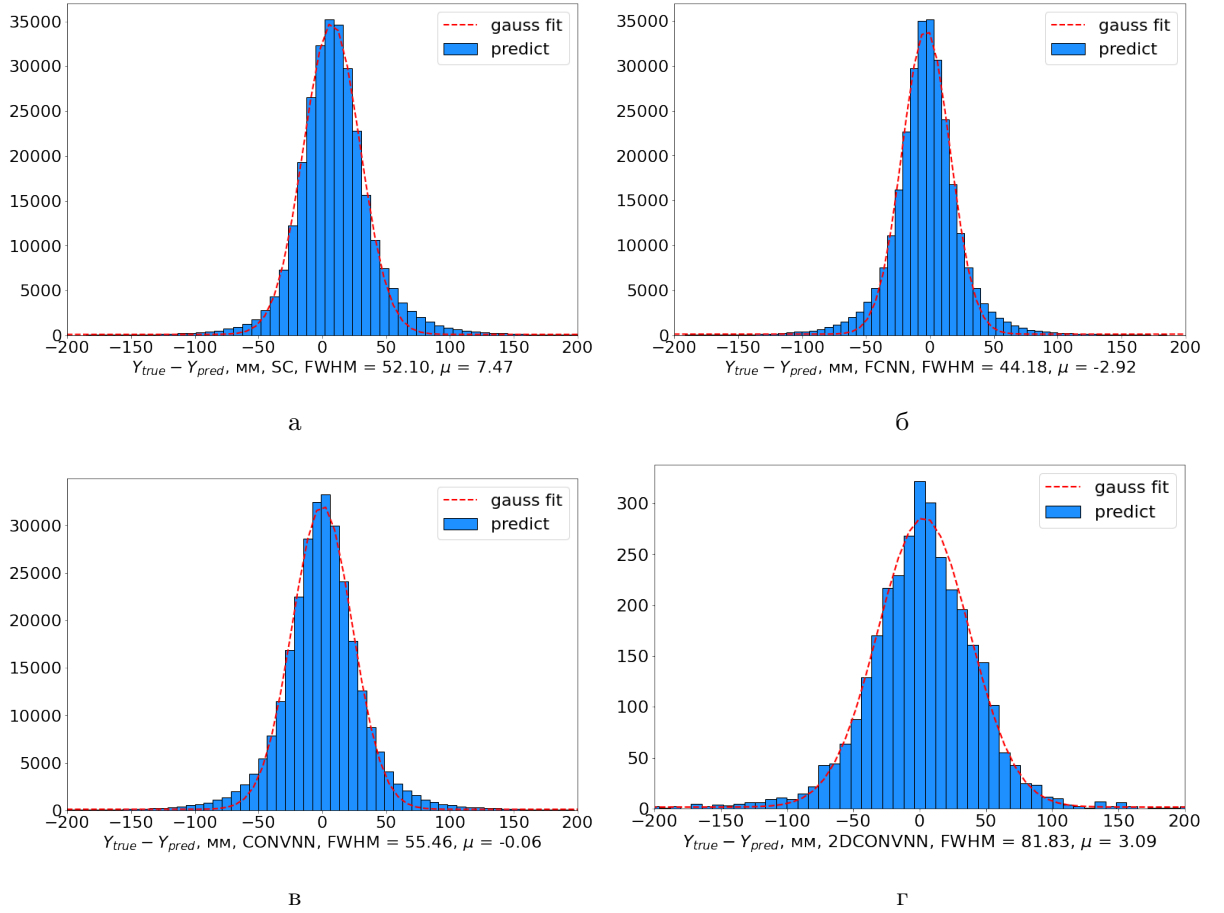


Рисунок 2.11 — Ошибка в определении координаты Y для модели на основе (а) — нейронной сети с короткими связями, (б) — полносвязной нейронной сети, (в) — свёрточной нейронной сети и (г) — двумерной свёрточной нейронной сети

связано с тем, что события внутри детектора распределяются равномерно во время Монте-Карло моделирования. Таким образом, рабочий объем детектора был разделен на 20 областей равного объема. Модели также тренировались на событиях из полного диапазона $R < 850$ мм, а тестировались на событиях из каждой оболочки. В результате были построены boxplot-графики для каждого диапазона для всех трех моделей машинного обучения, а также для алгоритмов MBLikelihood и TimeFit2 (рисунки 2.17, 2.18). Для каждого региона были вычислены значения FWHM и μ для каждой модели. Результаты представлены на рисунке 2.19.

Третий способ — проверка эффективности моделей на различных энергетических диапазонах. Для этого весь регистрируемый интервал от 0 до 4000 PE был разделен на полосы шириной 100 PE и проводилась аналогичная проверка: модели тренировались на полном энергетическом диапазоне, а тестировались только на событиях из заданного интервала PE. Результат проверки представ-

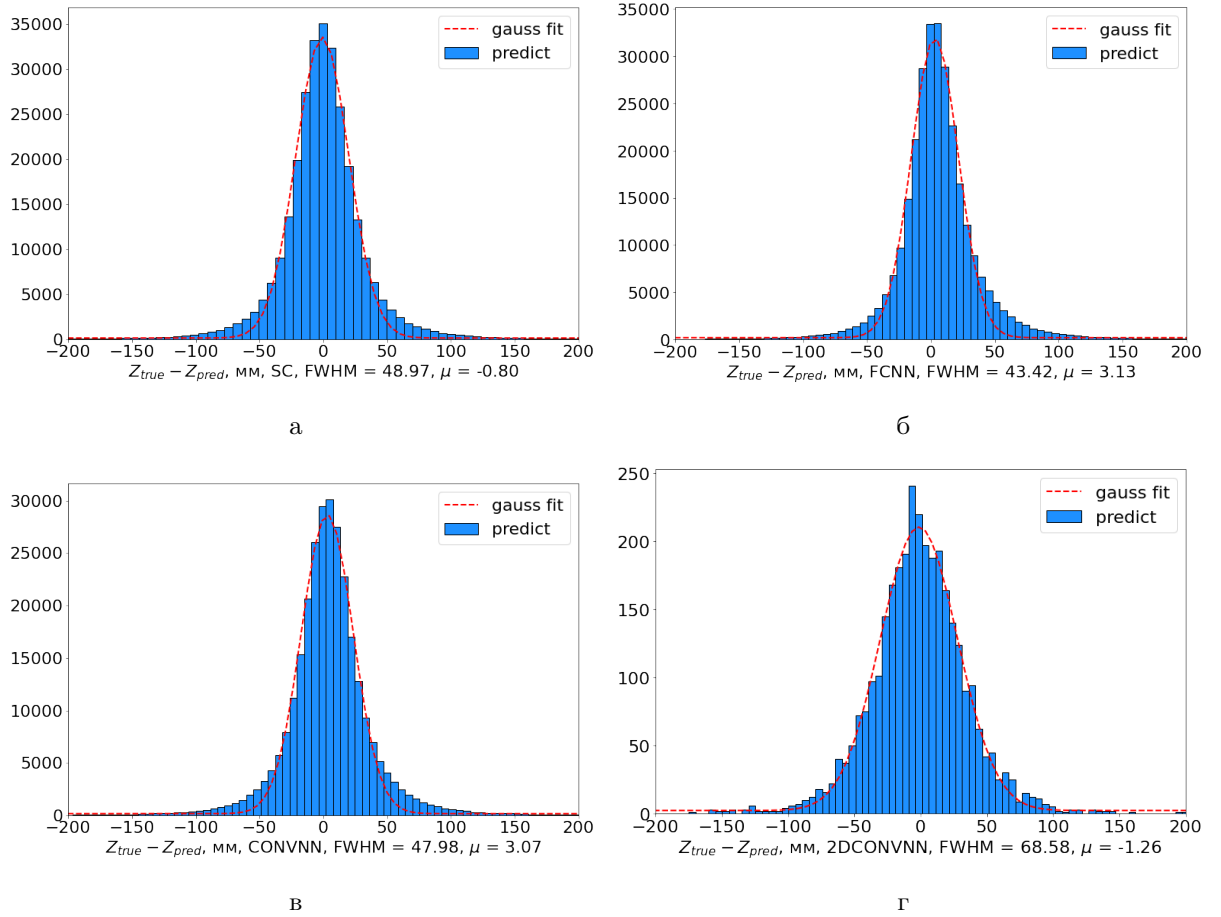


Рисунок 2.12 — Ошибка в определении координаты X для модели на основе (а) — нейронной сети с короткими связями, (б) — полносвязной нейронной сети, (в) — свёрточной нейронной сети и (г) — двумерной свёрточной нейронной сети

лен на рисунке 2.20.

На графиках видно, что любая из моделей машинного обучения лучше алгоритма TimeFit2 и сравнима по эффективности реконструкции с моделью MBLikelihood, лишь немного уступая ей на малых значениях энергии регистрируемых частиц. Такой результат объясняется тем, что малая энергия события ведет к уменьшению общего числа фотоэлектронов, что, в свою очередь, ведет к менее различимому распределению фотоэлектронов в каждом из 255 ФЭУ.

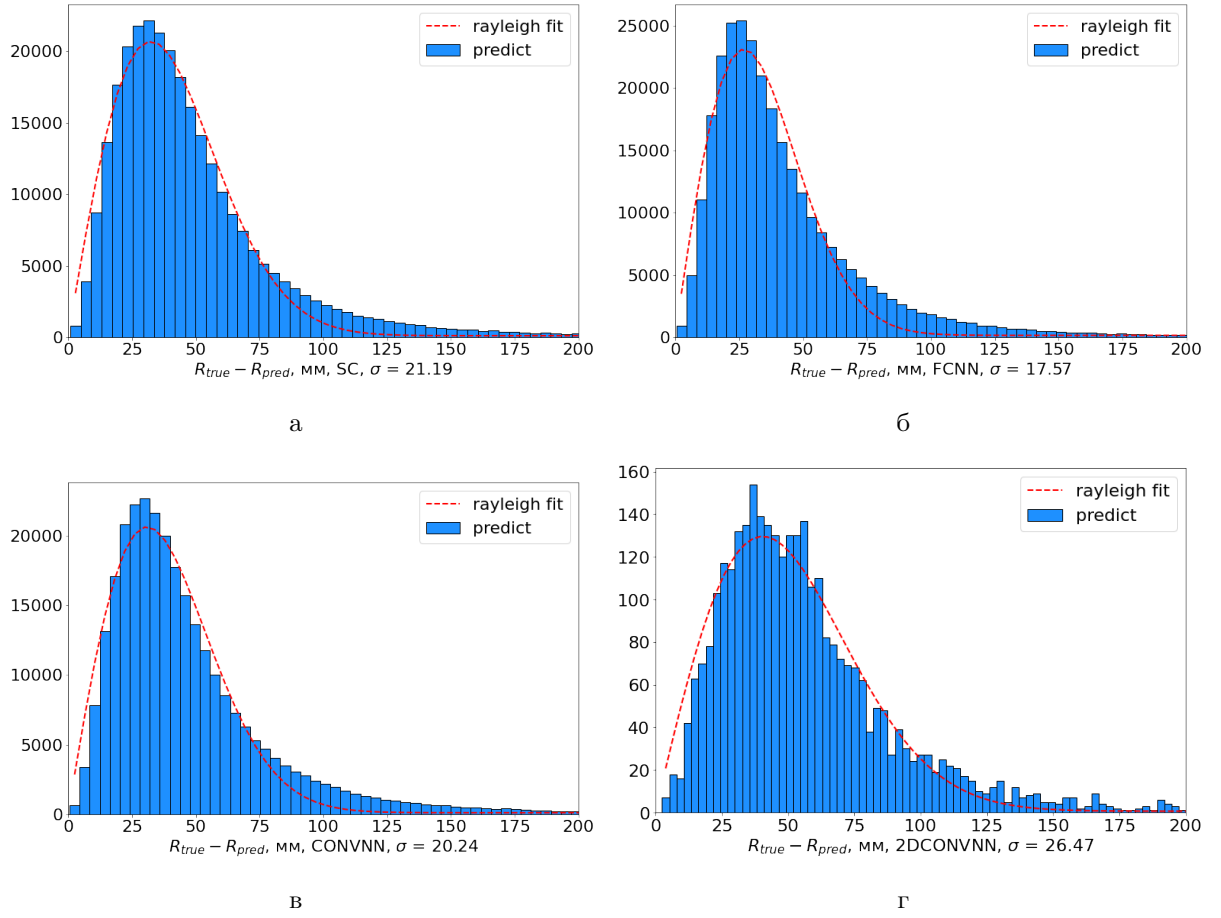


Рисунок 2.13 — Ошибка в определении положения R для модели на основе (а) — нейронной сети с короткими связями, (б) — полносвязной нейронной сети, (в) — свёрточной нейронной сети и (г) — двумерной свёрточной нейронной сети

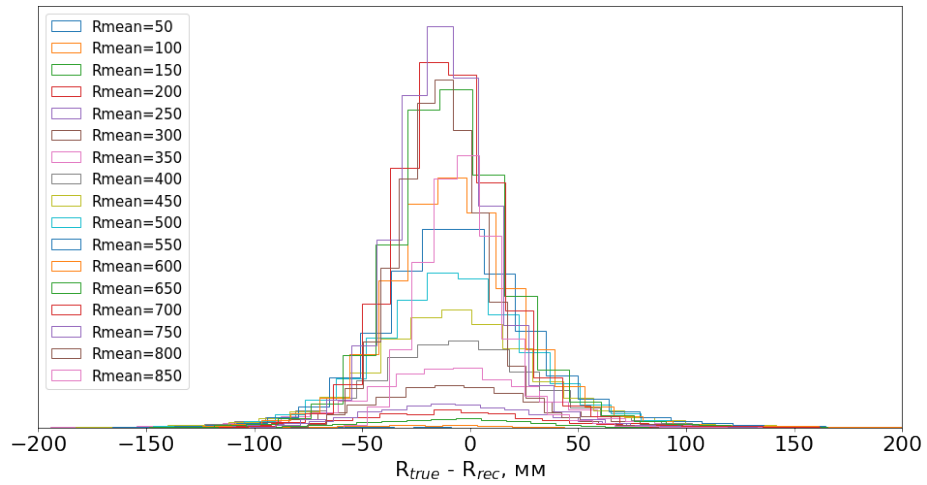


Рисунок 2.14 — Распределение ошибки в реконструкции положения для 20 диапазонов детектора

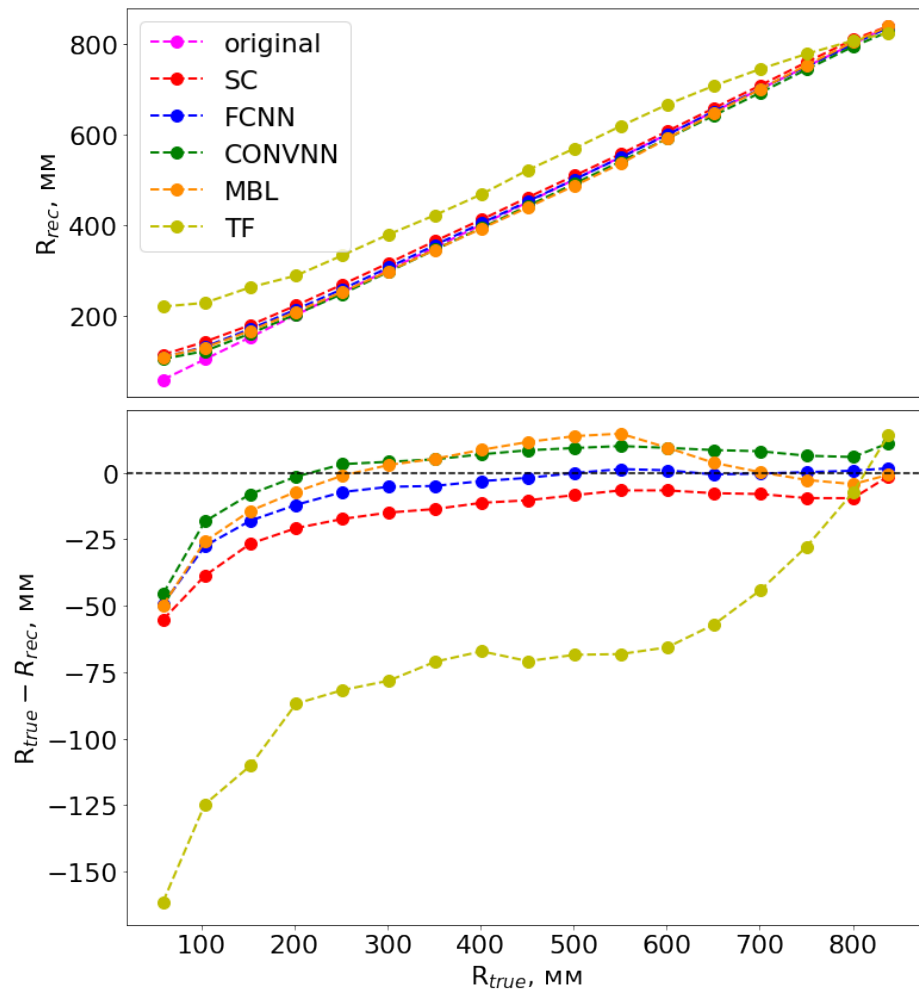


Рисунок 2.15 — Зависимость реконструированного среднего расстояния от истинного расстояния от центра детектора для трех моделей и существующих алгоритмов

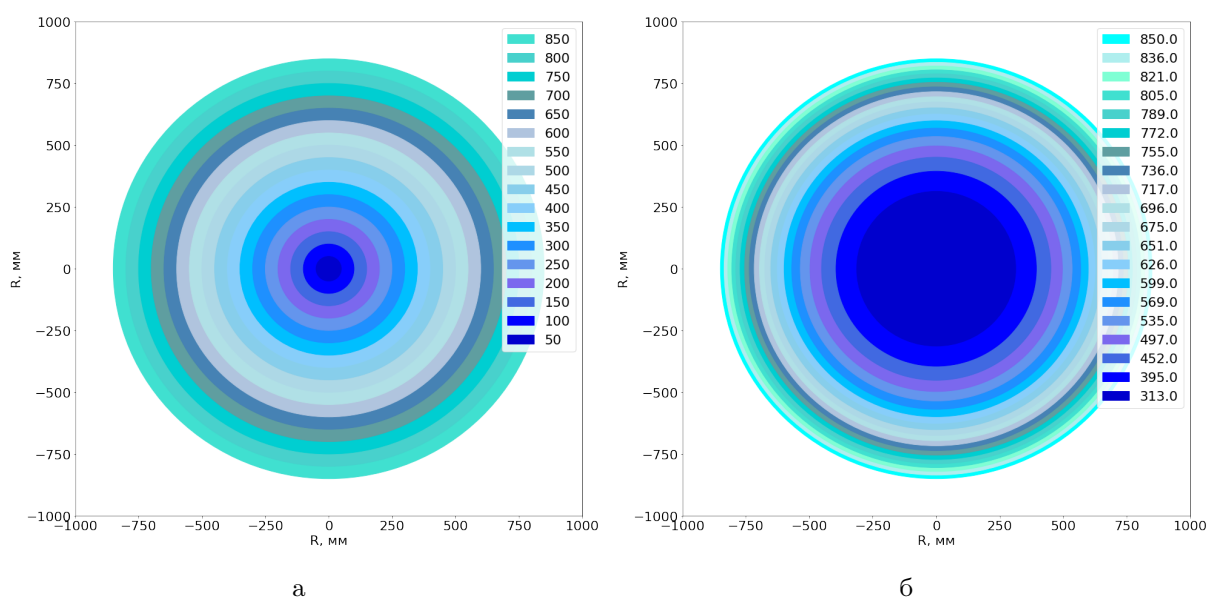


Рисунок 2.16 — Разделение объема детектора на (а) — оболочки равной толщины, (б) — оболочки равного объема

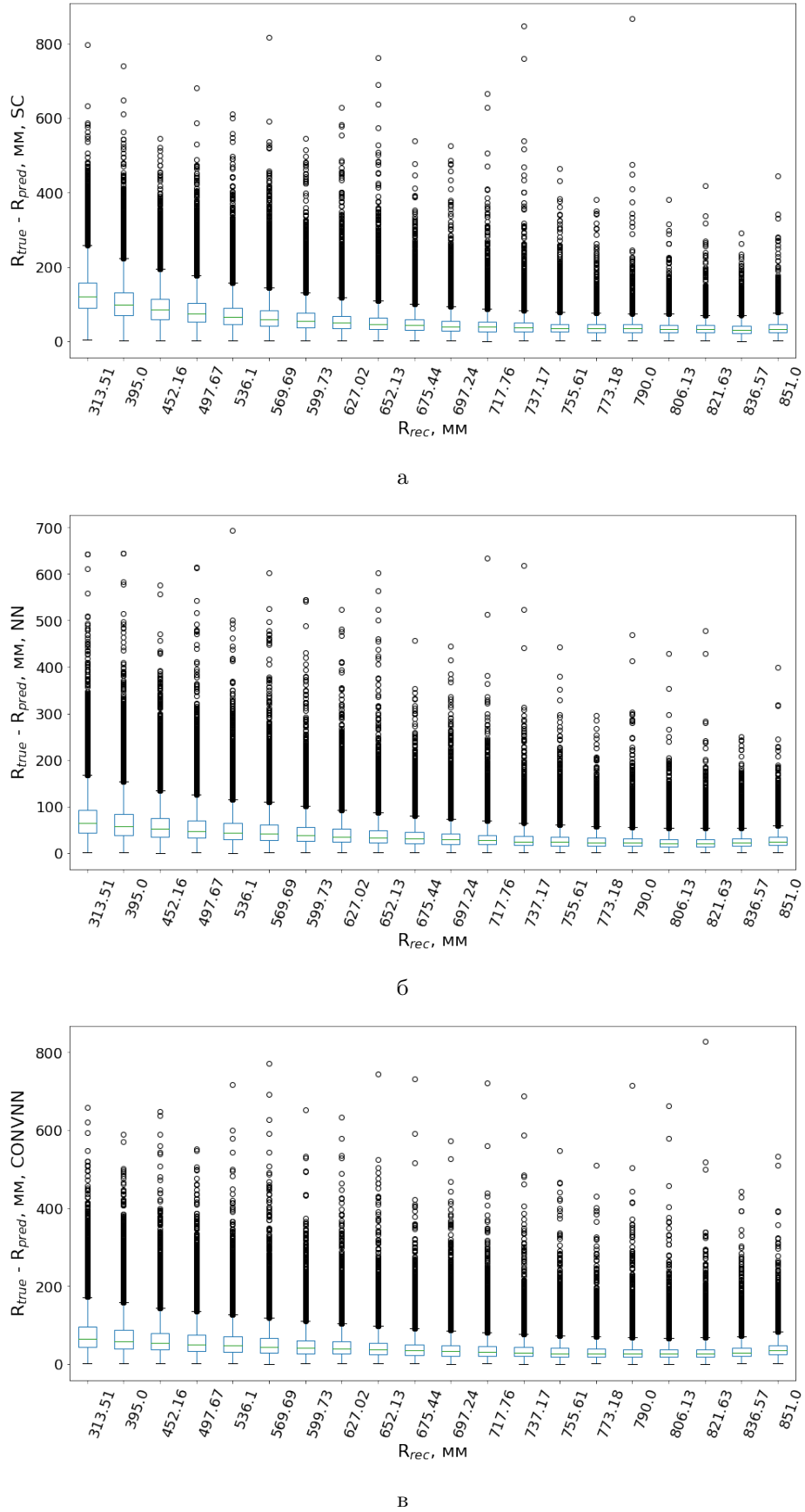
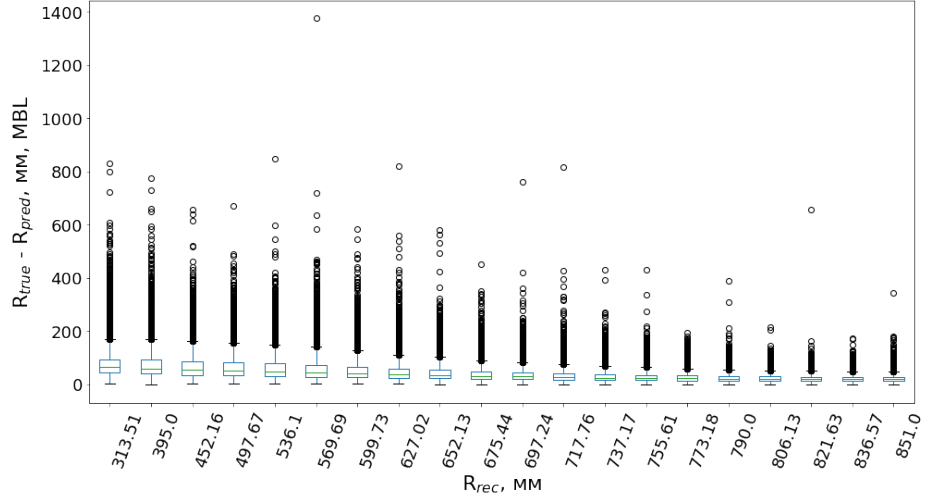
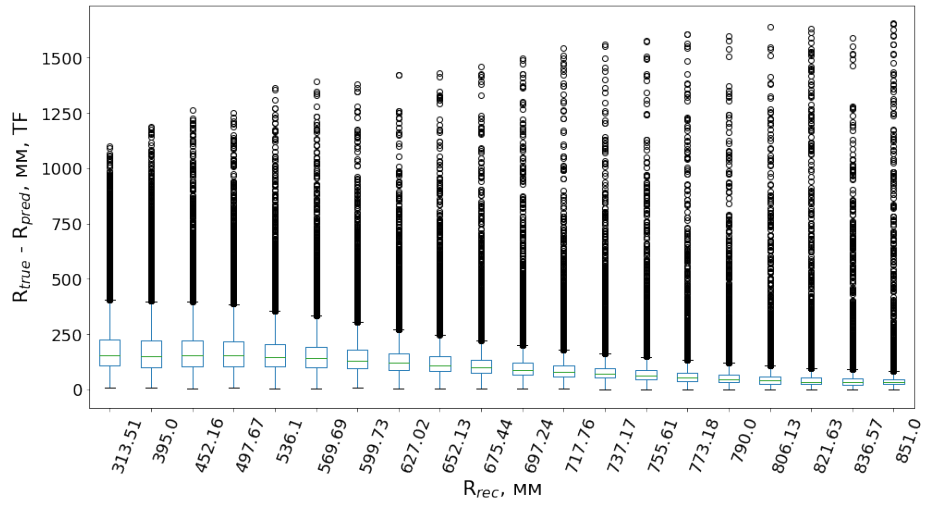


Рисунок 2.17 — Вохplot-графики ошибок реконструкции положения события (а) — нейронной сети с короткими связями, (б) — полносвязной нейронной сети, (в) — свёрточной нейронной сети

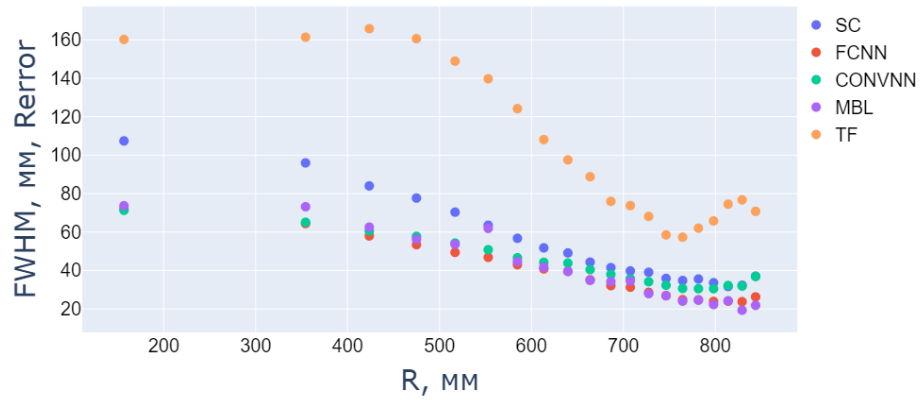


a

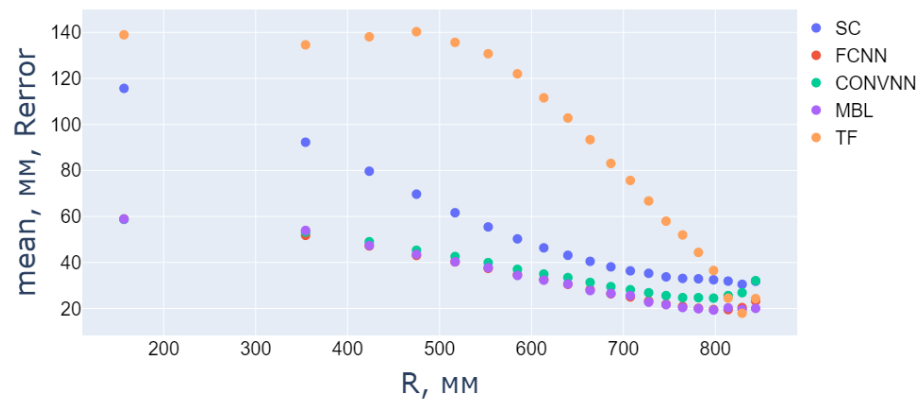


б

Рисунок 2.18 — Вохplot-графики ошибок реконструкции положения события (a) — алгоритма MBLikelihood, (б) — алгоритма TimeFit2

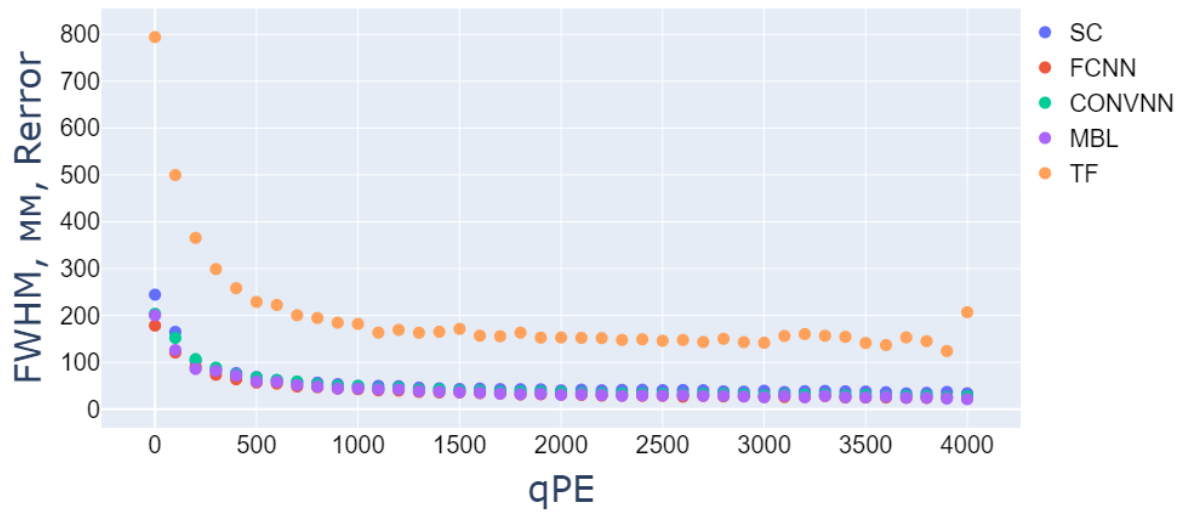


а

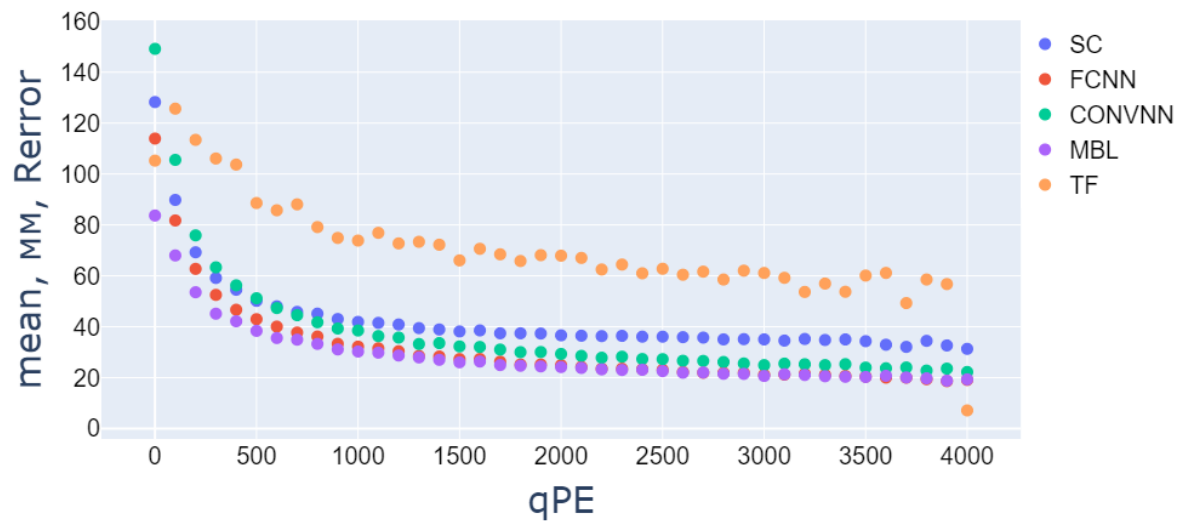


б

Рисунок 2.19 — Эффективность моделей на 5%-ых оболочках: (а) — разрешение, (б) — средняя ошибка алгоритма



a



б

Рисунок 2.20 — Эффективность моделей на энергетических интервалах 100 PE:
 (а) — разрешение, (б) — средняя ошибка алгоритма

ЗАКЛЮЧЕНИЕ

С помощью Монте-Карло моделирования были созданы данные естественной радиоактивности ^{39}Ar . Было проверено четыре модели машинного обучения (приложение А): полносвязная нейронная сеть, нейронная сеть с короткими связями, одномерная и двумерная свёрточная нейронная сеть. Модели выполняют свою задачу по реконструкции положения события (рисунок 2.9). Были проверены различные области детектора (таблица 2.1) и модели лучше всего работают во всем рабочем объеме $R < 850$ мм (таблица 2.2). Положение пика ошибки приходится на $\mu = 6$ мм с разрешением $\text{FWHM} = 50$ мм для 76% всех реконструируемых событий (рисунки 2.10, 2.11, 2.12, таблица 2.3). Хуже всего реконструируются события во внутренней части детектора ($R < 430$ мм). Эта проблема связана с недостаточной статистической выборкой в этой области, в связи с чем события, которые практически не отличаются по распределению фотоэлектронов плохо различаются моделью.

Тестирование моделей на равных оболочках детектора шириной 50 мм показало, что алгоритмы расходятся с теоретическими значениями во внутренней области $R < 200$ мм (рисунок 2.15). Такой результат связан с недостаточной статистикой в указанном регионе (рисунок 2.14).

Тестирование моделей на различных оболочках детектора разной ширины но одинакового объема показало, что в каждой из выбранной области модели на основе нейронной сети значительно превосходят алгоритм TimeFit2 и соотносятся по эффективности с алгоритмом MBLikelihood (рисунки 2.17, 2.18). Отметим, что количество выбросов (то есть событий, ошибка в реконструкции которых значительно превышает разрешение алгоритма, указаны черными точками на рисунках 2.17, 2.18) у алгоритмов на основе нейронных сетей во внутренней части детектора значительно меньше чем у MBLikelihood и TimeFit2. Эффективность моделей подтверждается графиками на рисунке 2.19.

Также эффективность реконструкции исследуемых моделей подтверждается графиками на рисунке 2.20. Рост средней ошибки и разрешения реконструкции моделей при уменьшении энергии события объясняется тем, что низ-

кое значение энергии события ведет к меньшему числу фотоэлектронов и к менее различимому шаблону события в этой энергетической области.

Указанные недостатки в низкоэнергетичной области и внутреннем регионе детектора устраняются с помощью увеличения количества событий в этих регионах в тренировочном наборе с целью более точной тренировки моделей. Поставленная задача выполнена в полном объеме: исследуемые модели работоспособны во всём объеме детектора, работают не менее эффективно чем существующие алгоритмы, и могут использоваться совместно с ними для реконструкции положения события.

СПИСОК ЛИТЕРАТУРЫ

1. *Zwicky F.* On the Masses of Nebulae and of Clusters of Nebulae // The Astrophysical Journal. — 1937. — Vol. 86. — P. 217.
2. *Зельдович Я. Б.* Теория тяготения и эволюция звезд. — Рипол Классик, 1971.
3. *Черепашук А.* Гравитационное микролинзирование и проблема скрытой массы // Соросовский образовательный журнал. — 1998. — № 3. — с. 92—99.
4. Planck 2018 results-VI. Cosmological parameters / N. Aghanim [et al.] // Astronomy & Astrophysics. — 2020. — Vol. 641. — A6.
5. *Горбунов Д. С., Рубаков В. А.* Введение в теорию ранней Вселенной. Теория горячего Большого взрыва. — Российская акад. наук, Ин-т ядерных исслед., 2007.
6. Antiproton flux, antiproton-to-proton flux ratio, and properties of elementary particle fluxes in primary cosmic rays measured with the alpha magnetic spectrometer on the international space station / M. Aguilar [et al.] // Physical Review Letters. — 2016. — Vol. 117, no. 9. — P. 091103.
7. Cosmic-ray positron energy spectrum measured by PAMELA / O. Adriani [et al.] // Physical review letters. — 2013. — Vol. 111, no. 8. — P. 081102.
8. First-year Wilkinson Microwave Anisotropy Probe (WMAP)* observations: determination of cosmological parameters / D. N. Spergel [et al.] // The Astrophysical Journal Supplement Series. — 2003. — Vol. 148, no. 1. — P. 175.
9. Planck 2018 results / P. Collaboration [et al.] // A&A. — 2020. — Vol. 641. — A12.
10. *Hooper D., Goodenough L.* Dark matter annihilation in the Galactic Center as seen by the Fermi Gamma Ray Space Telescope // Physics Letters B. — 2011. — Vol. 697, no. 5. — P. 412–428.

11. The CRESST dark matter search / M. Bravin [et al.] // *Astroparticle Physics*. — 1999. — Vol. 12, no. 1/2. — P. 107–114.
12. CoGeNT: A search for low-mass dark matter using p-type point contact germanium detectors / C. E. Aalseth [et al.] // *Physical Review D*. — 2013. — Vol. 88, no. 1. — P. 012002.
13. Final results of the EDELWEISS-I dark matter search with cryogenic heat-and-ionization Ge detectors / V. Sanglard [et al.] // *Physical Review D*. — 2005. — Vol. 71, no. 12. — P. 122002.
14. DarkSide-50 532-day dark matter search with low-radioactivity argon / P. Agnes [et al.] // *Physical Review D*. — 2018. — Vol. 98, no. 10. — P. 102006.
15. Design and construction of the DEAP-3600 dark matter detector / P.-A. Amaudruz [et al.] // *Astroparticle Physics*. — 2019. — Vol. 108. — P. 1–23.
16. *Zani A.* The WArP experiment: A double-phase argon detector for dark matter searches // *Advances in High Energy Physics*. — 2014. — Vol. 2014.
17. The XENON dark matter search experiment / E. Aprile [et al.] // *New Astronomy Reviews*. — 2005. — Vol. 49, no. 2–6. — P. 289–295.
18. *Evans N. W., O'Hare C. A., McCabe C.* Refinement of the standard halo model for dark matter searches in light of the Gaia Sausage // *Physical Review D*. — 2019. — Vol. 99, no. 2. — P. 023012.
19. Direct Detection of Dark Matter—APPEC Committee Report / J. Billard [et al.] // *arXiv preprint arXiv:2104.07634*. — 2021.
20. Review of particle physics / P. D. Group [et al.] // *Progress of Theoretical and Experimental Physics*. — 2020. — Vol. 2020, no. 8. — P. 083C01.
21. Machine learning at the energy and intensity frontiers of particle physics / A. Radovic [et al.] // *Nature*. — 2018. — Vol. 560, no. 7716. — P. 41–48.
22. *Glaser D. A.* Some effects of ionizing radiation on the formation of bubbles in liquids // *Physical Review*. — 1952. — Vol. 87, no. 4. — P. 665.
23. *Evans L., Bryant P.* LHC machine // *Journal of instrumentation*. — 2008. — Vol. 3, no. 08. — S08001.

24. The LHCb detector at the LHC / A. A. Alves Jr [et al.] // Journal of instrumentation. — 2008. — Vol. 3, no. 08. — S08005.
25. The ATLAS experiment at the CERN large hadron collider / G. Aad [и др.] // Journal of instrumentation. — 2008. — т. 3. — S08003.
26. The CMS experiment at the CERN LHC / S. Chatrchyan [et al.] // Journal of instrumentation. — 2008. — Vol. 3, no. 8. — S08004.
27. *Bhat P. C.* Multivariate analysis methods in particle physics // Annual Review of Nuclear and Particle Science. — 2011. — Vol. 61. — P. 281–309.
28. *Rosenblatt F.* Principles of neurodynamics. perceptrons and the theory of brain mechanisms : tech. rep. / Cornell Aeronautical Lab Inc Buffalo NY. — 1961.
29. *Reed R., MarksII R. J.* Neural smithing: supervised learning in feedforward artificial neural networks. — Mit Press, 1999.
30. Classification and regression trees / L. Breiman [et al.]. — CRC press, 1984.
31. *Freund Y., Schapire R. E.* A decision-theoretic generalization of on-line learning and an application to boosting // Journal of computer and system sciences. — 1997. — Vol. 55, no. 1. — P. 119–139.
32. *Peterson C., Rögnvaldsson T.* An introduction to artificial neural networks : tech. rep. / CERN. — 1991.
33. Determination of $—V_{ub}—$ from the measurement of the inclusive charmless semileptonic branching ratio of b hadrons / R. Barate [et al.] // The European Physical Journal C-Particles and Fields. — 1999. — Vol. 6, no. 4. — P. 555–574.
34. A measurement of the production of $D^{*\pm}$ mesons on the Z^0 resonance / R. Akers [et al.] // Zeitschrift für Physik C Particles and Fields. — 1995. — Vol. 67, no. 1. — P. 27–44.
35. Higgs search by neural networks at LHC / P. Chiappetta [et al.] // Physics Letters B. — 1994. — Vol. 322, no. 3. — P. 219–223.
36. *Peterson C., Rögnvaldsson T., Lönnblad L.* JETNET 3.0—A versatile artificial neural network package // Computer Physics Communications. — 1994. — Vol. 81, no. 1/2. — P. 185–220.

37. Measurement of the tau polarisation at the Z resonance / D. Buskulic [et al.] // Zeitschrift für Physik C Particles and Fields. — 1993. — Vol. 59, no. 3. — P. 369–386.
38. *Babbage W. S., Thompson L. F.* The use of neural networks in γ - π^0 discrimination // Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. — 1993. — Vol. 330, no. 3. — P. 482–486.
39. *Lönnblad L., Peterson C., Rögnerdsson T.* Pattern recognition in high energy physics with artificial neural networks—JETNET 2.0 // Computer Physics Communications. — 1992. — Vol. 70, no. 1. — P. 167–182.
40. *Lönnblad L., Peterson C., Rögnerdsson T.* Using neural networks to identify jets // Nuclear Physics B. — 1991. — Vol. 349, no. 3. — P. 675–702.
41. *Lönnblad L., Peterson C., Rögnerdsson T.* Finding gluon jets with a neural trigger // Physical review letters. — 1990. — Vol. 65, no. 11. — P. 1321.
42. *Denby B.* Neural networks and cellular automata in experimental high energy physics // Computer Physics Communications. — 1988. — Vol. 49, no. 3. — P. 429–448.
43. Boosted decision trees as an alternative to artificial neural networks for particle identification / B. P. Roe [et al.] // Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. — 2005. — Vol. 543, no. 2/3. — P. 577–584.
44. Mastering the game of Go with deep neural networks and tree search / D. Silver [et al.] // nature. — 2016. — Vol. 529, no. 7587. — P. 484–489.
45. Imagenet large scale visual recognition challenge / O. Russakovsky [et al.] // International journal of computer vision. — 2015. — Vol. 115, no. 3. — P. 211–252.
46. *Grobov A., Ilyasov A.* Boosted decision trees approach to neck alpha events discrimination in DEAP-3600 experiment // Physica Scripta. — 2020. — Vol. 95, no. 7. — P. 074007.
47. *Collaboration D.* Search for dark matter with a 231-day exposure of liquid argon using DEAP-3600 at SNOLAB // Physical Review D. — 2019. — Vol. 100, no. 2. — P. 022004.

48. Tetraphenyl-butadiene films: VUV-Vis optical characterization from room to liquid argon temperature / R. Francini [et al.] // Journal of Instrumentation. — 2013. — Vol. 8, no. 09. — P. C09010.
49. Electromagnetic backgrounds and potassium-42 activity in the DEAP-3600 dark matter detector / R. Ajaj [et al.] // Physical Review D. — 2019. — Vol. 100, no. 7. — P. 072009.
50. *Sinnock A., Smith B.* Refractive indices of the condensed inert gases // Physical Review. — 1969. — Vol. 181, no. 3. — P. 1297.
51. Index of refraction, Rayleigh scattering length, and Sellmeier coefficients in solid and liquid argon and xenon / E. Grace [et al.] // Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. — 2017. — Vol. 867. — P. 204–208.
52. *Hopfield J. J.* Neural networks and physical systems with emergent collective computational abilities // Proceedings of the national academy of sciences. — 1982. — Vol. 79, no. 8. — P. 2554–2558.
53. A concatenating framework of shortcut convolutional neural networks / Y. Li [et al.] // arXiv preprint:1710.00974. — 2017.
54. Shortcut learning in deep neural networks / R. Geirhos [et al.] // Nature Machine Intelligence. — 2020. — Vol. 2, no. 11. — P. 665–673.
55. Subject independent facial expression recognition with robust face detection using a convolutional neural network / M. Matsugu [et al.] // Neural Networks. — 2003. — Vol. 16, no. 5/6. — P. 555–559.
56. Backpropagation applied to handwritten zip code recognition / Y. LeCun [et al.] // Neural computation. — 1989. — Vol. 1, no. 4. — P. 541–551.
57. *Ketkar N.* Introduction to keras // Deep learning with Python. — Springer, 2017. — P. 97–111.

Приложение А

Построение моделей нейронной сети с помощью языка программирования Python

Вся работа проводилась с использованием пакета keras [57] языка программирования Python версии 3.7. Ниже представлен код для построения модели на основе А.1 — нейронной сети с короткими связями, А.2 — полносвязной нейронной сети, А.3 — одномерной нейронной сети со свёрточными слоями, А.4 — двумерной нейронной сети со свёрточными слоями.

```
def shortcut_part_1(X, N):
    X = layers.BatchNormalization(name='block_%i'%N)(X)

    X = layers.Dense(units=1000,
                      activation='relu',
                      name='relu_block_%i_1000nodes'%N)(X)
    X = layers.Dense(units=255,
                      activation='relu',
                      name='relu_block_%i_255nodes'%N)(X)
    X = layers.Dense(units=3,
                      activation='relu',
                      name='relu_block_%i_3nodes'%N)(X)
    X = layers.Dense(units=255,
                      activation='relu',
                      name='relu_block_%i_765nodes'%N)(X)

    return X

input_shape = (255,)
```

```

X_input = layers.Input(shape=input_shape,
                        name='Input_765')

N = 1
X1 = shortcut_part_1(X_input, N)
X = layers.add([X1, X_input], name='add_from_Input')
N = 2
X2 = shortcut_part_1(X, N)
X = layers.add([X1, X2], name='add_from_block1')
N = 3
X3 = shortcut_part_1(X, N)
X = layers.add([X2, X3],
               name='add_from_block2')

X = layers.BatchNormalization(name='output_block')(X)
X = layers.Dense(units=255,
                 activation='linear',
                 name='linear_255nodes')(X)

X = layers.Dense(units=3,
                 activation='linear',
                 name='linear_3nodes_output')(X)

SCmodel = models.Model(X_input, X)
SCmodel.compile(loss='MSE',
               optimizer='adam',
               metrics=['accuracy'])

history_SC = SCmodel.fit(x=x_train,
                        y=y_train,
                        epochs=30,
                        batch_size=1000,
                        validation_split=0.3)

```

Фрагмент кода А.1 — Функция построения модели нейронной сети с короткими связями

```

def dense_model():

    model = Sequential()

    model.add(Dense(units=1000,
                     activation='relu',
                     name='Input_765'))

    model.add(Dense(units=765,
                     activation='relu',
                     name='relu_765nodes'))

    model.add(Dense(units=510,
                     activation='relu',
                     name='relu_510nodes'))

    model.add(Dense(units=255,
                     activation='relu',
                     name='relu_255nodes'))

    model.add(Dense(units=3,
                     name='3nodes_output'))

    model.compile(optimizer='adam',
                  loss='mean_squared_error',
                  metrics=['mean_squared_error'])

    return model

model_FCNN = dense_model()

history_FCNN = model_FCNN.fit(x=x_train,
                              y=y_train,
                              epochs=30,
                              batch_size=1000,

```

```
validation_split=0.3)
```

Фрагмент кода A.2 — Функция построения модели полносвязной нейронной сети

```
def convolutional_model():

    model = Sequential()

    model.add(Input(shape=(255,1),
                        name='Input_255'))

    model.add(Conv1D(filters=16,
                      kernel_size=4,
                      strides=2,
                      activation='relu',
                      padding="same",
                      name='relu_16f_k4_s2'))

    model.add(Conv1D(filters=32,
                      kernel_size=4,
                      strides=2,
                      activation='relu',
                      padding="same",
                      name='relu_32f_k4_s2'))

    model.add(Conv1D(filters=64,
                      kernel_size=4,
                      strides=2,
                      activation='relu',
                      padding="same",
                      name='relu_64f_k4_s2'))

    model.add(Conv1D(filters=128,
                      kernel_size=2,
```


Фрагмент кода A.3 — Функция построения модели одномерной свёрточной нейронной сети

```
def convolutional_model_2D():

    model = Sequential()

    model.add(Input(shape=(128, 128, 3),
                        name='128x128x3'))

    model.add(Conv2D(filters=16,
                     kernel_size=(5, 5),
                     activation='relu',
                     padding="same",
                     input_shape=(128, 128, 3),
                     name='16f_5x5k_relu_same'))

    model.add(MaxPooling2D(pool_size=(2, 2),
                          strides=(2, 2),
                          padding="same",
                          name='2x2p_2x2s_1'))

    model.add(Conv2D(filters=64,
                     kernel_size=(2, 2),
                     activation='relu',
                     padding="same",
                     name='64f_2x2k_relu_same'))

    model.add(MaxPooling2D(pool_size=(2, 2),
                          strides=(2, 2),
                          padding="same",
                          name='2x2p_2x2s_2'))

    model.add(Conv2D(filters=128,
                     kernel_size=(2, 2),
```

```

        activation='relu',
        padding="same",
        name='128f_2x2k_relu_same'))

model.add(MaxPooling2D(pool_size=(2, 2),
                        strides=(2, 2),
                        padding="same",
                        name='2x2p_2x2s_3'))

model.add(Flatten(name='flatten'))

model.add(Dense(units=4096,
                activation='relu',
                name='relu_4096nodes'))

model.add(Dense(units=512,
                activation='relu',
                name='relu_512nodes'))

model.add(Dense(units=128,
                activation='relu',
                name='relu_128nodes'))

model.add(Dense(units=32,
                activation='relu',
                name='relu_32nodes'))

model.add(Dense(units=3,
                name='output'))

model.compile(optimizer='adam',
              loss='mean_squared_error',
              metrics=['mean_squared_error'])

return model

```

```

model_conv2d = convolutional_model_2D()

history_2DCONVNN = model_conv2d.fit(x=X_train_images,
                                     y=y_train_CONV,
                                     epochs=30,
                                     batch_size=500,
                                     validation_split=0.3)

```

Фрагмент кода А.4 — Функция построения модели двумерной свёрточной нейронной сети

В коде использовались следующие слои:

- *Input* — используется для создания тензора Keras. Тензор Keras — это символьный тензорный объект, который дополняется определенными атрибутами, которые позволяют нам строить модель Keras, просто зная входные и выходные данные модели.
- *Dense* — полносвязный слой (то есть слой, каждый нейрон которого связан со всеми нейронами следующего слоя). Вывод вычисляется следующим образом:

$$\text{output} = \text{activation}(\text{dot}(\text{input}, \text{kernel}) + \text{bias}),$$

где *activation* — функция активации нейрона (принцип ее работы на рисунке 2.2), *input* — входная матрица, *kernel* — матрица весов, *dot* — скалярное произведение, *bias* — константа, помогающая настроить модель наилучшим образом;

- *BatchNormalization* — Слой, который нормализует свои входы. то есть трансформирует входной набор событий в набор событий со средним значением, близким к 0, и стандартным отклонением близким к 1;
- *Add* — Слой, который суммирует входящие списки. Он принимает в качестве входных данных список тензоров одинаковой формы и возвращает один тензор (такой же формы);
- *Conv1D* — одномерный свёрточный слой. Этот слой создает ядро свертки, которое сворачивается со входным слоем в одном пространственном

измерении для создания тензора выходных данных. Принцип работы одномерного ядра свёрточного слоя представлен на рисунке 2.4. В качестве аргументов этот слой имеет: *filters* — целое число, размерность выходного пространства (количество выходных фильтров в свертке), *kernel_size* — целое число, определяющее длину окна одномерной свертки, *strides* — целое число определяющее длину шага свертки, *activation* — функция активации, *padding* — размер выходного тензора при свёртке, *name* — название слоя;

- *Conv2D* — двумерный свёрточный слой. Этот слой создает ядро свертки, которое сворачивается со входным слоем для создания тензора выходных данных. Принцип работы двумерного ядра свёрточного слоя представлен на рисунке 2.6. В качестве аргументов данный слой имеет те же параметры, что и *Conv1D*;
- *MaxPooling2D* — совершает операцию максимального объединения для двумерных пространственных данных (принцип работы представлен на рисунке A.1). Уменьшает размер входных данных по его пространственным размерам (высоте и ширине), принимая максимальное значение в окне ввода (размер которого определяется параметром *pool_size*) для каждого канала входа. Окно перемещается шагами по каждому измерению, размер шагов задан параметром *strides*;

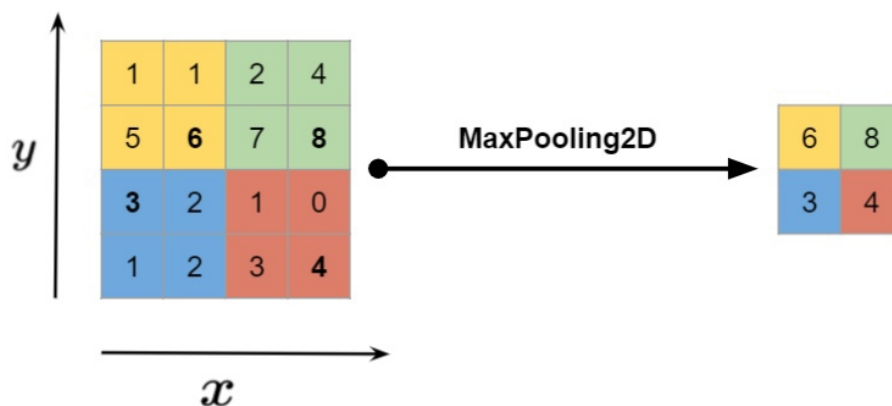


Рисунок A.1 — Принцип работы слоя MaxPooling2D двумерной свёрточной нейронной сети

- *Flatten* — Сглаживает ввод, трансформируя матрицу в вектор. Применяется в свёрточной нейронной сети перед использованием полносвязной нейронной сети.

После создания модели у нее настраиваются метрики и функция потерь

(значение которой будет минимизироваться в процессе обучения) с помощью `model.compile()`, обучается с помощью `model.fit()` и используется для предсказания с помощью `model.predict()`.

Для наглядной демонстрации исследуемые модели нейронной сети (фрагменты кода [A.1](#), [A.2](#), [A.3](#), [A.4](#)) представлены на рисунках [A.2](#), [A.3](#) [A.4](#) и [A.5](#). Здесь слева показана схема слоев, справа — параметры слоёв и выходная структура из каждого слоя.

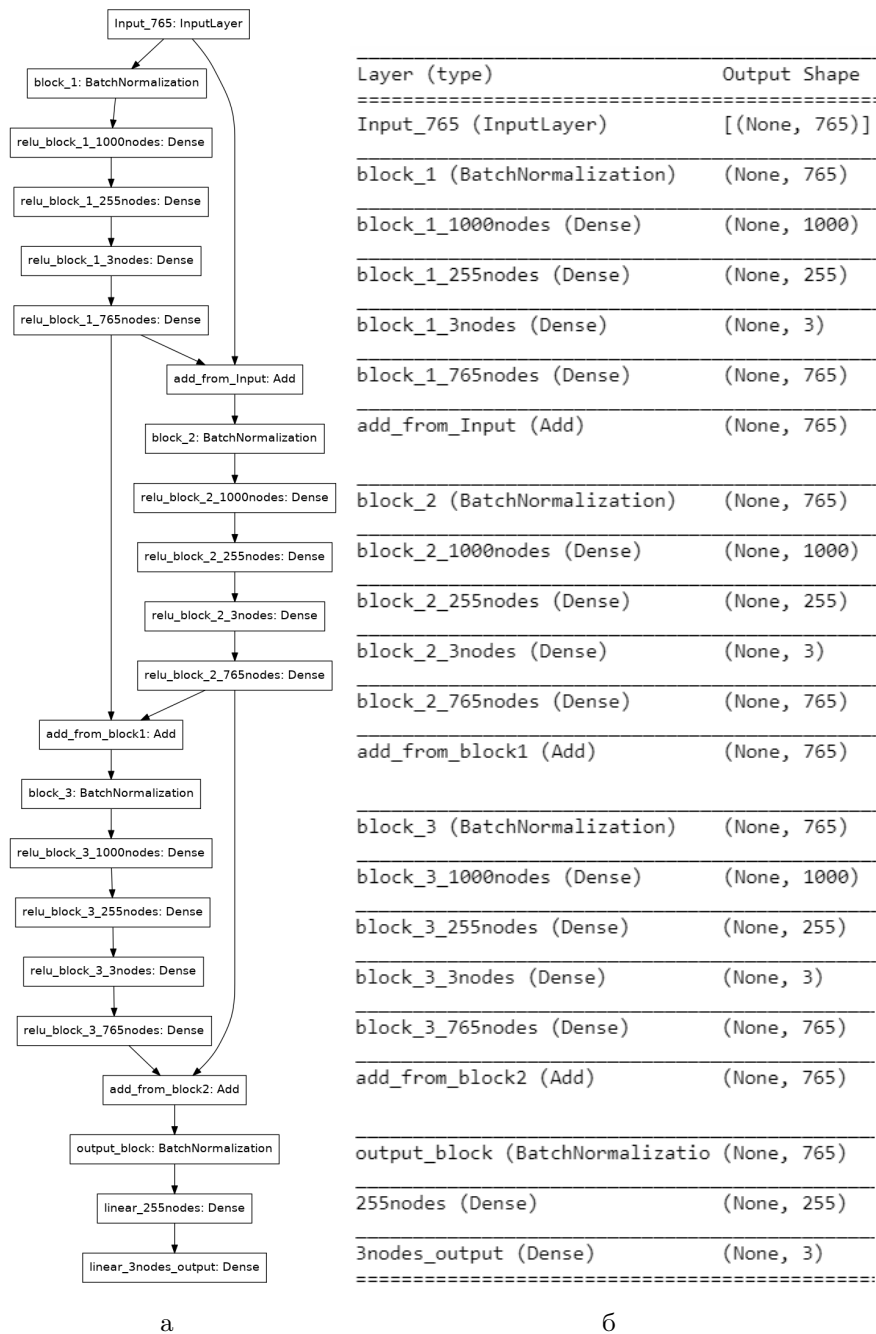


Рисунок A.2 — *Shortcut* нейронная сеть: **а** — схема слоёв, имеющая 3 одинаковых “блока”, **б** — структура данных внутри сети. Слева — параметры слоев, справа — структура выхода из слоя

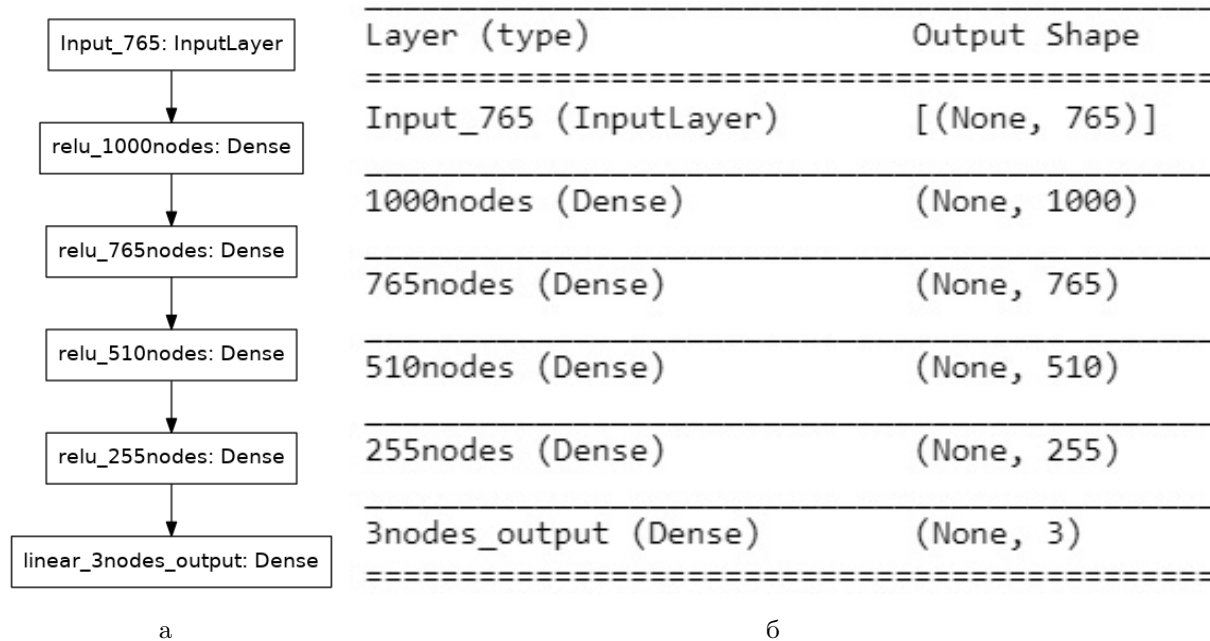


Рисунок А.3 — Полносвязная нейронная сеть: **а** — схема слоёв, **б** — структура данных внутри сети. Слева — параметры слоев, справа — структура выхода из слоя

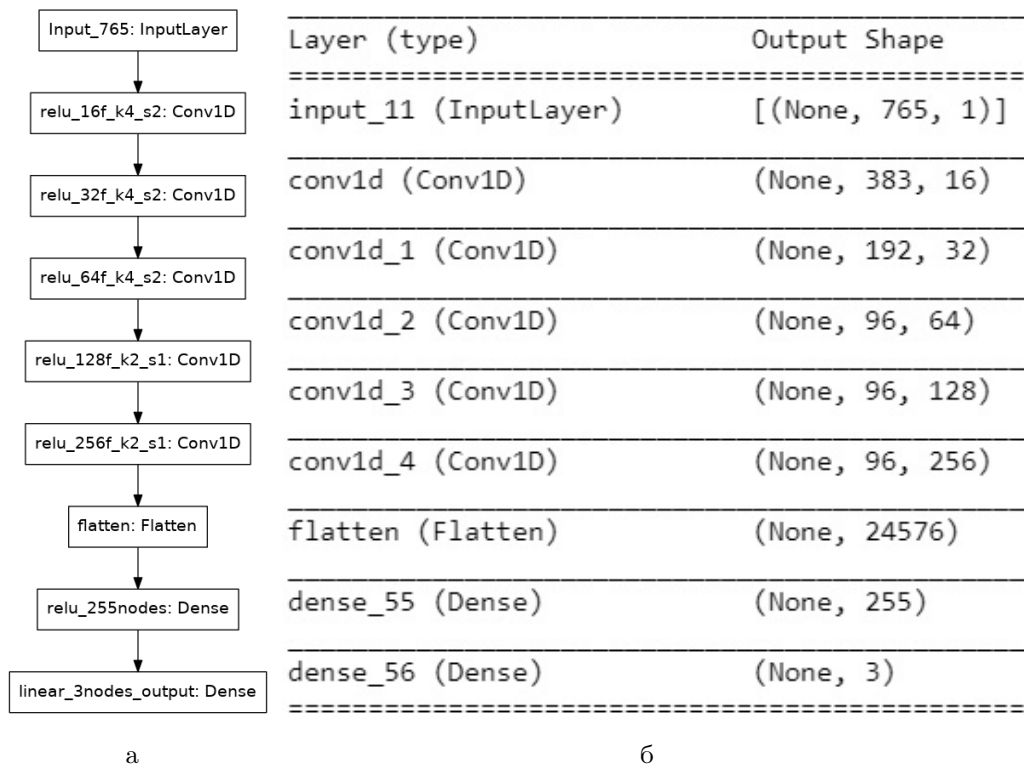


Рисунок А.4 — Нейронная сеть со свёрточными слоями: **а** — схема слоёв, **б** — структура данных внутри сети. Слева — параметры слоев, справа — структура выхода из слоя

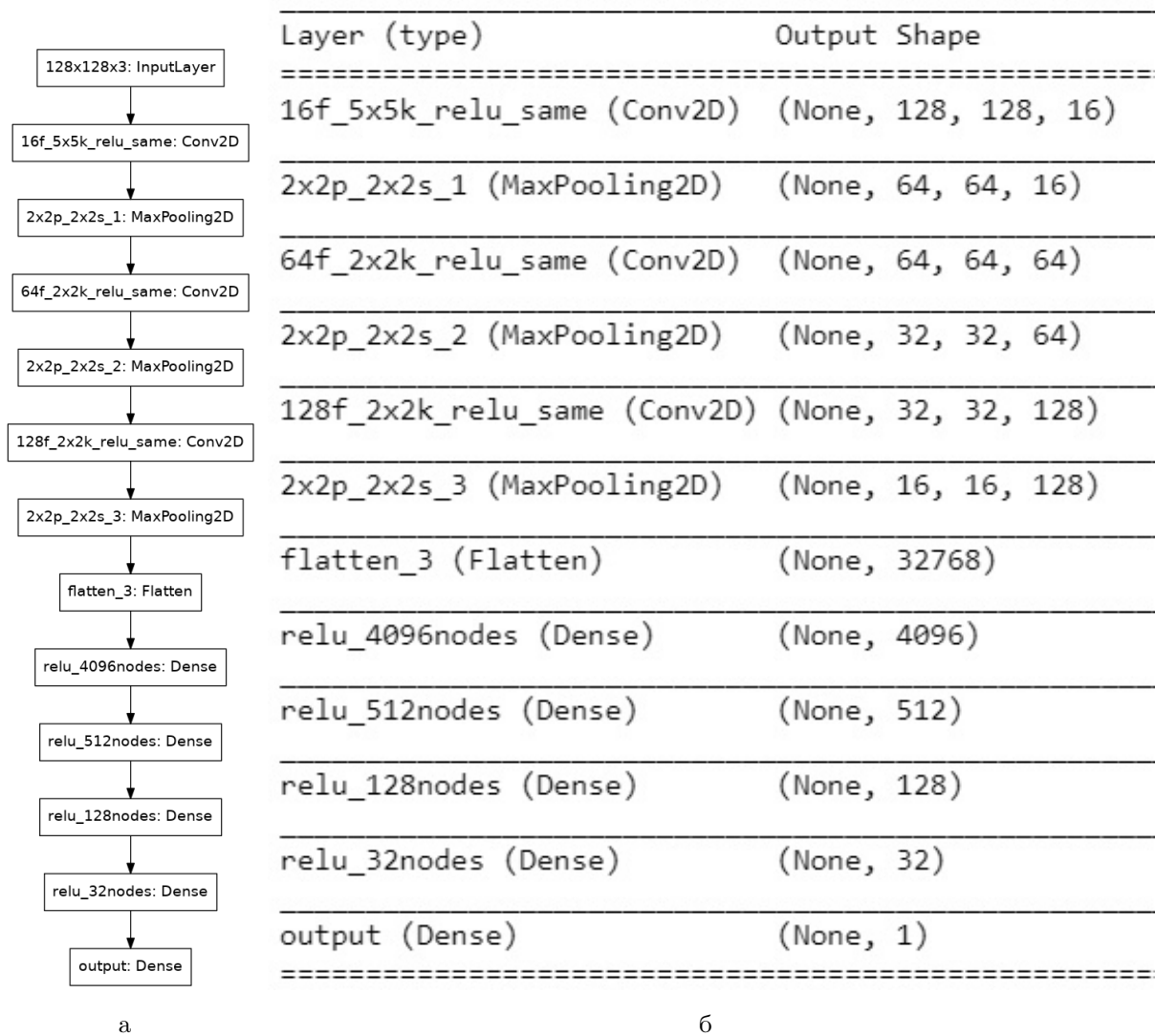


Рисунок А.5 — Двумерная нейронная сеть со свёрточными слоями: **а** — схема слоёв, **б** — структура данных внутри сети. Слева — параметры слоев, справа — структура выхода из слоя