|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство образования и науки Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ Радиотехнический \_

КАФЕДРА Системы обработки информации и управления

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ***

***НА ТЕМУ:***

***Система реконструкции трёхмерной поверхности человеческого лица на основе фотографий***

Студент \_\_РТ5-81Б\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_Л. В. Забурунов\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель ВКР **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_Д. В. Семёнов\_\_\_**

(Подпись, дата) (И.О.Фамилия)

Нормоконтролёр **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_Ю. Н. Кротов \_\_**

(Подпись, дата) (И.О.Фамилия)

2022 г.

# РЕФЕРАТ

Расчётно-пояснительная записка выпускной квалификационной работы бакалавра содержит 67 страниц (с приложениями – 96 страниц). В работе приведены 9 рисунков, 3 таблицы, 23 формулы и использовано 54 источника. В конце работы находится три приложения.

Целью работы является создание системы, оценивающей трёхмерную поверхность человеческого лица с помощью одной и только одной фотографии. Выходными данными должна являться трёхмерная модель, представленная в одном из общеиспользуемых форматов.

В рамках ВКР была изучена тема лицевой реконструкции в условиях недостаточных входных данных, то есть при невозможности использовать технологии трёхмерной оценки поверхности с помощью аппаратных средств, с использованием методов глубокого обучения.

Расчётно-пояснительная записка состоит из трёх частей.

Первый раздел – «Постановка задач» – описывает предметную область, предпосылки для построения системы, а также требования к её возможностям.

Во втором разделе – «Исследовательская часть» – рассмотрены основные подходы к построению систем лицевой реконструкции.

Третий раздел – «Конструкторско-технологическая часть» – посвящён непосредственно создаваемой системе.

Результатом выполненной работы является графическое приложение для персональных компьютеров, осуществляющее обработку двумерного изображения и создание трёхмерной модели на основе лица на этом изображении.

# **СОДЕРЖАНИЕ**

[РЕФЕРАТ 2](#_Toc106718397)

[СПИСОК ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ 4](#_Toc106718398)

[ВВЕДЕНИЕ 5](#_Toc106718399)

[1. ПОСТАНОВКА ЗАДАЧ РАЗРАБОТКИ 7](#_Toc106718400)

[1.1 Описание предметной области 7](#_Toc106718401)

[1.2 Постановка задач проектирования 7](#_Toc106718402)

[1.3 Требования к разрабатываемой системе 8](#_Toc106718403)

[2. ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ 10](#_Toc106718404)

[2.1 Подходы к трёхмерной реконструкции лица человека 10](#_Toc106718405)

[2.1.1 Аналитические методы реконструкции 11](#_Toc106718406)

[2.1.2 Статистические методы реконструкции 15](#_Toc106718407)

[2.1.3 Нейросетевые методы реконструкции 18](#_Toc106718408)

[2.2 Итоги анализа 22](#_Toc106718409)

[3. КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ 26](#_Toc106718410)

[3.1 Архитектура проектируемой системы 27](#_Toc106718411)

[3.2 Набор данных для обучения и проверки 29](#_Toc106718412)

[3.3 Описание алгоритма трёхмерной реконструкции 32](#_Toc106718413)

[3.4 Функции потерь 40](#_Toc106718414)

[3.5 Используемые программные инструменты 46](#_Toc106718415)

[3.6 Создание приложения с графическим интерфейсом 49](#_Toc106718416)

[3.7 Результаты работы 53](#_Toc106718417)

[3.8 Выводы 58](#_Toc106718418)

[ЗАКЛЮЧЕНИЕ 60](#_Toc106718419)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 61](#_Toc106718420)

[ПРИЛОЖЕНИЕ А 68](#_Toc106718421)

[ПРИЛОЖЕНИЕ B 88](#_Toc106718422)

[ПРИЛОЖЕНИЕ C 93](#_Toc106718423)

# СПИСОК ОБОЗНАЧЕНИЙ И СОКРАЩЕНИЙ

3D (*three-dimensional*) – трёхмерный объект, определённый в пространстве, а не на плоскости;

SfS (*Shape from Shading*) – группа методов, направленных на получение трёхмерной поверхности на основании градиентов освещённости изображения;

*BFM* (*Basel Face Model*) – базовая 3D-модель лица, впервые представленная в [5] и часто используемая в исследованиях, связанных с темой человеческого лица;

МГК (Метод Главных Компонент) – метод снижения размерности, осуществляющий переход к новому пространству с минимальными дисперсиями вдоль новых осей;

*PCA* (*Principal Component Analysis*) – название МГК в англоязычных материалах;

ИНС – искусственная нейронная сеть;

*ReLU* (*rectified linear unit*) – функция активации нейронной сети, ведущая себя как стандартная линейная функция, но ограничивающая минимальное значение нулём;

*UI* (*user interface*) – элементы пользовательского интерфейса, то есть объекты, реагирующие на действия пользователя (кнопки, текстовые поля, анимированные баннеры и др.);

*BN* (*batch normalization*) – вспомогательный слой нейронной сети, приводящий значения к стандартному нормальному распределениюlenya для ускорения обучения.

# ВВЕДЕНИЕ

Развитие технологий приводит к большим возможностям для обычных пользователей с одной стороны, и к возрастающим требованиям со стороны профессионалов и/или закрытых структур – с другой. Задача трёхмерной реконструкции лица появляется, как правило, в двух областях: дополнительный способ визуализации для погружения пользователя и системы безопасности (идентификация лица).

Несмотря на схожесть всех человеческих лиц на уровне глобальных черт (наличие носа, ушей, бровей и др.), каждое обладает множеством уникальных черт, которые несут в себе самую ценную информацию как с точки зрения визуализации – именно по мелким деталям пользователь будет определять точность полученной модели, – так и с точки зрения идентификации, где каждая черта вносит существенный вклад в оценку соответствия.

В вопросе трёхмерной реконструкции имеется следующее препятствие: те данные, которые естественно считать рабочим материалом – результат трёхмерного сканирования лиц, – являются крайне дорогостоящими, в связи с чем проектирование систем реконструкции усложняется. Идея использования двумерных изображений позволила существенно упростить процесс проектирования, поскольку теперь рабочие материалы можно собирать с помощью поиска в интернете в любом необходимом количестве; вдобавок, если используется глубокое обучение, то для различных задач уже создано множество наборов данных (датасетов) с фотографиями лиц.

Помимо сложностей с данными, существуют также проблемы с постановкой задачи как таковой, и проблема трёхмерной реконструкции до сих пор не имеет строгого решения. Формально ручное построение трёхмерной модели можно считать решением, но речь идёт именно об автоматической системе, которая позволит получать если не готовый, приближенный к идеальному варианту результат, то хотя бы грубую оценку, пригодную для дальнейшей ручной доработки. Решить такую задачу очень сложно в силу высокой дисперсии входных данных: положения лица на фотографии, окружение на фотографии (в том числе потенциальное перекрытие частей лица), условия освещения в помещении и другие факты практически исключают возможность полностью аналитической постановки вопроса о нахождении решений.

В рамках выполнения работы проводится анализ различных подходов к трёхмерной реконструкции лица, а также создаётся программа на базе методов глубокого обучения. Если получение высокоточного алгоритма для бакалаврской работы оказывается маловероятным, то я постараюсь создать прототип, пригодный для дальнейшего развития и приближения к реалистичному виду выходных моделей.

# ПОСТАНОВКА ЗАДАЧ РАЗРАБОТКИ

## Описание предметной области

Предметную область (рис. 1) составляет совокупность объектов, цель которых – получение трёхмерной модели по заданному изображению человеческого лица.

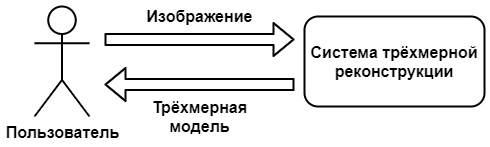


Рисунок 1 – Описание предметной области

В состав предметной области входят следующие сущности:

* Пользователь – человек, выбирающий исходную фотографию;
* Система – программа, обрабатывающая фотографию и отвечающую пользователю готовой к сохранению трёхмерной моделью.

## Постановка задач проектирования

И так, целью работы является исследование возможностей создания системы качественной реконструкции человеческого лица с помощью одной-единственной фотографии, то есть в условиях недостаточных входных данных.

В рамках выполнения выпускной квалификационной работы следует обратить внимание на готовность системы к дальнейшему развитию. Наиболее целесообразным будет вариант выбора такой архитектуры системы, которая на данный момент не считается устаревшей, но в то же время способна к дальнейшему развитию.

Первоочерёдной задачей является первичное погружение в тему, поскольку лицевая реконструкция – вопрос нетривиальный и охватывающий множество областей технической науки (см. часть 2). Тогда важным фактором изучения темы является вопрос получения промежуточных данных (например, различных представлений внутри нейронной сети на разных слоях), поэтому проектируемая система также должна быть способна дать объяснения на тему того, почему был получен тот или иной ответ.

Таким образом, итогом проектирования должна стать гибкая система, выдающая адекватный результат и обладающая потенциалом для улучшения; в то же время существенным плюсом станет возможность «внутренней работы», то есть использования не только пользователями, но и авторами в целях анализа.

## Требования к разрабатываемой системе

На вход система должна принимать изображение в одном из классических форматов файла (*.jpg*, *.png*). Внутренний алгоритм, по которому осуществляется лицевая реконструкция, должен быть способен работать со сложными условиями на фотографии (например, нестандартное освещение) и стремиться к тому, чтобы выдавать один и тот же результат для самых разных фотографий одного человека.

По итогам обработки изображения пользователь должен иметь возможность просмотра результата, чтобы удостовериться в его адекватности. Просмотр трёхмерной модели должен быть удобен для пользователя, обязательно наличие базовых функций в виде возможности приблизить/отдалить, повернуть или же переместить модель на окне с демонстрацией.

Система должна предоставлять настройки по визуализации, чтобы при желании пользователь мог изучить все аспекты результата. Обязательно наличие возможности просмотра результата реконструкции только в виде набора вершин, без закрашивания.

Программа должна сохранять результат в одном из общедоступных форматов 3D-моделей, поскольку это также способствует изучению системы, и в то же время позволяет применить результат в других средах (трёхмерные редакторы, компьютерные игры и др.).

# ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ

## Подходы к трёхмерной реконструкции лица человека

В задаче лицевой реконструкции доминируют два основных подхода: двумерный и трёхмерный. Первый подход является гораздо менее точным [54] в силу больших потерь информации при переходе из пространства в плоскость с помощью перспективной проекции (то есть, при фотографировании), поэтому предпочтительным является работа в трёхмерном пространстве. Несмотря на наличие у второго подхода существенного недостатка в виде высокой стоимости получения данных и построения систем мы считаем, что именно трёхмерные оценки являются наиболее перспективной методологией [54], в связи с чем рассмотрение некоторых методов, которые включают в себя работу исключительно с изображением и не делают никаких попыток совершить переход к третьему измерению, мы оставляем за рамками данной работы.

Поскольку входными данными у нас выступают изображения, подходы к реконструкции тесно связаны с более широкой областью – (цифровой) обработкой изображений. Поэтому мы можем выделить три больших группы методов реконструкции:

1. Аналитические методы;
2. Статистические методы;
3. Нейросетевые методы.

## Аналитические методы реконструкции

Возможность строгой аналитической постановки задачи реконструкции лица, вообще говоря, не доказана. Каким образом можно исчерпывающе формализовать задачу, если одно и то же лицо может находиться в совершенно разных условиях? Среди таких условий:

* Освещение (число источников, интенсивность излучаемого света и т. д.);
* Ракурс полученной фотографии;
* Параметры камеры (угол обзора, фокальное расстояние и т. д.);
* Перекрытие части лица посторонними предметами (это могут быть, например, волосы).

Перечисленные факторы относятся в первую очередь к объёму поверхности лица, но если задача подразумевает также цветовую реконструкцию, то возникнут и другие проблемы (например, влияние освещения на цвет участка кожи в реконструированной модели).

Описанные проблемы создают – при использовании аналитических моделей – критическую необходимость в инвариантных преобразованиях с целью сглаживания влияний внешних условий.

По части обработки изображений важным понятием является группа методов под названием *Shape from Shading* (*SfS*). Дословно *SfS* означает «объём из затенения», то есть третье измерение объекта восстанавливается с помощью данных об освещённости участков изображения. Впервые данное понятие вводит Бертольд Хорн [1], проводящий основные исследования данной области в 70-х и 80-х годах прошлого века. Тогда данная проблема была поднята для современных вопросов: появление электронных микроскопов, исследование поверхности Луны и прочие области. Для решения проблемы *SfS* автор приводит систему дифференциальных уравнений, где рассматриваются частные производные освещённости по области, охваченной изображением; данные уравнения впоследствии стали называться уравнениями освещённости (*image irradiance equation* или *image illumination equation*).

Однако со временем появились доказательства [2, 3] того, что методика *SfS* не является универсальным способом получения ответа. Вопросы существования корней и биективности отображения не были до конца изучены теоретически, в результате чего выяснилось, что получить один и тот же результат можно с помощью различных исходных данных в ситуации, когда объекты намеренно разные. Таким образом, *SfS* является некорректно поставленной задачей (*ill-posed problem*), и в первую очередь это связано с недостатком информации, которую можно получить только от одного изображения.

Имеется также и другая сложность – разнообразие типов источников и моделей освещения. Приведём здесь самые популярные типы источников:

* Фоновое освещение (*ambient light*), то есть источника света как такового нет, вместо этого существует константа для прибавления к имеющемуся значению освещённости или же для умножения на неё;
* Точечный источник света (*point light)*, интенсивность потока которого убывает пропорционально квадрату расстояния до объекта;
* Удалённый источник освещения, лучи от которого считаются параллельными (*directional light*) и незатухающими;
* Источник светового потока конической формы (*spotlight*).

И модели освещения:

* Модель освещения Ламберта

Данная модель восходит к физическому закону косинусов:

(1)

То есть, по данной модели имеют значение только направление света и форма поверхности, при этом положение камеры не рассматривается. Эту модель также называют моделью диффузного (рассеянного) освещения, поскольку предполагается, что рассеивание происходит равномерно во всех направлениях.

* Модель освещения Фонга (Блинна-Фонга)

Данная модель имеет более сложный вид:

(2)

То есть, помимо рассеянного освещения здесь также принимается в расчёт точка обзора. Вторая составляющая также называется зеркальным освещением.

И так, вот краткий список основных вариантов для моделирования света. Для моделей освещения были приведены «прямые» уравнения, в то время как предлагается решить «обратную» задачу, то есть получить направления нормалей из известных значений освещённости. Отсюда и вытекают уравнения с частными производными.

Во всём этом многообразии вариантов для моделирования тяжело сориентироваться. Для каждого типа источника разработан свой математический аппарат, для каждой модели освещения имеется свой набор уравнения освещённости точки, в связи с чем возникает вопрос о том, какой из вариантов следует использовать или же по какому алгоритму следует чередовать эти варианты?

Большинство исследований предполагает работу только с рассеянным освещением. В приложении А1 приведён пример вклада каждой составляющей в итоговый вид объекта на трёхмерной сцене. С одной стороны, человеческие лица не являются глянцевыми поверхностями и какими-либо бликами действительно можно пренебречь, а с другой – имеем полностью матовую поверхность, некоторые участки которой можно ошибочно счесть плоскими. На этом изображении в нижнем ряду изображена тороидальная поверхность, на примере которой очень хорошо видно, что при использовании только рассеянной составляющей освещения обращённая к нам сторона кажется скорее множеством квадратов со скруглёнными углами, нежели множеством кругов, поскольку самые освещённые участки визуально кажутся прямыми линиями. Подобного рода сложности в сумме приводят к тому, что технология *SfS* оказывается несостоятельным методом реконструкции лица, поскольку здесь важная каждая мельчайшая деталь.

Даже те исследования, которые используют более продвинутые методы аппроксимации поверхности (например, [4]), страдают от недостатка информации. Авторы исследования предлагают вместо оценки «с нуля» использовать базовую модель лица, а данные с изображения используются для деформации базовой модели с целью получения модели реконструированной. Световые параметры оцениваются с помощью сферических гармоник – ортонормированного базиса сферических функций, с помощью которого через линейную комбинацию этих базисных функций можно описывать освещённость поверхности. Но даже несмотря на такие усложнения модель всё ещё оказывается неспособна выдать достойный результат в некоторых ситуациях: в приложении А2 находятся примеры получаемого результата, где можно видеть, что улыбка человека или надувание щёк приводит к совершенно нечеловеческой форме реконструированного лица.

Таким образом, поставить и решить задачу с использованием аналитических методов (или, по крайней мере, с использованием только их) оказывается крайне сложно; до сих пор неизвестно, возможно ли использовать данный подход для получения визуально точной реконструкции.

## Статистические методы реконструкции

В противовес аналитическому подходу существует также сугубо статистический – попытка уйти от сложностей с постановкой через выявление статистических закономерностей.

Публикация [4], приведённая выше, использует главную идею статистического подхода – наличие базовой формы лица, для которой следует задавать деформации. Группа статистических методов восходит к статье [5], где впервые была предложена концепция реконструкции на основе статистики. Основная идея состоит в получении усреднённой лицевой модели, которую можно использовать как базис, тогда как все вариации – индивидуальные черты, выражения лица или же условия ракурса и освещения – описываются через смещения в этом базисе. Для формирования базиса используется заданное число 3D-моделей, линейная комбинация которых позволяет – по предположению авторов – получить любое необходимое лицо; тогда каждое лицо характеризуется уникальным вектором коэффициентов разложения по элементам базиса. Таких векторов два: один используется для описания формы лица, второй – для текстуры (цвета).

Размерность такого пространства равняется числу вершин 3D-модели лица, то есть нескольким десяткам тысяч измерений, поэтому используется сжатие данных по методу главных компонентов или *PCA*. *PCA* [40, 41] – это способ представления данных с помощью пространства, имеющего размерность меньше исходного, для которого в качестве критерия используется потеря информации. Новая координатная система определяется найденными собственными значениями матрицы ковариаций входных данных, а её оси упорядочены по убыванию соответствующих собственных значений. Преобразование по МГК нацелено на получение максимальной дисперсии вдоль осей новой координатной системы, причём идеальным считается тот случай, когда суммы дисперсий по старым и по новым осям совпадают (потеря информации отсутствует). Итогом такого преобразования становится новое представление данных меньшего размера, в которых содержится информация об объекте в другом виде; процесс получения нового представления обычно называется «извлечением признаков» (*feature extraction*).

Задав границу потери информации каким-то числом (1%, 5%, 10% или любой другой показатель), можно уменьшить количество базисных векторов в несколько раз, что существенно сократит объём вычислений, не оказывая при этом критического влияния на качество результата. В приложении А3 приведена схема моделирующей системы от авторов.

Статистический метод получил очень большое внимание исследователей, о чём свидетельствует число цитирований публикации [5]. Идея действительно оказалась очень привлекательной, поскольку на базе такой методики можно было не только реконструировать лица с изображений, но и создавать новые; как результат, применение метода было очень разнообразным, а полученные модели лица широко используются в исследованиях до сих пор.

Направлений развития данного подхода имеется множество. Авторы исследования [11] адаптируют алгоритм под системы распознавания, в исследовании [12] используется многомерное нормальное распределение для оценки глубины, множество различных вариаций также можно найти в анализе публикаций из [7] (пункты 1 и 2), а также в пункте 2.1 публикации [8].

Статистические модели имеют значительные преимущества в сравнении с аналитическими:

1. Отсутствует необходимость в строгой аналитической постановке, поскольку в оценках и преобразованиях задействованы только статистические методы;
2. Отсутствуют ограничения на размерность пространства (число базовых моделей), что позволяет регулировать сложность работы алгоритма для получения индивидуального компромисса между производительностью и точностью;
3. Состав базовых моделей также позволяет регулировать работу алгоритма с точки зрения репрезентации различных групп людей (пол, цвет кожи и др.).

Главная проблема статистического метода – зависимость от самой статистики. Качество прогнозирования зависит от того, какие образцы будут включены в множество, из которого формируется базис. Соответственно, требуется репрезентация всех человеческих подгрупп и по возрасту, и по цвету кожи, и по всем остальным чертам. Вдобавок, модель всё ещё линейна, а допущение о линейности в условиях такого разнообразия лиц может оказаться критическим.

В статье [5] задача была упрощена до базиса из 200 лиц схожей этнической принадлежности и возраста. Обучение статистической модели происходило с помощью сопоставленных результатов 3D-сканирования и фотографий, причём сканирование проводилось в лабораторных условиях, что означает полный контроль над освещением, положением человека и выражением его лица. В результате у созданной модели имеются проблемы с обобщением на большинство людей. Согласно анализа публикаций из [7], многочисленные попытки улучшений модели так и не привели к качественному улучшению модели: всё ещё использовались «дорогие» данные, полученные при трёхмерном сканировании, всё ещё оставалась проблема недостаточной размерности базиса и всё ещё существовал барьер, возникший из-за допущения о линейности. Там же даны рассуждения о том, что использование нормального распределения, которое подразумевается в классических статистических моделях, не способно дать качественный результат.

В результате можно сказать, что применение статистических методов в поставленной мной задаче не представляется осмысленным в силу невозможности получения результата исключительно из изображений, даже если они все будут сделаны в лабораторных условиях.

## Нейросетевые методы реконструкции

В настоящий момент в задаче реконструкции лиц доминируют искусственные нейронные сети (ИНС). При описании статистического подхода мы отметили, что методика получения реконструированного лица сводится к линейной комбинации базисных элементов; в то же время качество извлечённых знаний при опоре на мат. статистику критически зависит от используемой выборки. Здесь мы можем обозначить две основные проблемы – линейный характер преобразований и обобщающая способность.

С этой точки зрения нейронная сеть является вполне закономерным развитием идей, поскольку она вполне естественным образом включает в себя нелинейные преобразования и имеет теоретически неограниченный потенциал к обобщающей способности. ИНС позволяют получить неявные представления данных различной сложности, тем самым проектировщики систем элегантно уходят от сложностей постановки задачи.

Каждый слой классической полносвязной нейронной сети (однослойный персептрон, рис. 2) осуществляет преобразование пространства по следующему правилу [42, 45, 46]:

(3)

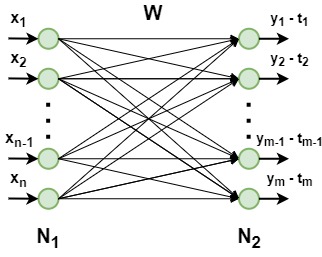


Рисунок 2 – структура полносвязного слоя

Отметим, что с точки зрения математических операций метод главных компонент – это поиск матрицы перехода от одного базиса к другому и умножение на неё всех входных элементов. Можем сказать, что данное преобразование эквивалентно однослойному персептрону без активационной функции. Соответственно, при обнаружении пределов возможностей применения МГК исследователи стали переходить к нейронным сетям с несколькими слоями и различными активационными функциями. Осуществляя многократное преобразование пространства, нейронные сети позволяют получать сложные неявные представления [49], что может приводить к ещё большему коэффициенту сжатия данных при отсутствии существенных потерь информации [46]. Таким образом, применение нейронных сетей позволяет избежать линейности в постановке задачи.

Важным этапом развития нейросетевой обработки изображений в общем, и работы с изображениями лиц – в частности, является переход к повсеместному использованию свёрточных нейронных сетей. СНС по своей задумке являются развитием многослойного персептрона, но при этом опирается на принцип локальной связности входных данных [46, 47].

Во-первых, свёрточный слой отличается от полносвязного отсутствием полной матрицы весов. Здесь вместо связи «каждый с каждым» используется [43] окно (или «ядро» (*kernel*)) небольшого размера, и это окно используется для преобразования изображения небольшими участками, а не сразу целиком.

Во-вторых, свёрточный слой разбивается на «фильтры» - двумерные наборы данных одинакового размера. Действия, совершаемые свёрточной нейронной сетью с изображением, по сути являются последовательным наложением на изображение различных фильтров, известных в области цифровой обработки изображений (фильтры Габора, Собеля и др.), поэтому данный термин здесь неслучайно.

В-третьих, в рамках одного слоя для всех фильтров используется один набор весов. Это позволяет в сочетании с первым пунктом существенно сократить количество весов и применять на каждом слое множество фильтров, тем самым потенциально расширяя количество различных признаков, которые можно извлечь из изображения.

В-четвёртых, помимо операции свёртки используется операция подвыборки. «Подвыборка» заключается в отсеивании наименее активных участков фильтров; это позволяет снизить размерность данных и не привлекать к дальнейшим свёрткам те значения, на которых предыдущие фильтры ничего не дали (грубо говоря, не обнаружили). Обычно используется подвыборка по принципу выбора наибольшего значения, реже – среднего значения, могут встретиться и другие виды.

Эти и другие факторы привели исследователей к положению о большем потенциале свёрточных сетей в обработке изображений. Последнее десятилетие можно назвать эпохой свёрточных сетей: примерно с 2012 года подавляющее большинство публикаций, затрагивающих тему обработки изображений, имеет в своём тексте слово «свёрточная нейронная сеть» в различных формах в контексте используемой архитектуры; среди таких публикаций – примеры из глав 2.1, 2.2, 2.4 и 2.6 в публикации [8]. Приведённые примеры относятся непосредственно к задаче реконструкции, однако использование свёрточных архитектур достаточно широко в любой задаче, связанной с изображениями, о чём свидетельствуют четырёх- и пятизначные количества цитирований публикаций, в которых были представлены популярные архитектуры ИНС, например, *ImageNet* [21], *VGG-Face* [22], *Inception* [23-25].

Главная идея в большинстве исследований остаётся неизменной – имеем базовую модель и оцениваем поточечные цвет и объём (деформацию). Примерами использования подхода с базовой моделью и статистическими величинами являются [7-10]. В такого рода системы, как правило, входят следующие компоненты:

* Собственно, базовая модель лица с известными заранее параметрами;
* Нейронная сеть, извлекающая признаки из исходного изображения и дающая на выходе сжатое представление;
* Другая нейронная сеть, формирующая текстуру лица из полученного представления;
* Третья нейронная сеть, оценивающая объём поверхности лица из полученного представления, либо же объём оценивается непосредственно через линейную комбинацию базисных элементов статистической модели;
* Подпрограмма, синтезирующая модель и изображение;
* Сложная функция потерь, составленная по принципу *multi-task learning* [13], то есть состоящая из нескольких слагаемых со взвешивающими коэффициентами.

При этом даже в рамках приведённых примеров есть существенные различия, например, в [10] нейронная сеть используется только оценки текстуры, а оценка объёма остаётся «на плечах» метода главных компонент. Структура системы реконструкции из [7] приведена в приложении А4.

Таким образом, предполагается, что ИНС использует для обучения результаты собственной работы. Здесь могут быть полезными генеративно-состязательные [14] или автоэнкодерные [48] сети.

## Итоги анализа

По итогам проведённого исследования можем составить общую характеристику трёх вариантов. Сделаем это в формате *SWOT*-анализа [51], дав оценку каждой группе методов по четырём пунктам: сильные стороны (*Strengths*), слабые стороны (*Weaknesses*), возможности (*Opportunities*), угрозы (*Threats*):

1. Группа аналитических методов

* (*S*) В качестве сильной стороны аналитического подхода можно выделить разве что предсказуемость. Производимые в рамках *SfS* действия абсолютно понятны с точек зрения и строго математической, и чисто человеческой;
* (*W*) Предсказуемость – как следствие строгой постановки задачи – в нашем случае неизбежно влечёт за собой упрощение, поскольку мы задаём для алгоритма крайне строгие условия работы. Как уже говорилось, человек не всегда может исчерпывающе формализовать задачу;
* (*O*) Количество и значимость факторов, не позволяющие применить эти методы в качестве основы для реконструкции, не означают полную неспособность получить полезную информацию. Аналитические методы могут оказаться полезными в качестве вспомогательных и использоваться наравне с другими методами обработки изображений;
* (*T*) По сути дела проблемой является любой резкий переход на участке изображения. Таким резким переходом может стать не только перепад в освещении, но и край лица или родимое пятно;

1. Группа статистических методов

* (*S*) Главные преимущества использования метода главных компонент – это самостоятельный поиск оптимального решения алгоритмом вместо строгих условий, а также гибкость в выборе между потерей информации и вычислительной сложностью;
* (*W*) Основным ограничением является допущение о линейности, что для сложной задачи (лицевая реконструкция может быть отнесена к таковым по праву) может оказаться губительным;
* (*O*) Системы на основе МГК способны быстро перестраиваться при обновлении набора данных, что позволяет проводить множество экспериментов и выявить оптимальную конфигурацию;
* (*T*) Серьёзным препятствием может оказаться нехватка данных, поскольку качественные трёхмерные данные создать тяжело. Также имеет место проблема перекоса выборки, то есть отсутствия репрезентации некоторой человеческой группы;

1. Группа нейросетевых методов

* (*S*) Нейронные сети имеют огромный потенциал, и главным их преимуществом можно назвать представление данных: оно может быть нелинейным и оно может быть представлено огромным множеством способов в зависимости от архитектуры внутренних слоёв;
* (*W*) Результаты работы ИНС сложно, а иногда и вовсе невозможно интерпретировать, то есть система оказывается непрозрачной и непонятной. Также можно отметить сложности с обучением больших систем (и по времени, и по качеству);
* (*O*) Серьёзно помогает то обстоятельство, что сейчас внимание исследователей сконцентрировано по большей части именно на нейронных сетях. Это позволит подвергать структуру системы небольшим изменениям, проводить эксперименты и таким образом искать путь к повышению качества работы;
* (*T*) Проблемы, связанные с выборкой, никуда не деваются: нейросети – суть статистика, и нужно тщательно отбирать данные для обучения;

Рассмотрев основные вопросы, связанные с подходами к решению поставленной задачи, можно сделать следующие выводы:

* Применение *SfS* и других вспомогательных методик в рамках попыток аналитического решения задачи не представляется целесообразным

При дальнейшей работе (выходящей за рамки ВКРБ) эти технологии анализа изображений можно будет применить в качестве вспомогательных на этапе подготовки данных, поскольку невозможно быть уверенным в том, что дополнительная предобработка данных не пойдёт модели на пользу. Как правило, оказывается ровно наоборот, и качество моделей машинного обучения, включая ИНС, очень и очень сильно зависят от «качества» данных. На входы сети можно подавать не только само изображение, но и результаты его обработки различными методами, например, фильтры Габора или Собеля, а также гистограмма направленных градиентов [52].

* Проектируемая система будет построена на нейронных сетях в рамках расширения статистического подхода

Использование других парадигм – гораздо более сложная задача за счёт меньшего количества информации. Вдобавок, совершенно не гарантировано, что эти парадигмы смогут дать решение на порядок качественнее, поэтому на текущий момент останавливаемся на самой популярной подгруппе методов.

* Проектируемая система должна иметь возможность обучаться как на обыкновенных изображениях, так и на вспомогательных трёхмерных данных

Мы нацелены на построение системы, обучающейся на изображениях, поскольку их гораздо больше, нежели данных лазерного сканирования, но в то же время последняя группа – это самые точные и высококачественные данные, поэтому пренебрегать ими нельзя.

* Наиболее целесообразная архитектура ИНС – свёрточные сети

На данный момент нет причин для использования других архитектур. Построение предварительного макета на СНС позволит получить какой-никакой базовый результат, и в то же время я в рамках выполнения работы смогу гораздо лучше познакомиться с темой нейронных сетей и различных способов их построения и обучения.

# КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ

Данный раздел посвящён проектируемой системе.

В главе 3.1 приведены общие архитектурные принципы, сформированные в рамках исследовательской части и анализа публикаций, затрагивающие рассматриваемую область или смежные с ней.

Глава 3.2 описывает набор данных, использование которого необходимо в связи с тем, что система опирается на методы глубокого обучения.

Далее следует глава 3.3, где общие принципы раскрываются; даётся полное описание алгоритма реконструкции лица со схемами и выкладками в виде формул.

Глава 3.4 представляет описание сложной функции потерь, которая использовалась для обучения нейронной сети; приведено описание каждого слагаемого с обоснованием необходимости.

В рамках главы 3.5 делается выбор программных фрэймворков и библиотек, которые будут использоваться для достижения цели и удовлетворения всех требований к системе.

Глава 3.6 посвящена созданию интерфейса системы, который реализован в виде графического приложения; описывается и демонстрируется логика работы интерфейсных элементов, приводится граф диалога пользователя.

В главе 3.7 проводится анализ результатов работы спроектированной системы: оценивается общее качество результатов реконструкции, исследуется устойчивость получаемого результата по отношении к преобразованиям входных данных (сдвиг, вращение, масштабирование), определяется способность системы сохранять уникальность черт лица, а также выдвигаются предположения по итогам каждого из пунктов анализа.

В главе 3.8 подводятся итоги и делается заключение о состоянии системы и возможности дальнейшей модернизации.

## Архитектура проектируемой системы

Для создания собственной системы реконструкции будем в большей степени опираться на публикацию [7]. Центром системы являются три нейронных сети:

* Первая сеть – кодировщик: принимает на вход изображения и выдаёт сжатое представление (кодовый вектор);
* Вторая сеть – расшифровщик формы: принимает на вход соответствующую часть сжатого представления и выдаёт текстуру, в которой зашифрован объём (смещения вершин);
* Третья сеть – расшифровщик цвета: принимает на вход соответствующую часть сжатого представления и выдаёт текстуру, в которой зашифрован цвет (но ещё не готовая текстура).

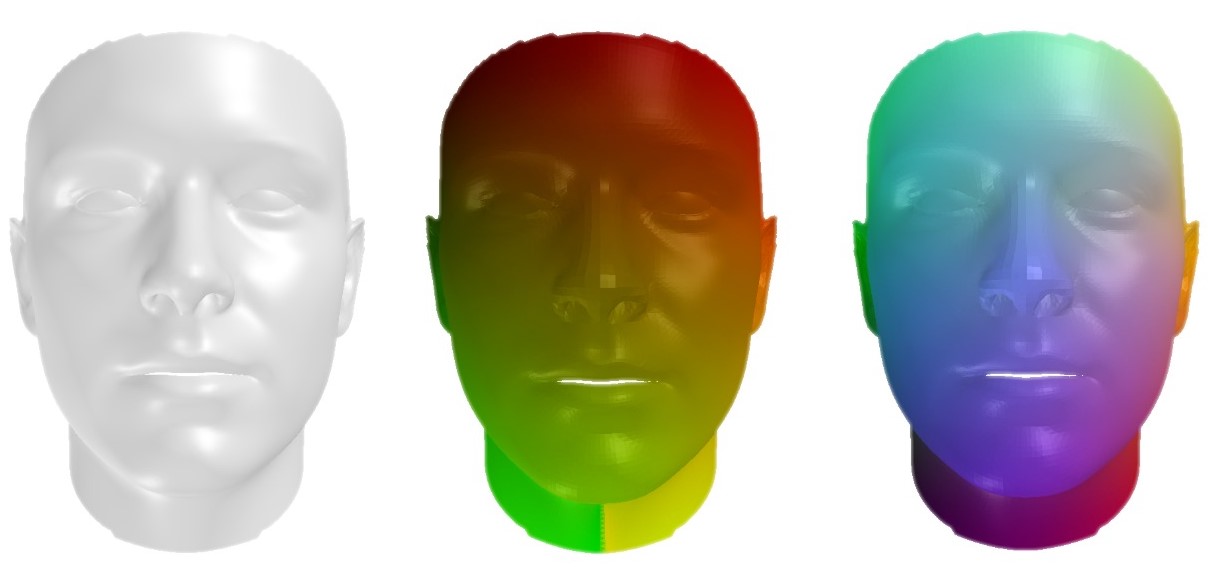
Размерности сжатого представления можно выбрать любыми, но авторы проводили эксперименты с числом 160 (и для формы, и для цвета), поэтому оставим такое же значение.

В сжатое представление также входят данные о проекции и об освещении. Проекция представляет собой 6 чисел, которые составляют два вектора; третий вектор получается путём векторного произведения, после чего из трёх векторов составляется матрица вращения. Эту матрицу вращения мы будем использовать для трёхмерной модели, чтобы корректно расположить объект по отношению к камере. Коэффициенты освещения – это коэффициенты разложения по сферическим гармоникам [15-17] – технике моделирования глобального освещения, работающей в трёхмерном сферическом пространстве аналогично разложению Фурье. Примечательно то, что сферические гармоники составляют ортонормированный базис, поэтому мы можем выразить освещение через линейную комбинацию базисных функций; ИНС выдаёт девять коэффициентов, аппроксимируя освещение гармониками до 2 степени.

Форма и цвет, то есть выходы расшифровывающих нейросетей выражены в текстурном формате, что является естественной формой, поскольку расшифровщики – свёрточные. При необходимости текстуры переводятся в вершинный формат с помощью стандартного UV-преобразования [29]. В приложении А5 приведена обобщённая блок-схема создаваемой системы.

Устроенная таким образом архитектура полностью соответствует предъявляемым требованиям и задачам разработки, поскольку мы имеем множество этапов, на которых появляются промежуточные данные; эти данные позволят лучше понять то, как программа «думает».

Все манипуляции будут производиться на базовой модели лица, изображённой на рисунке 3. Текстурные координаты, представленные по центру, позволяют осуществить упомянутое выше UV-преобразование: текстурная координата показывает, с какого места в текстуре будет взят цвет для данной модельной вершины. Например, красная зона около лба означает, что значение Y близко к нулю, то есть значения будут браться из верха текстуры.



Слева: форма лица

По центру: визуализация текстурных координат на форме (красный цвет – ось X, зелёный цвет – ось Y)

Справа: визуализация модельных координат на форме (красный цвет – ось X, зелёный цвет – ось Y, синий цвет – ось Z)

Рисунок 3 – Базовая модель лица

## Набор данных для обучения и проверки

Поскольку мы остановились на системе с использованием ИНС, потребуется набор данных для обучения. Вообще для различных задач обработки человеческих лиц на изображениях (определение ракурса, определение ключевых точек, распознавание как таковое и др.) есть огромное множество различных наборов данных [18, 19]. В ситуации, когда в качестве входных данных используются только изображения, нам нужно только проконтролировать, что в выбранном нами варианте имеется достаточное разнообразие в лицах, чтобы нейронная сеть имела хоть какую-то обобщающую способность.

Авторы работы [7] на основе наработок из [20] подготовили для нескольких наборов данных с лицами также соответствующие данные о текстуре, вершинах и проекции для упрощения процесса обучения (см. об этом в пункте 3.4): ничего не мешает обучать нейронную сеть исключительно на изображениях, но этот процесс может оказаться чересчур долгим и нестабильным, поэтому данные наработки будут использоваться в качестве материала для первичного обучения нейронной сети и мной. Более подробно вопросы обучения пояснены в пунктах 3.3 и 3.4

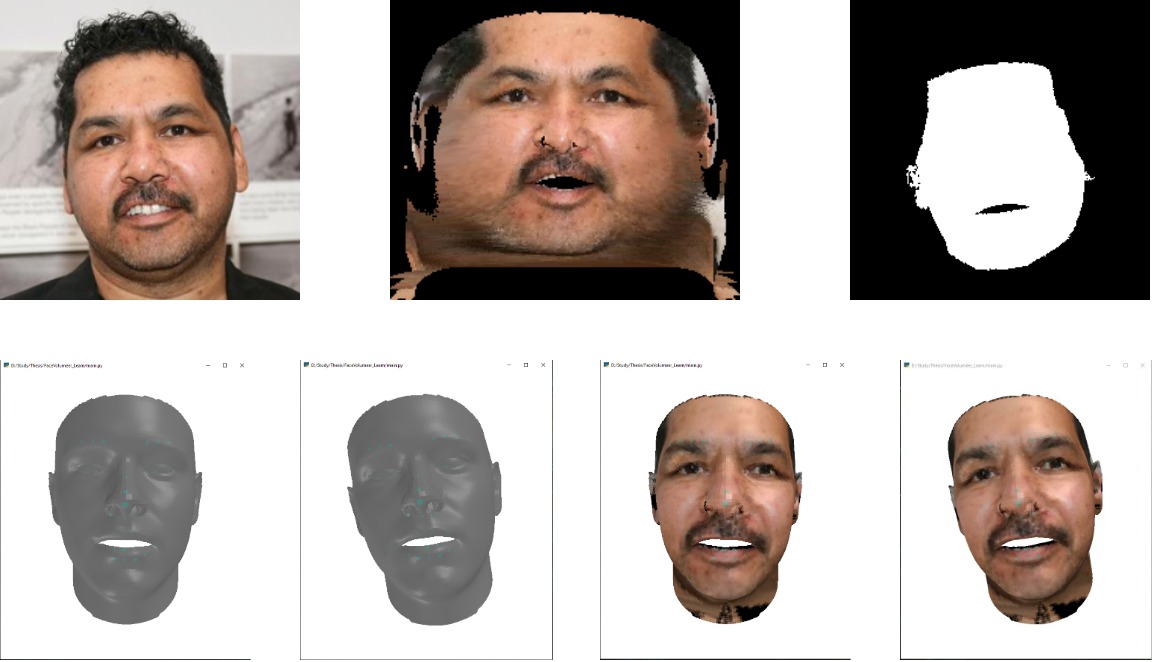
В исходных материалах работы [7] представлено огромное количество изображений (в сумме – около 124 тысяч). Данные являются аугментированными, то есть итоговое число в 124 тысячи составляют не только уникальные образцы, но и модифицированные версии одного и того же. Во-первых, каждое изображение было пропущено через предобученную нейронную сеть, осуществляющую преобразование ракурса на фотографии. Это позволяет получить несколько представлений одного лица на фотографии, тем самым нейронная сеть должна обучить инвариантным преобразованиям для достижения (в идеальном случае) идентичного результата при различных положениях камеры. Во-вторых, каждое из полученных таким образом изображений дополняется двойником, полученным зеркальным отражением по горизонтали. В результате представленные данные предлагают широкий диапазон расположений лиц (от -90 до +90 градусов к камере по горизонтали и по вертикали); каждое изображение дополняется до набора из двух-трёх десятков образцов.

Обеспечение инвариантности по смещению лица на изображении авторы [7] предлагают достичь следующим образом: исходное изображение имеет размер 256 пикселей в обоих измерениях, а в нейронную сеть будет подаваться экземпляр размером 224 (также в обоих измерениях), полученный обрезанием случайного участка по краям. В результате мы будем иметь потенциально бесконечный набор данных, поскольку каждую эпоху входные данные будут слегка отличаться.

Столь масштабное расширение обучающей выборки имеет два побочных эффекта:

* Сложность статистики входных данных, обеспечиваемая высокой дисперсией, практически исключает возможность переобучения, что является безусловным плюсом;
* Количество входных данных настолько огромно, что на полное обучение модели может понадобиться несколько месяцев при условии отсутствия кластера из *GPU*, что является огромной проблемой, поскольку есть риск не успеть завершить работу;

В приложении А6 продемонстрировано два примера аугментации исходных изображений. На рисунке 4 представлены входные данные для одного образца и их визуализация:



Верхняя строка: исходное изображение, заготовленная текстура и маска;

Нижняя строка: визуализация по заготовленным данным, слева направо: данные о форме; данные о форме и проекции; данные о форме и текстура; данные о форме и проекции, а также текстура

Рисунок 4 – Комплект входных данных

## Описание алгоритма трёхмерной реконструкции

В пункте 3.1 была представлена общая архитектура системы. Здесь будет представлен более подробный алгоритм с рассмотрением всех шагов.

Изображения подаются в нейронные сети. Большой размер набора данных не позволяет использовать пакетный режим обучения из-за технических ограничений компьютера; использование стохастического градиентного спуска не видится хорошей идеей в силу большого различия обучающих образцов, то есть имеется риск получения слишком высоких колебаний функции потерь; мини-пакетный режим остаётся единственным вариантом.

В связи с высоким размером входных данных также применяется технология пакетной нормализации [26] (*batch normalization*). При использовании *BN* во время обучения данные в мини-пакете нормализуются, то есть приводятся к стандартному нормальному распределению путём вычитания выборочного среднего и деления на выборочное стандартное отклонение. Помимо осуществления преобразования каждый такой слой полезен наличием двух параметров: сдвига и масштабирования. Эти два параметра применяются после нормировки и являются обучаемыми (то есть, подстраиваются вместе с весами). Таким образом, на этапе прогнозирования каждое изображение будет преобразовываться в соответствии со статистикой, накопленной в процессе обучения нейронных сетей. Польза *BN* также обосновывается тем [26, 44], что образцы, которые имеют одинаковый или схожий результат, могут иметь очень далёкое друг от друга представление на исходных изображениях, что сильно усложняет задачу работы классификатора или любой другой системы.

Экспериментально была определена возможность компьютера работы с мини-пакетом размером до 8, именно такой размер и был настроен для обучения. Используемый алгоритм обновления весов – *Adam* [27] в модификации *AMSGrad* [53] с параметрами: шаг обучения (*learning rate*) – 0,0005, первый момент (*beta 1*) – 0,5, второй момент (*beta 2*) – 0,999, константа во избежание деления на ноль (*epsilon*) – 1,0.

Как уже было сказано, все три сети – свёрточные. Структуры ИНС представлены в таблицах 1-2, а также в приложениях А7 и А8 в виде схем.

Таблица 1 – Архитектура кодировщика

| **#** | **Размерности входных данных (без учёта размера пакета обучения)** | **Параметры ядра свёртки/ подвыборки (С/П)** | **Размерности выходных данных (без учёта размера пакета обучения)** | **Дополнительные подслои** |
| --- | --- | --- | --- | --- |
| 1 | 224х224х3 | Размер ядра: 7х7  Шаг: 2  (С) | 112х112х32 | Пакетная нормализация  Активационная функция *ReLU* [28] |
| 2 | 112х112х32 | Размер ядра: 5х5  Шаг: 1  (С) | 112х112х96 | Пакетная нормализация  Активационная функция *ReLU* |
| 3 | 112х112х96 | Размер ядра: 5х5  Шаг: 2  (С) | 56x56x64 | Пакетная нормализация  Активационная функция *ReLU* |
| 4 | 56x56x64 | Размер ядра: 3х3  Шаг: 1  (С) | 56x56x128 | Пакетная нормализация  Активационная функция *ReLU* |
| 5 | 56x56x128 | Размер ядра: 3х3  Шаг: 2  (С) | 28x28x192 | Пакетная нормализация  Активационная функция *ReLU* |
| 6 | 28x28x192 | Размер ядра: 3х3  Шаг: 2  (С) | 14x14x256 | Пакетная нормализация  Активационная функция *ReLU* |
| 7 | 14x14x256 | Размер ядра: 3х3  Шаг: 2  (С) | 7x7x256 | Пакетная нормализация  Активационная функция *ReLU* |
| 8.1\* | 7x7x256 | Размер ядра: 3х3  Шаг: 1  (С) | 7x7x160 | Активационная функция *tanh* [28] |
| 8.2\* | 7x7x256 | Размер ядра: 3х3  Шаг: 1  (С) | 7x7x160 | Активационная функция *tanh* |
| 8.3\*, 8.4\* | 7x7x256 | Размер ядра: 3х3  Шаг: 1  (С) | 7x7x64 | Отсутствуют |
| 9.1 | 7x7x160 | Размер ядра: 7х7  Шаг: 1  (П) | 160 | Отсутствуют |
| 9.2 | 7x7x160 | Размер ядра: 7х7  Шаг: 1  (П) | 160 | Отсутствуют |
| 9.3, 9.4 | 7x7x64 | Размер ядра: 7х7  Шаг: 1  (П) | 64 | Отсутствуют |
| 10.3 | 64 | Отсутствуют (полносвязный слой) | 8 | Активационная функция *tanh* |
| 10.4 | 64 | Отсутствуют (полносвязный слой) | 9 | Активационная функция *tanh* |

\* – кодировщик имеет разветвлённую архитектуру. Второй индекс слоя означает следующую принадлежность подгруппы: 1 – данные о форме, 2 – данные о цвете, 3 – данные о проекции, 4 – данные об освещении; первый индекс слоя точно так же определяет последовательность, то есть 9.1 следует после 8.1, но перед 10.1, и так далее

Таблица 2 – Архитектура расшифровщика (полностью идентичны для расшифровщиков формы и цвета)

| **#** | **Размерности входных данных (без учёта размера пакета обучения)** | **Параметры ядра свёртки** | **Размерности выходных данных (без учёта размера пакета обучения)** | **Дополнительные подслои** |
| --- | --- | --- | --- | --- |
| 1 | 160 | Отсутствуют (полносвязный слой) | 13440 | Активационная функция *ReLU* |
| 2 | 13440 | Отсутствуют (слой перегруппировки) | 6х7х320 | Отсутствуют |
| 3 | 6х7х320 | Размер ядра: 3х3  Шаг: 2 | 12х14х160 | Пакетная нормализация  Активационная функция *ReLU* |
| 4 | 12х14х160 | Размер ядра: 3х3  Шаг: 1 | 12х14х256 | Пакетная нормализация  Активационная функция *ReLU* |
| 5 | 12х14х256 | Размер ядра: 3х3  Шаг: 2 | 24х28х256 | Пакетная нормализация  Активационная функция *ReLU* |
| 6 | 24х28х256 | Размер ядра: 3х3  Шаг: 1 | 24х28х128 | Пакетная нормализация  Активационная функция *ReLU* |
| 7 | 24х28х128 | Размер ядра: 3х3  Шаг: 2 | 48х56х128 | Пакетная нормализация  Активационная функция *ReLU* |
| 8 | 48х56х128 | Размер ядра: 3х3  Шаг: 1 | 48х56х64 | Пакетная нормализация  Активационная функция *ReLU* |
| 9 | 48х56х64 | Размер ядра: 3х3  Шаг: 2 | 96х112х64 | Пакетная нормализация  Активационная функция *ReLU* |
| 10 | 96х112х64 | Размер ядра: 3х3  Шаг: 2 | 192х224х32 | Пакетная нормализация  Активационная функция *ReLU* |
| 11 | 192х224х32 | Размер ядра: 3х3  Шаг: 1 | 192х224х12 | Пакетная нормализация  Активационная функция *ReLU* |
| 12 | 192х224х12 | Размер ядра: 3х3  Шаг: 1 | 192х224х3 | Активационная функция *tanh* |

Выходные данные системы ИНС используются для полностью аналитических (имеется в виду определённость, а не описанная во второй части группа методов) преобразований:

1. Полученная «текстура» объёма преобразуется в вершинный формат для получения деформированной модели

Здесь стоит отметить, что перед подачей в нейронные сети данные о форме нормируются (среднее значение и стандартное отклонения вычисляются при входе в программу на основе всех рабочих экземпляров), поэтому данный этап включает в себя два шага:

* Переход от текстурных координат смещения к вершинным. Осуществляется с помощью заготовленных для базовой модели *UV*-координат:

(4)

* Переход от смещения к деформированной модели:

(5)

1. Деформированная модель поворачивается с помощью составленной матрицы вращения

Данные о проекциях нормируются подобно форме, поэтому здесь имеем следующие шаги:

* Статистическое преобразование значений (наподобие формулы 5);
* Перевод выходных данных к паре векторов – простая группировка значений на две тройки;
* Вычисление третьего вектора и формирование матрицы вращения:

(6)

* Поворот трёхмерной модели:

(7)

1. По деформированной модели считаются нормали вершин

Для вычисления нормалей вершин требуется знать нормали треугольников, поэтому здесь имеем два действия:

* Вычисление нормалей треугольников. Каждая нормаль считается следующим образом:

(8)

* Переход от нормалей треугольников к нормалям вершин. Нормаль каждой вершины считается следующим образом:

(9)

1. Нормали треугольников преобразуются в текстурный формат с помощью обратного UV-преобразования, получается «карта нормалей»

Для осуществления такого преобразования используются заранее заготовленные данные о том, каким образом нормали переходят из вершинного пространства в текстурное. Эти данные представляют из себя карту, где каждому пикселю в соответствие поставлен треугольник, то есть являются неким подобием текстурных координат, представленных ранее на рис. 3. Нам уже известны нормали вершин, поэтому мы берём для каждого треугольника три нормали вершин и берём их среднее. Такое решение – в сравнении с возможностью просто взять те нормали, которые были посчитаны в рамках первого шага третьего этапа – позволит сделать карту нормалей гладкой текстурой, это обусловлено усреднением нормалей вершин по треугольникам (см. третий этап, второй шаг).

1. По полученной карте нормалей высчитывается освещение, получается «карта теней»

Нейронная сеть даёт нам девять коэффициентов разложения, а карту теней мы получаем следующим образом:

(10)

Освещённость и нормаль здесь описаны как вершинные условно, поскольку вся работа идёт в рамках двумерных текстур.

Значения сферической гармоники для случая действительных чисел считаются по известным формулам непосредственно в декартовых координатах, см., например, [16], стр. 22.

1. Из цветовой текстуры и карты теней формируется итоговая текстура

Цветовая текстура по замыслу несёт в себе информацию о «чистых цветах» лица человека, а сформированная карта теней содержит информацию об освещённости каждой точки, поэтому сформировать итоговую текстуру мы можем просто поэлементным умножением:

(11)

1. Текстура модели переводится в вершинный формат
2. Полученная модель – форма, ракурс, цвета – помещается на виртуальную трёхмерную сцену с камерой, где происходит рендеринг изображения

Подробнее про рендеринг изображения будет сказано в пункте 3.5

1. На основе синтезированного изображения оценивается функция потерь

Функции потерь подробно описаны в пункте 3.4.

В приложении А9 находится блок-схема, в которой описанный выше алгоритм представлен вместе с примером на одном образце.

## Функции потерь

Для соответствия ответов ИНС замыслам авторов и близости синтезированных данных к исходным в создаваемой системе будут применяться следующие функции потери:

1. Функция потерь по реконструированному изображению

(12)

Данная функция потерь является центральным слагаемым итоговой функции потерь, поскольку вся работа сводится именно к синтезированию изображений. Здесь происходит попиксельное сравнение исходных данных и того, что получилось после реконструкции, в качестве меры сравнения взято евклидово расстояние.

1. Функция потерь по реконструированной текстуре

(13)

Данная функция потерь направляет нейронные сети с точки зрения реконструкции итоговой текстуры, производя попиксельное сравнение, как и в предыдущем пункте, но с помощью абсолютной разницы, а не евклидового расстояния. Оказывает влияние на всю систему, поскольку итоговая текстура в соответствии с алгоритмом реконструкции складывается не только из данных от расшифровщика цвета, но также и от текстуры затенения, которая, в свою очередь, зависит от проекции и от направления нормалей, а нормали зависят от итоговой формы.

1. Функция потерь по реконструированной проекции

(14)

Потери по проекции заключаются в оценке расхождений между компонентами вектора ракурса, получаемого из кодировщика, и исходного вектора.

1. Функция потерь по реконструированной форме

(15)

Потери по форме также оцениваются по каждой компоненте, то есть вершине базовой модели. Мера – евклидового расстояние между исходной и реконструированной вершиной.

1. Функция потерь по симметрии цветовой текстуры

(16)

Данная функция потерь может быть отнесена к вспомогательным. Для случая повёрнутого лица (что встречается гораздо чаще, чем лицо, непосредственно смотрящее в камеру) алгоритм восстановления отсутствующей части лица не определён, поэтому введение данного слагаемого позволяет исключить неопределённость за счёт поощрения горизонтальной симметрии, то есть переноса значений с видимой части лица на невидимую.

1. Функция потерь по гладкости реконструированной формы

(17)

Ещё одна вспомогательная функция потерь, используемая для оценки текстуры, несущей информацию о форме лица. Исходная базовая модель не имеет каких-либо перепадов между соседними вершинами, поэтому мы хотим избежать ситуации, когда расшифровщик формы создаёт нам эти визуальные проблемы. Сравнивая вершину со средним значением по всем соседям, мы штрафуем за обнаруженные выбросы в виде выступов или углублений.

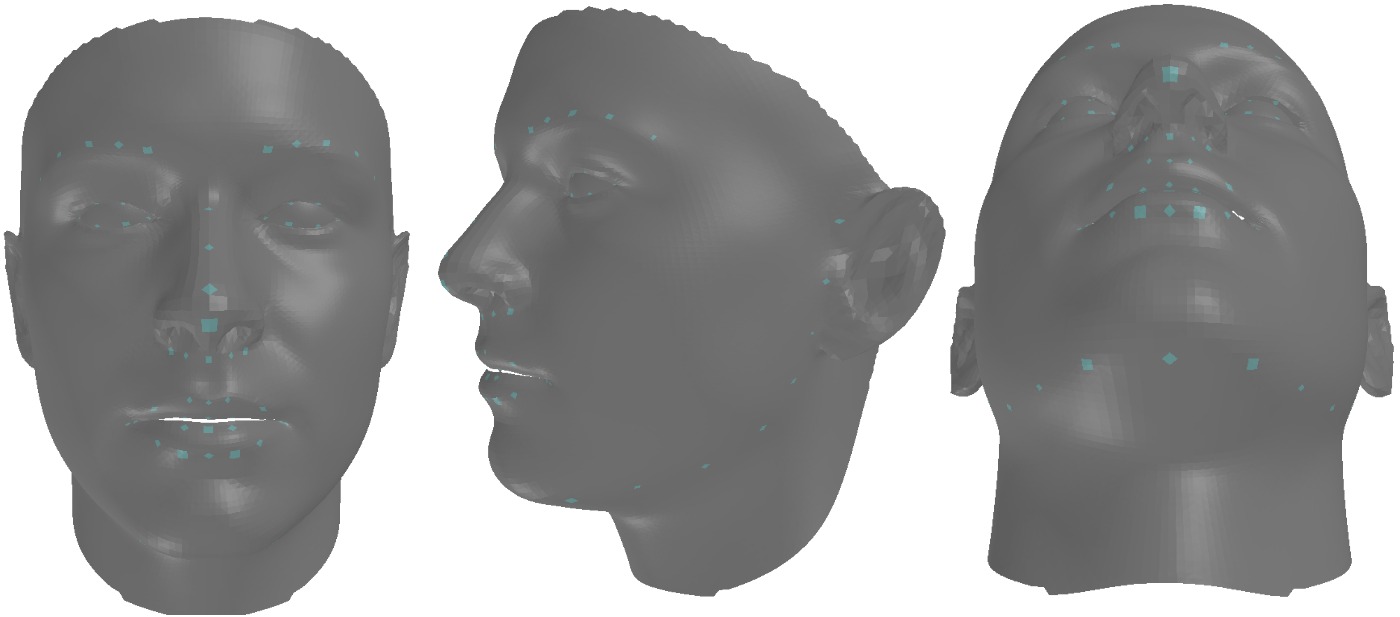
1. Функция потерь по ключевым точкам

(18)

(19)

Эта функция потерь используется в слегка специфической форме: классическая задача поиска ключевых точек лица здесь отсутствует, вместо этого точки даны номерами вершин на базовой модели, а функция потерь высчитывается из разности координат каждой из вершин для исходной и реконструированной форм. Таким образом, эта функция потерь оценивается в вершинном пространстве, но здесь отбрасывается глубина (координата z), чтобы имитировать потери на двумерных изображениях.

На рисунке 5 продемонстрированы сравниваемые ключевые точки:



Голубым цветом выделены положения сравниваемых ключевых точек

Рисунок 5 – Ключевые точки на базовой модели лица

1. Функция потерь по весам нейронных сетей (регуляризация)

(20)

Стандартная функция потерь, известная как *L2­*-регуляризация. Позволяет снижать вероятность и степень переобучения, штрафуя за большие веса.

Описание всех функций потерь проводилось в рамках одного экземпляра (одна пара изображений, текстур, вершин или векторов). В рамках мини-пакетного обучения мы также будем делить на количество экземпляров в пакете.

Итоговая система имеет сложную структуру, и ИНС обучались были чересчур долго, поэтому в [7] предложено разделить обучение на две фазы:

* Фаза первичного обучения, в ходе которой используются заготовленные данные о вершинах, текстурах и проекциях, и функция потерь учитывает сходства с этими «идеальными» значениями, а сходство изображений оставляет в стороне.
* Фаза вторичного обучения, в ходе которой заготовленные данные убираются, и функция потерь работает непосредственно с изображениями.

Объясняется такое разделение тем, что если начинать сразу с потерь по изображениям, то нейросети на первых итерациях могут уйти слишком далеко от исходного решения, поскольку данных для изменения весов оказывается недостаточно, с риском ухода в локальные минимумы и/или седловые точки и, соответственно, невозвращения к адекватным значениям.

В результате имеем следующие итоговые функции потерь:

(21)

(22)

## Используемые программные инструменты

И так, мы определились с видом модели. Для построения макета нужно озаботиться следующими большими подзадачами:

1. Построение нейронных сетей, а также программы для проведения обучения;
2. Работа с базовой трёхмерной моделью (деформирование, наложение текстур и др.);
3. Визуализация результата для демонстрационного макета.

С точки зрения обучения ИНС важным является то, что у системы будет очень много весов и в то же время будет использоваться множество различных ресурсоёмких операций; при этом для обучения глубоких нейронных сетей может потребоваться большое число эпох обучения, что потребует много времени. В силу этого обстоятельства необходимо найти библиотеки, реализации методов в которой хорошо оптимизированы и подвержены распараллеливанию как между ядрами центрального процессора, так и между центральным и графическим процессорами.

Для создания высокопроизводительных программ в задачах машинного обучения имеется множество различных пакетов: *Deep Learning Toolbox* для *MATLAB* [30], *Intel Math Kernel Library* и *Intel Deep Neural Network Library* для *C++* [31, 32], *Theano* для *Python* [33], а также несколько экземпляров с поддержкой множества языков – *TensorFlow* [34]*, PyTorch* [35]*, Caffe* [36]. Первые два пакета имеют проприетарную лицензию, в связи с чем их использование невозможно, разработка *Theano* прекращена, у *PyTorch* есть проблемы с работой на *Windows*. Из двух оставшихся вариантов я предпочту *TensorFlow*, поскольку успел познакомиться с этой библиотекой в рамках выполнения учебных работ.

Таким образом, здесь оптимальным вариантом является *TensorFlow* по следующим причинам:

* Широкий набор операций по работе с многомерными данными и непосредственно по работе с нейронными сетями;
* Возможность использования дополнения *TensorFlow Graphics* – дополнительного пакета, используемого для работы с трёхмерной графикой;
* Наличие рабочей сборки под *Windows*;
* Наличие программного интерфейса на *Python* при исходном коде на *C++*, а также возможность работы на некоторых других языках программирования;
* В рамках реализации на *Python* имеются совместимость с популярными пакетами (*NumPy, SciPy* и др.) или же понятные механизмы для перевода модели с этих пакетов на *TensorFlow*;
* Полноценная поддержка *GPU* и интеграция с *NVIDIA CUDA;*
* Сравнительно высокая популярность среди исследователей.

Как итог, библиотека отлично подходит для построения и обучения макета в рамках нашей задачи, поскольку есть возможность получить результат без лишних (с точки зрения ВКРБ) сложностей.

Остановившись на *Python*, мы сильно сузили выбор и в библиотеках для работы с трёхмерной графикой, поскольку абсолютное большинство используются для *C++* в силу использования *OpenGL*. Среди имеющихся вариантов приятно выделяется *PyRender* [37] – пакет для трёхмерной визуализации на базе интерфейса *OpenGL* для *Python*. В рамках данного пакета можно создавать трёхмерные сцены, задавать положения объектов, источники света, а также проводить рендеринг с выводом на экран или без него. Последний случай нам интересен для синтезирования изображений: можно деформировать базовую модель, повернуть её, натянуть на неё текстуру и задать освещение, после чего провести рендер трёхмерной сцены и получить таким образом синтезированное лицо, готовое для сравнения с исходным.

*PyRender* также опирается на *Trimesh* [38] – эта библиотека предназначена уже не столько для рендеринга (хотя возможность имеется), сколько для работы с трёхмерной моделью как со структурой данных. *Trimesh* позволяет загружать трёхмерные модели различных форматов, проводить необходимые манипуляции и сохранять эти модели в различных же форматах обратно. Тогда создаваемая система позволит использовать результаты работы в различных приложениях для работы с трёхмерными моделями.

## Создание приложения с графическим интерфейсом

Для макета также понадобится «внешнее» приложение, то есть интерфейс, соответствующий техническому заданию (приложение В).

Графическое приложение в данном случае будем создавать с помощью фреймворка *Qt* [39]. Это очень мощный инструмент для создания десктопных и мобильных приложений, изначально предназначенный для *C++,* но также имеющий официальный интерфейс для *Python*.

Функционал требуется небольшой, в связи с чем мы разместим все *UI*-элементы на главном окне. Соответственно, этот функционал может быть разделён на три части:

1. Подсистема входных данных, где пользователь выбирает изображение для подачи на обработку

Здесь нам требуется по сути дела только один элемент интерфейса – кнопка загрузки изображения. Для более приятной работы и во избежание случайных ошибок мы также будем демонстрировать пользователю загруженное изображение и путь до файла.

1. Подсистема обработки изображений, где происходит общение между внешней программой и проектируемой системой

В данном блоке тоже обязателен только один элемент – кнопка запуска обработки образца. Но мы также добавим небольшое окно для системных сообщений, чтобы пользователь сразу видел возникающие сбои.

Обработку изображения следует вынести в отдельный исполняемый поток, поскольку процесс длится до 10 секунд, что при выполнении в основном потоке приведёт к полной блокировке приложения.

1. Подсистема обработки результата, где пользователь может просмотреть созданную модель и сохранить её в одном из общеиспользуемых форматов.

Для просмотра модели сделаем две опции: применение оценённого ракурса и наложение синтезированной текстуры. В результате можно будет оценить работы различных аспектов системы.

Для сохранения модели дадим несколько опций форматов и возможность выбрать, стоит ли сохранять вместе с моделью текстуру.

Также введём дополнительную функцию: возможность сохранить данные, полученные на разных этапах реконструкции (начиная от выхода нейронных сетей и заканчивая синтезированным изображением). Созданное «внешнее» приложение можно будет использовать для анализа результатов, потому что из промежуточных данных можно получить много информации о работе системы.

Наконец, пропишем логику блокировки интерфейсных элементов, поскольку в рамках системы требуется соблюдать строгий порядок (например, нельзя запустить обработку изображения до его выбора).

При входе из доступных элементов – только кнопка выбора исходного изображения, поскольку у нас нет входных данных и, соответственно, не из чего получать выходные. Вид приложения в начальный момент времени изображён на рис. 6:

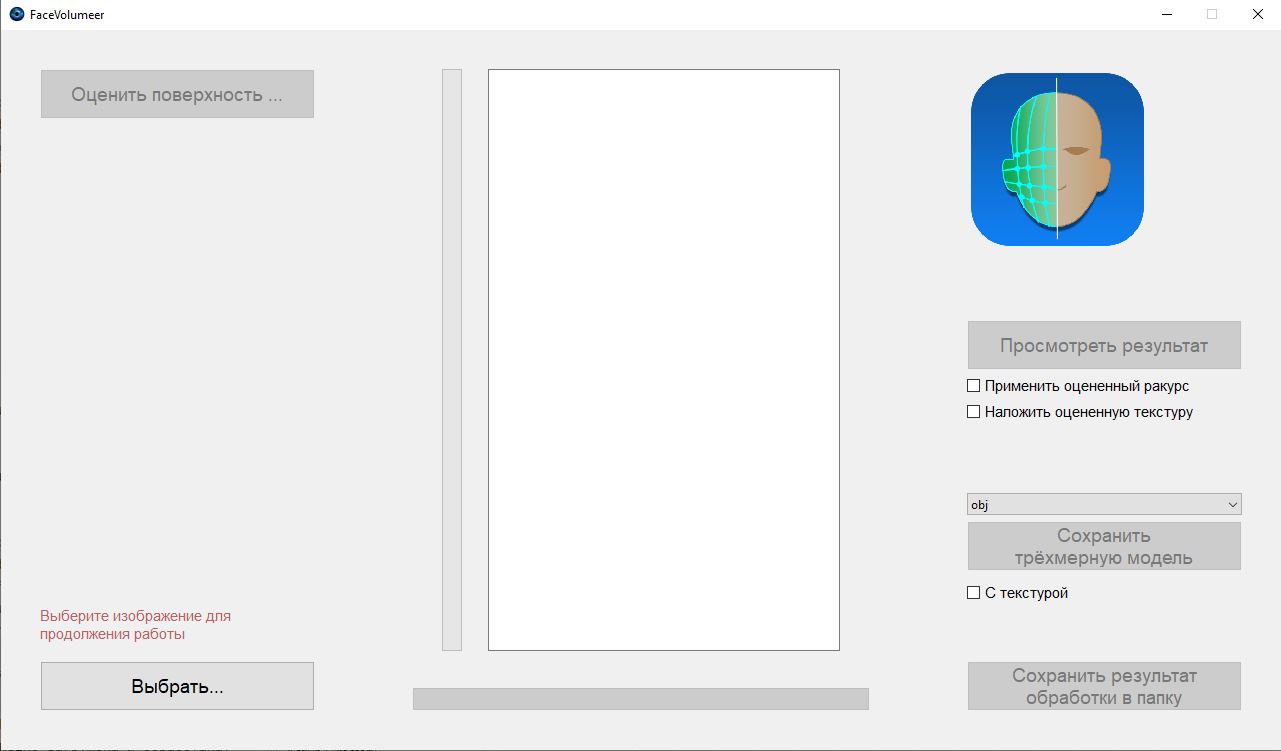


Рисунок 6 – начальный экран интерфейса системы

После выбора изображения нам открывается кнопка запуска системы, рис. 7:

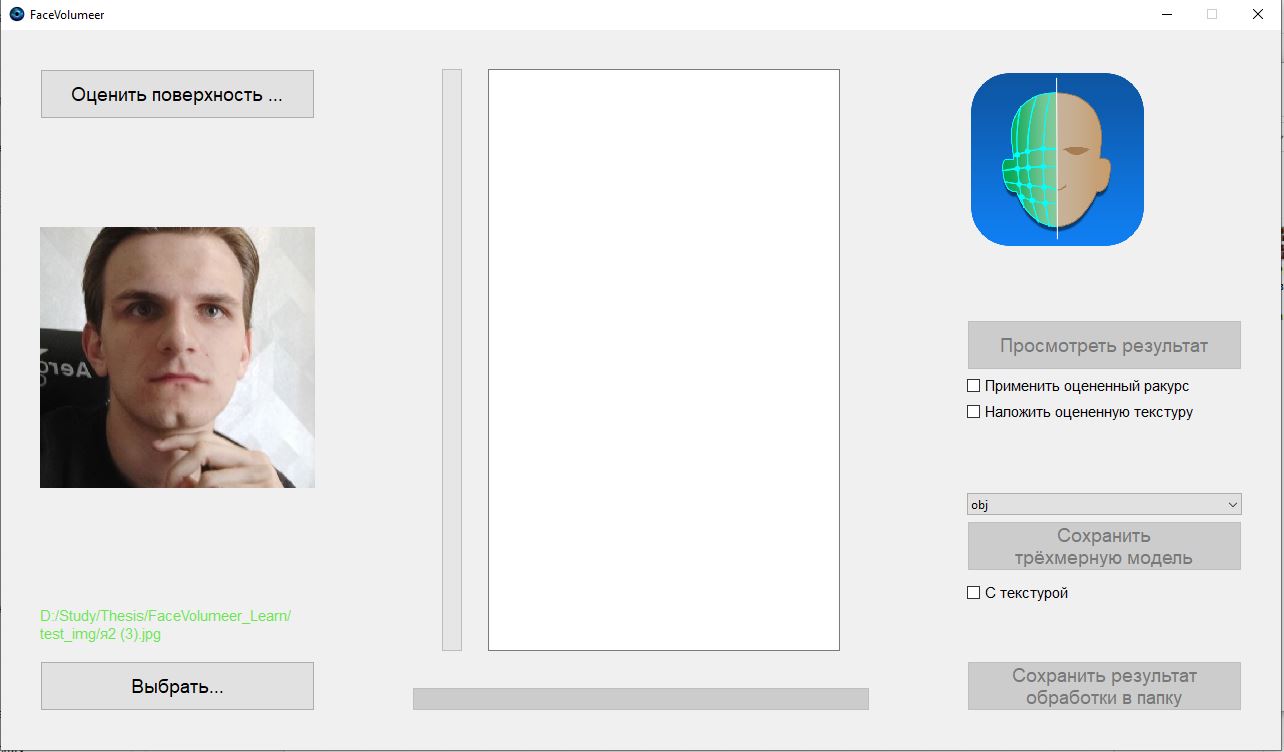


Рисунок 7 – интерфейс системы после выбора входных данных

При оценке поверхности системные сообщения записываются в текстовое поле по центру экрана, слева от этого поля – вертикально-ориентированная шкала прогресса. Состояние приложения по завершении работы системы приведено на рис. 8:

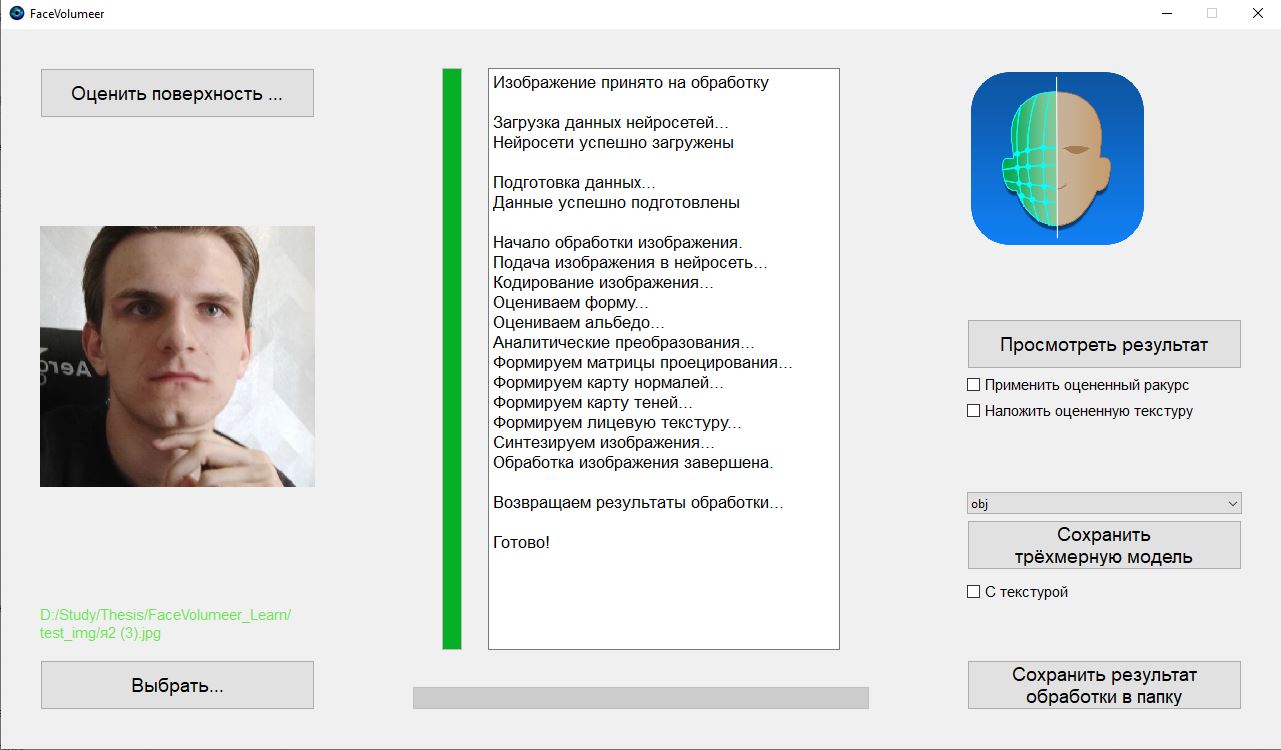


Рисунок 8 – интерфейс системы после оценки трёхмерной поверхности

При этом если запустить оценку заново, то происходит сброс данных и все элементы, связанные с выходными данными (расположены в правой части окна), снова блокируются до завершения работы.

Граф диалога пользователя представлен в приложении А10. Результаты работы приложения с точки зрения выходных данных можно увидеть в следующих приложениях:

* Демонстрация результата реконструкции с помощью приложения в виде трёхмерной модели – приложение А11;
* Демонстрация результата экспорта трёхмерной модели с текстурой и без – приложение А12;
* Демонстрация результата экспорта промежуточных данных – приложение А13;
* Демонстрация возможности работы с получаемыми результатами в редакторе трёхмерных моделей *Blender* [50] – приложение А14.

## Результаты работы

Стоит сделать два замечания о проделанной работе. Во-первых, несмотря на заложенное на обучение время в один месяц нейронная сеть не успела до конца обучиться. На используемом компьютере одна эпоха обучения длится в среднем 3 часа, при этом для качественного обучения системы потребовалось бы несколько тысяч эпох. Было проделано множество попыток ускорения обучения за счёт сокращения числа экземпляров, входящих в эпоху, и ручного изменения скорости обучения, однако даже это не помогло довести обучение до логического завершения. Таким образом, подтвердилось сказанное в пункте 3.2.

В приложении А15 приведены графики функции потерь на каждой эпохе; на этих графиках можно заметить то, как часто менялись весовые коэффициенты и другие параметры сети, из-за чего первоначальный график крайне неоднородный; то, что процесс обучения ещё не завершён, подтверждается тем, что значение функции потерь продолжает равномерно «ползти» вниз вплоть до последних эпох; то, что переобучение до сих пор не наступило, определяется тем, что значения функции потерь для обучающей и проверочной выборок остаются примерно равными. Наглядная демонстрация отсутствия переобучения будет дана позже.

Во-вторых, по ходу работы оказалось, что множество функций библиотек всё ещё недоступны на *Windows*, в первую очередь это коснулось части финального рендера изображений (многочисленные проблемы с различными модификациями *OpenGL*).

Единственный рабочий вариант замедлял скорость вычислений в 4-5 раз (выше данное значение в 3 часа на эпоху – это без стадии рендера) и при этом не обеспечивал надёжную работу и повышение качества, в связи с чем задача на ВКРБ была сокращена до первой фазы обучения. Таким образом, все результаты получены при обучении с учителем. Несмотря на возникшие сложности, система была доведена до той стадии, где выходные данные представляют собой осмысленный результат и пригодны для анализа.

В приложении А16 представлены результаты для нескольких образцов из тех, что входили в обучающую выборку. В глаза сразу бросается общая нехватка чувствительности: несмотря на то, что общие признаки по типу цвета лица и наличия бороды/усов передаётся на реконструкции, система не способна распознать детали. Если говорить о выражении лица, то здесь видно, что на текущий момент передаётся либо нейтральное лицо, либо улыбка, причём эти два выражения лица несильно меняются от образца к образцу, то есть выделяется категория, но не конкретика.

Можно также заметить, что на карту теней влияет общий цвет фотографии, однако это не оказывает критического влияния, поскольку в большинстве случаев свет на реконструированной модели распределён достаточно достоверно и близко к входному изображению. В то же время в цветовой текстуре заметны следы затенения; два этих факта вместе позволяют выдвинуть предположение, что нейронные сети ещё не научились разграничивать цвет лица и его освещение, в связи с чем цвета на итоговой текстуре получаются не такими близкими к исходным, как хотелось бы.

В приложении А17 представлены результаты для некоторых образцов, которые нейронная сеть не видела на этапе обучения ни разу. Здесь появляется ещё один аргумент в пользу отсутствия переобучения: мы видим ровно те же проблемы, что и для образцов, используемых в обучении. Точно также схватываются глобальные черты и точно также не хватает чувствительности для передачи индивидуальных деталей. Вдобавок, последний образец позволяет судить о том, что происходит с обработкой кожи возрастного человека: мы видим, что на цветовой текстуре есть некоторые намёки на неровности, но ничего такого нет на карте теней и, соответственно, на итоговой текстуре, то есть участки нейронной сети, отвечающие за реконструкцию формы, не приспособились к работе со складками, морщинами и т. д.

Один важный недостаток системы – отсутствие инвариантных преобразований для масштаба лица. При описании набора данных мы сказали о том, что лица представлены в полном разнообразии ракурсов и что дополнительно мы проводим смещения, однако для полного набора нужны не только перемещение и вращение, но и масштабирование. В связи с тем, что изображения представлены одного размера и с примерно одинаковым процентом пикселей, передающих непосредственно лицо, нейронная сеть оказывается неспособна самостоятельно вычленить лицо на большом изображении для дальнейшей работы с обрезанной версией. В приложении А18 эта проблема проиллюстрирована на примере. Можно видеть, что на самых разных изображениях проблема одна и та же, поскольку на самых первых версиях, где человек наиболее удалён от камеры, реконструированные лица имеют практически идентичный вид, а по мере приближения – приобретают всё больше индивидуальности. По этой причине многие фотографии, не участвовавшие в обучении, были обрезаны вручную.

С другой стороны, заложенная в процесс обучения обработка сдвига показывает себя хорошо несмотря на то, что мы далеки от получения конечного результата. В приложении А19 продемонстрировано, как программа справляется с одной и той же фотографией, подверженной сдвигу.

Оценить устойчивость системы по вращению лица на произвольных образцах без специального оборудования (или хотя бы штатива) сложно. Тем не менее, можно сделать предварительный вывод о том, что система показала себя хорошо. Результат продемонстрирован в приложении А20.

Наконец, посмотрим на то, насколько генерируемые ответы получаются уникальными. На рисунке 9 приведён набор из 12 образцов, которые будут участвовать в сравнении. В этой выборке находятся изображения, сгруппированные и по людям, и по выражениям лица, причём ни один из образцов не участвовал в обучении.

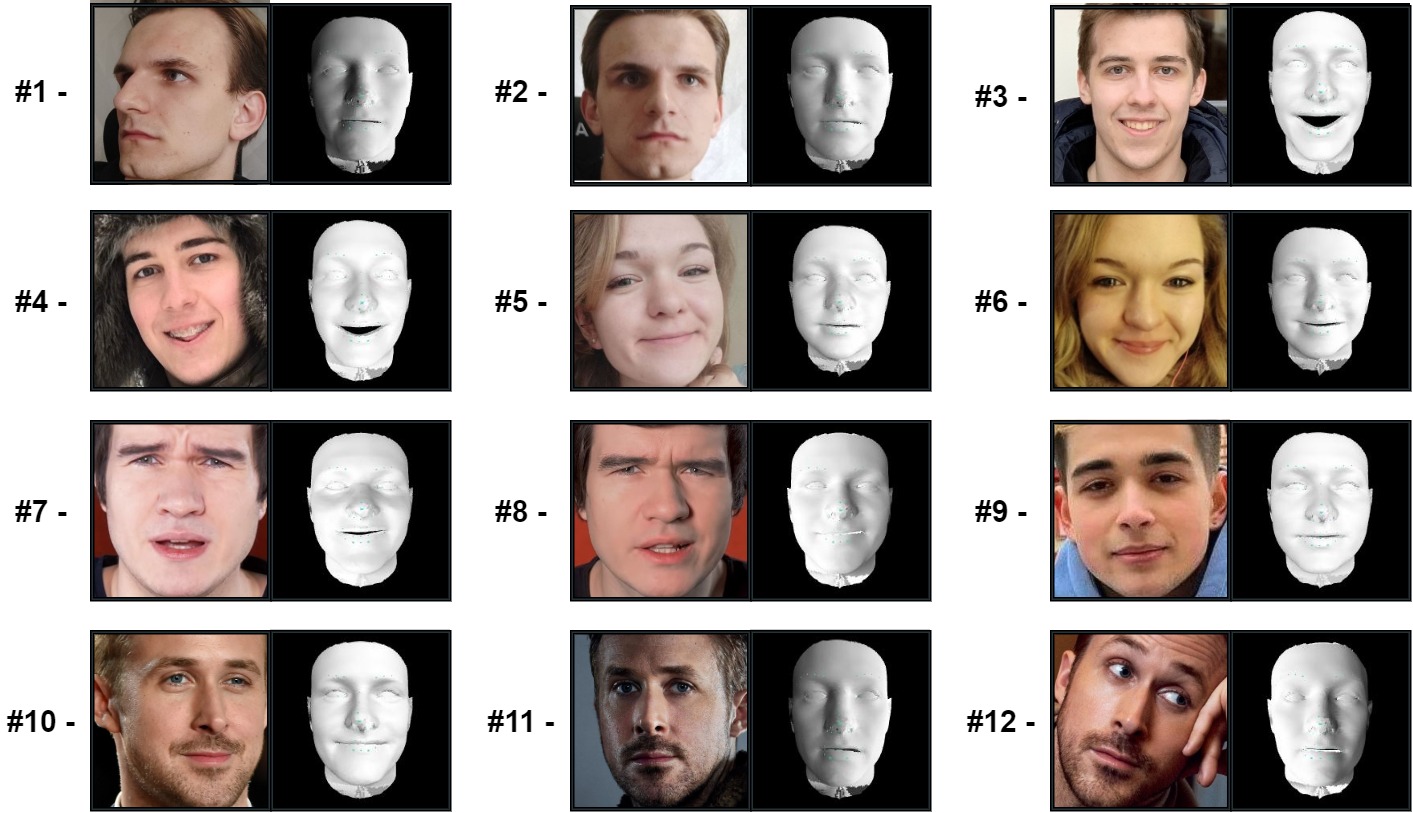


Рисунок 9 – образцы для сравнения на схожесть

В таблице 3 приведён результат сравнения по принципу «каждый с каждым». Для сравнения использовались объёмные текстуры, в качестве меры схожести выступает косинусная мера:

(23)

Таблица 3 – Результат сравнения образцов по объёмным текстурам

(курсивом выделены ячейки, где сравнивались результаты из образцов, относящихся к одному человеку)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
| **1** | 0,0 | *0,21* | 0,9 | 0,78 | 0,47 | 0,48 | 0,3 | 0,3 | 0,32 | 0,28 | 0,51 | 0,45 |
| **2** |  | 0,0 | 0,99 | 0,85 | 0,33 | 0,38 | 0,38 | 0,34 | 0,26 | 0,38 | 0,4 | 0,4 |
| **3** |  |  | 0,0 | *0,22* | 0,86 | 0,77 | 0,73 | 0,82 | 0,84 | 0,81 | 0,96 | 0,96 |
| **4** |  |  |  | 0,0 | 0,72 | 0,68 | 0,72 | 0,82 | 0,69 | 0,78 | 0,73 | 0,71 |
| **5** |  |  |  |  | 0,0 | *0,05* | 0,45 | 0,43 | 0,25 | 0,61 | 0,47 | 0,31 |
| **6** |  |  |  |  |  | 0,0 | 0,35 | 0,35 | 0,21 | 0,54 | 0,58 | 0,35 |
| **7** |  |  |  |  |  |  | 0,0 | *0,08* | 0,2 | 0,15 | 0,8 | 0,54 |
| **8** |  |  |  |  |  |  |  | 0,0 | 0,21 | 0,19 | 0,79 | 0,56 |
| **9** |  |  |  |  |  |  |  |  | 0,0 | 0,34 | 0,58 | 0,35 |
| **10** |  |  |  |  |  |  |  |  |  | 0,0 | *0,76* | *0,6* |
| **11** |  |  |  |  |  |  |  |  |  |  | 0,0 | *0,38* |
| **12** |  |  |  |  |  |  |  |  |  |  |  | 0,0 |

Соответственно, ноль в таблице означает полное совпадение, а полная противоположность соответствует числу два. Приведённые данные показывают, что в большинстве случаев результат ожидаемый: наименьшие значения оказываются при сравнении двух различных реконструированных лиц одного человека. Но нельзя не заметить, что это выполняется не всегда (см., например, всё относящееся к образцу #10).

Если посмотреть на промежуточные данные, а именно на генерируемую нейронными сетями объёмную текстуру, то такой результат оказывается вполне ожидаемым (сопоставление образцов объёмных текстур представлено в приложении А21). Объёмные текстуры группируются по людям небезупречно, но всё же можно видеть схожие рисунки на изображениях одного человека; гораздо интереснее то, что не менее чётко эти текстуры группируются по выражению лица, поскольку мы можем выделить две больших группы – нейтральное выражение лица и улыбка. В рамках этих двух групп участки текстуры формируют практически идентичные паттерны, причём цвета соответствующих участков также практически совпадают, при этом при сравнении изображений одного человека даже с одной формой лица можно видеть существенные различия в цветах. Это ещё раз подтверждает, скажем так, незрелость системы на данном этапе.

## Выводы

Стадия, до которой были доведены нейронные сети, позволяют сделать предварительные выводы. В настоящее время в исследованиях, связанных с темой глубокого обучения, доминируют большие архитектуры, что можно увидеть на примере [6-10] и [21-25] (и не только). Под большими архитектурами понимаем сети, в состав которых входит как минимум три десятка слоёв, причём это число может доходить до нескольких сотен. Созданная в данной работе архитектура имеет гораздо меньшее число слоёв и, соответственно, обучаемых параметров, нежели в передовых технологиях.

Несмотря на «скромную» архитектуру, можно сказать, что у системы есть большой потенциал к улучшению и что может оказаться так, что она будет достойно (если не выигрышно) смотреться в сравнении с «большими братьями». Предел возможностей не был исследован вследствие чересчур долгого обучения, а результаты на текущий момент пусть и не слишком впечатляют, но всё же демонстрируют способность программы к выявлению признаков и к построению поверхности, которая и схожа с тем, что мы видим на фотографии, и уникальна в рамках одного человека даже на разных образцах, и не имеет серьёзных проблем с точки зрения чисто визуальной (например, выбросы на некоторых вершинах). Сложность и длительность процесса обучения не стала неожиданностью, поскольку это предполагалось ещё до запуска программы и было обосновано в пункте 3.2.

Система в текущем виде не способна создавать такие поверхности, которые позволяли бы отличить одного человека от другого и, соответственно, для работы в системах распознавания требуется существенная доработка, но это ни в коем случае не означает, что работа была проделана зря и что от всех заложенных идей придётся оказаться. Таким образом, можно заключить, что мы движемся в правильном направлении, но при этом о хоть сколь полном решении задачи речи не идёт даже близко.

# ЗАКЛЮЧЕНИЕ

Результатом выпускной квалификационной работа бакалавра является рабочая система трёхмерной лицевой реконструкции. В ходе проделанной работы было проведено исследование основных методологий, связанных с получением трёхмерной лицевой поверхности из фотографии без вспомогательных данных. По результатам данного исследования была спроектирована система, сочетающая принципы глубокого обучения и компьютерной графики. Для обеспечения доступа к системе пользователям было создано графическое приложение для персональных компьютеров, позволяющее загрузить в неё любое изображение, представленное в одном из популярных форматов, и провести реконструкцию с целью дальнейшего экспорта в виде 3D-модели и/или просмотра результата.

Выходные данные генерируются корректно, какие-либо артефакты (выбросы) отсутствуют, экспортированная трёхмерная модель может быть использована в других средах без каких-либо ограничений. Созданная система хорошо себя показала несмотря на незавершённость обучения и небольшое количество внутренних слоёв нейронных сетей, и таким образом дальнейшее изучение и улучшение становится возможным и полностью обоснованным. Дальнейшая работа может быть направлена как на работу с визуализацией (то есть, с большим уклоном в лицевые текстуры) так и на работу с идентификацией и прочими аспектами, связанными непосредственно с формой лица.

В рамках развития системы первоочерёдной задачей является доработка подсистема рендера изображения для обеспечения полного цикла обучения на двумерных изображениях. Следует также провести более глубокий анализ подходов для оценки возможности сочетания различных методов (например, применение не только самой фотографии, но и результата применения к ней различных фильтров).

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Horn, Berthold & Brooks, Michael. (1989). *Shape from Shading*. // ResearchGate URL: [https://www.researchgate.net/publication/230687001\_  
   Shape\_from\_Shading](https://www.researchgate.net/publication/230687001_Shape_from_Shading) (дата обращения: 31.05.2022).
2. *Shape From Shading* // Perception URL: <https://perception.inrialpes.fr/Publications/2006/PF06a/chapter-prados-faugeras.pdf> (дата обращения: 31.05.2022).
3. *Shape From Shading: A Survey* // ResearchGate URL: <https://www.researchgate.net/publication/262213310_Shape_from_Shading_A_Survey> (дата обращения: 31.05.2022).
4. 3D Face Reconstruction from a Single Image Using a Single Reference Face Shape // ResearchGate URL: https://www.researchgate.net/publication/  
   49716955\_3D\_Face\_Reconstruction\_from\_a\_Single\_Image\_Using\_a\_Single\_Reference\_Face\_Shape (дата обращения: 31.05.2022).
5. *A Morphable Model for the Synthesis of 3D Faces* // University of Basel URL: <https://gravis.dmi.unibas.ch/publications/Sigg99/morphmod2.pdf> (дата обращения: 31.05.2022).
6. "Освещение", лекция из курса "Компьютерная Графика" // Казанский Федеральный Университет URL: https://kpfu.ru/staff\_files/  
   F823558158/Lecture5.pdf (дата обращения: 31.05.2022).
7. *On Learning 3D Face Morphable Model from In-the-wild Images* // Michigan State University URL: [https://cvlab.cse.msu.edu/pdfs/Tran\_Liu\_CVPR2018  
   .pdf](https://cvlab.cse.msu.edu/pdfs/Tran_Liu_CVPR2018.pdf) (дата обращения: 31.05.2022).
8. *3D Face Reconstruction in Deep Learning Era: A Survey* // ResearchGate URL: https://www.researchgate.net/publication/357726978\_3D\_Face\_Reconstruction  
   \_in\_Deep\_Learning\_Era\_A\_Survey (дата обращения: 31.05.2022).
9. *GANFIT: Generative Adversarial Network Fitting for High Fidelity 3D Face Reconstruction* // ResearchGate URL: https://www.researchgate.net/  
   publication/331195716\_GANFIT\_Generative\_Adversarial\_Network\_Fitting\_for\_High\_Fidelity\_3D\_Face\_Reconstruction (дата обращения: 31.05.2022).
10. *Deep video portraits* // arXiv URL: https://arxiv.org/abs/1805.11714 (дата обращения: 31.05.2022).
11. АЛГОРИТМ ПОСТРОЕНИЯ ДЕФОРМИРУЕМЫХ 3D МОДЕЛЕЙ ЛИЦА И ОБОСНОВАНИЕ ЕГО ПРИМЕНИМОСТИ В СИСТЕМАХ РАСПОЗНАВАНИЯ ЛИЧНОСТИ // Научная Электронная библиотека URL: <https://www.elibrary.ru/item.asp?id=29296340> (дата обращения: 31.05.2022).
12. ТРЕХМЕРНАЯ РЕКОНСТРУКЦИЯ ЛИЦА ИЗ ОДНОГО 2D-ИЗОБРАЖЕНИЯ НА ОСНОВЕ ОТЛИЧИТЕЛЬНЫХ ОСОБЕННОСТЕЙ И БАЗЕЛЬСКОЙ МОДЕЛИ ЛИЦА // Научная Электронная библиотека URL: https://www.elibrary.ru/item.asp?id=46542520 (дата обращения: 31.05.2022).
13. *Facial Landmark Detection by Deep Multi-task Learning* // ResearchGate URL: https://www.researchgate.net/publication/264786906\_Facial\_Landmark\_Detect  
    ion\_by\_Deep\_Multi-task\_Learning (дата обращения: 31.05.2022).
14. *Generative Adversarial Nets* // arXiv URL: <https://arxiv.org/pdf/1406.2661.pdf> (дата обращения: 31.05.2022).
15. *A User’s Guide to Spherical Harmonics* // Ohio University URL: <http://ohiouniversityfaculty.com/mohlenka/research/uguide.pdf> (дата обращения: 31.05.2022).
16. *Spherical Harmonic Lighting: The Gritty Details* // Gotesburg University URL: http://www.cse.chalmers.se/~uffe/xjobb/Readings/GlobalIllumination/  
    Spherical%20Harmonic%20Lighting%20-%20the%20gritty%20details.pdf (дата обращения: 31.05.2022).
17. *Spherical Harmonics* // ResearchGate URL: https://www.researchgate.net/publication/236136792\_Spherical\_Harmonics (дата обращения: 31.05.2022).
18. *A survey on analysis of human faces and facial expressions datasets* // ResearchGate URL: https://www.researchgate.net/publication/335007375  
    \_A\_survey\_on\_analysis\_of\_human\_faces\_and\_facial\_expressions\_datasets (дата обращения: 31.05.2022).
19. *60 Facial Recognition Databases* // Kairos URL: https://www.kairos.com/blog/60-facial-recognition-databases (дата обращения: 31.05.2022).
20. *Face Alignment Across Large Poses: A 3D Solution* // Semantic Scholar URL: <https://arxiv.org/pdf/1511.07212.pdf> (дата обращения: 31.05.2022).
21. *ImageNet Classification with Deep Convolutional Neural Networks* // ResearchGate URL: https://www.researchgate.net/publication/267960550\_ImageNet  
    \_Classification\_with\_Deep\_Convolutional\_Neural\_Networks (дата обращения: 31.05.2022).
22. *Deep Face Recognition* // Semantic Scholar URL: https://www.semanticscholar.org/paper/Deep-Face-Recognition-Parkhi-Vedaldi  
    /162ea969d1929ed180cc6de9f0bf116993ff6e06 (дата обращения: 31.05.2022).
23. *Going deeper with convolutions* // ResearchGate URL: https://www.researchgate.net/publication/305196650\_Going\_deeper\_with\_convolutions (дата обращения: 31.05.2022).
24. *Rethinking the Inception Architecture for Computer Vision* // Semantic Scholar URL: https://www.semanticscholar.org/paper/Rethinking-the-Inception-Architecture-for-Computer-Szegedy-Vanhoucke  
    /23ffaa0fe06eae05817f527a47ac3291077f9e58 (дата обращения: 31.05.2022).
25. *Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning* // ResearchGate URL: https://www.researchgate.net/publication/  
    301874967\_Inception-v4\_Inception\_ResNet\_and\_the\_Impact\_of\_Residual\_  
    Connections\_on\_Learning (дата обращения: 31.05.2022).
26. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // arXiv URL: https://arxiv.org/abs/1502.03167 (дата обращения: 31.05.2022).
27. *Adam: A Method for Stochastic Optimization* // ResearchGate URL: <https://www.researchgate.net/publication/269935079_Adam_A_Method_for_Stochastic_Optimization> (дата обращения: 31.05.2022).
28. Функции активации нейросети: сигмоида, линейная, ступенчатая, ReLu, tanh // Neurohive RU URL: <https://neurohive.io/ru/osnovy-data-science/activation-functions/> (дата обращения: 31.05.2022).
29. "Системы координат в трёхмерной графике", лекция из курса "Компьютерная Графика" // Казанский Федеральный Университет URL: <https://kpfu.ru//staff_files/F_326219710/Lecture4.pdf> (дата обращения: 31.05.2022).
30. *MATLAB Deep Learning Toolbox* // MathWorks URL: <https://www.mathworks.com/products/deep-learning.html> (дата обращения: 31.05.2022).
31. *Intel® oneAPI Math Kernel Library* // Intel URL: <https://www.intel.com/content/www/us/en/developer/tools/oneapi/onemkl.html> (дата обращения: 31.05.2022).
32. *Intel® oneAPI Deep Neural Network Library* // Intel URL: <https://www.intel.com/content/www/us/en/developer/tools/oneapi/onednn.html> (дата обращения: 31.05.2022).
33. *Theano Library* // GitHub URL: <https://github.com/Theano/Theano> (дата обращения: 31.05.2022).
34. *TensorFlow: Open Source Platform for Machine Learning* // TensorFlow URL: <https://www.tensorflow.org/> (дата обращения: 31.05.2022);
35. *PyTorch: An Open Source Machine Learning Framework* // PyTorch URL: <https://pytorch.org/> (дата обращения: 31.05.2022).
36. *Caffe: Deep Learning Framework* // Caffe URL: <https://caffe.berkeleyvision.org/> (дата обращения: 31.05.2022).
37. *PyRender* // GitHub URL: <https://github.com/mmatl/pyrender> (дата обращения: 31.05.2022).
38. *Trimesh* // Trimesh Library URL: <https://trimsh.org/> (дата обращения: 31.05.2022).
39. *Qt for Python* // Qt URL: <https://doc.qt.io/qtforpython/> (дата обращения: 31.05.2022).
40. Метод главных компонент // Университет ИТМО URL: <https://neerc.ifmo.ru/wiki/index.php?title=Метод_главных_компонент_(PCA)> (дата обращения: 31.05.2022).
41. Метод главных компонент // MachineLearning-RU URL: <http://www.machinelearning.ru/wiki/index.php?title=Метод_главных_компонент> (дата обращения: 31.05.2022).
42. Хайкин С. Нейронные сети: полный курс. - 2-е издание изд. - М.: Вильямс, 2008. - 1104 с.
43. *Intuitively Understanding Convolutions for Deep Learning* // Towards Data Science URL: <https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1> (дата обращения: 15.06.2022).
44. Пакетная нормализация // Университет ИТМО URL: <https://neerc.ifmo.ru/wiki/index.php?title=Batch-normalization> (дата обращения: 15.06.2022).
45. «Раздел 2: Нейронные сети с одним обрабатывающим слоем». Методы поддержки принятия решений. Курс лекций. // Московский государственный технический университет имени Н. Э. Баумана. Автор: Семёнов Д. В. – 2021 г. – 31 с.
46. «Раздел 3: Многослойные персептроны». Методы поддержки принятия решений. Курс лекций. // Московский государственный технический университет имени Н. Э. Баумана. Автор: Семёнов Д. В. – 2021 г. – 49 с.
47. «Раздел 5: Свёрточные нейронные сети». Методы поддержки принятия решений. Курс лекций. // Московский государственный технический университет имени Н. Э. Баумана. Автор: Семёнов Д. В. – 2021 г. – 9 с.
48. «Раздел 6: Автоэнкодерные нейронные сети». Методы поддержки принятия решений. Курс лекций. // Московский государственный технический университет имени Н. Э. Баумана. Автор: Семёнов Д. В. – 2021 г. – 34 с.
49. «Раздел 9: Глубокие нейронные сети». Методы поддержки принятия решений. Курс лекций. // Московский государственный технический университет имени Н. Э. Баумана. Автор: Семёнов Д. В. – 2021 г. – 29 с.
50. *Blender Homepage* // Blender URL: <https://www.blender.org/> (дата обращения: 15.06.2022).
51. Катькало В.С., Веселова А.С., Смельцова С.В. Методические указания для подготовки курсового проекта «SWOT-анализ». М.: Высшая школа бизнеса НИУ ВШЭ, 2021.
52. *Histograms Of Oriented Gradients for Human Detection* // IEEE Explore URL: <https://ieeexplore.ieee.org/document/1467360> (дата обращения: 15.06.2022).
53. *On the Convergence of Adam and Beyond* // arXiv URL: <https://arxiv.org/abs/1904.09237> (дата обращения: 15.06.2022).
54. Заварикин Д. Принципы построения и преимущества системы 3D-распознавания лиц // Технологии защиты. - 2010. - №4.

# ПРИЛОЖЕНИЕ А. ГРАФИЧЕСКИЕ МАТЕРИАЛЫ

А1. Составляющие моделей освещения в компьютерной графике [6]

А2. Результаты работы алгоритма из [4]

А3. Архитектура системы реконструкции из [5]

А4. Архитектура системы реконструкции из [7]

А5. Обобщённая блок-схема реконструкции

А6. Аугментация входных данных

А7. Архитектура кодирующей нейронной сети

А8. Архитектура расшифровывающих нейронных сетей

А9. Подробная блок-схема алгоритма реконструкции

А10. Граф диалога пользователя

А11. Просмотр результата реконструкции через интерфейс системы

А12. Экспорт трёхмерной модели

А13. Экспорт промежуточных данных

А14. Пример работы с полученными данными в *Blender*

А15. Динамика обучения нейронных сетей

А16. Результаты работы для образцов из обучающей выборки

А17. Результаты работы для произвольных образцов

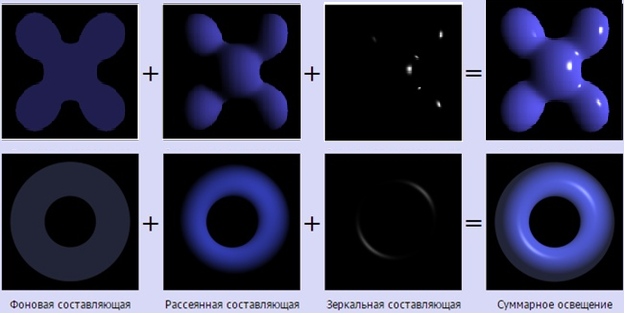
А18. Проблема масштаба лица

А19. Проблема сдвига лица

А20. Проблема вращения лица

А21. Сравнение объёмных текстур

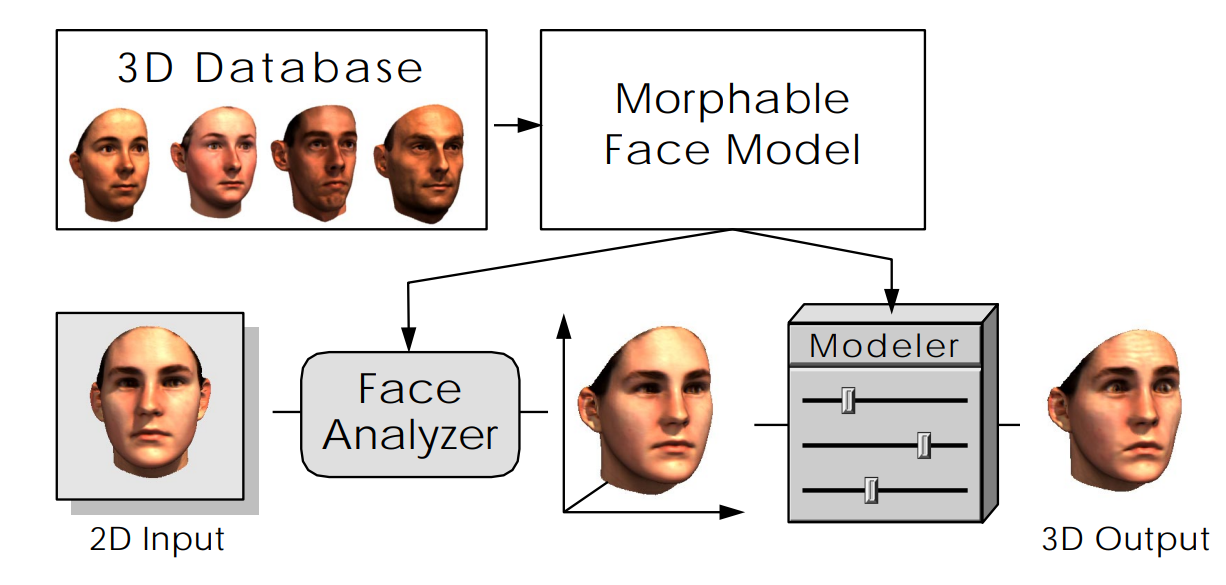
А1 – Составляющие моделей освещения в компьютерной графике [6]



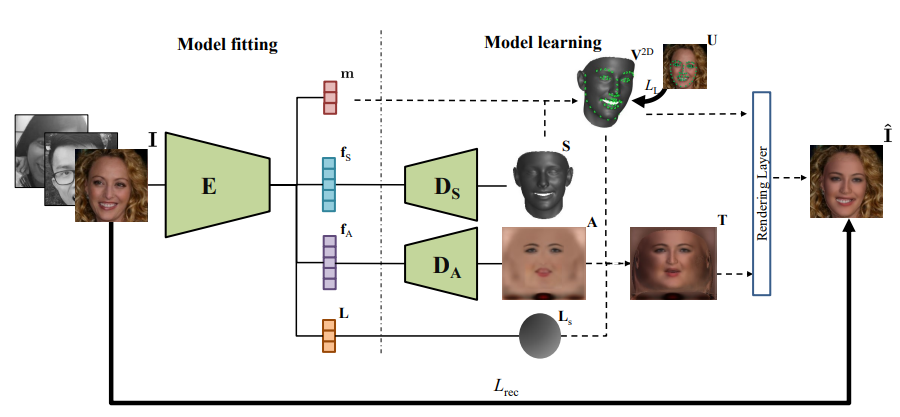
А2 – Результаты работы алгоритма из [4]



А3 –Архитектура системы реконструкции из [5]



А4 – Архитектура системы реконструкции из [7]



*E* – кодирующая свёрточная сеть (*encoder*),

*Ds* – расшифровывающая форму свёрточная сеть (*shape decoder*),

*Da* – расшифровывающая цвет свёрточная сеть (*albedo decoder*),

*m* – выходные данные о ракурсе,

*Fs* – выходные данные о форме (кодовый вектор),

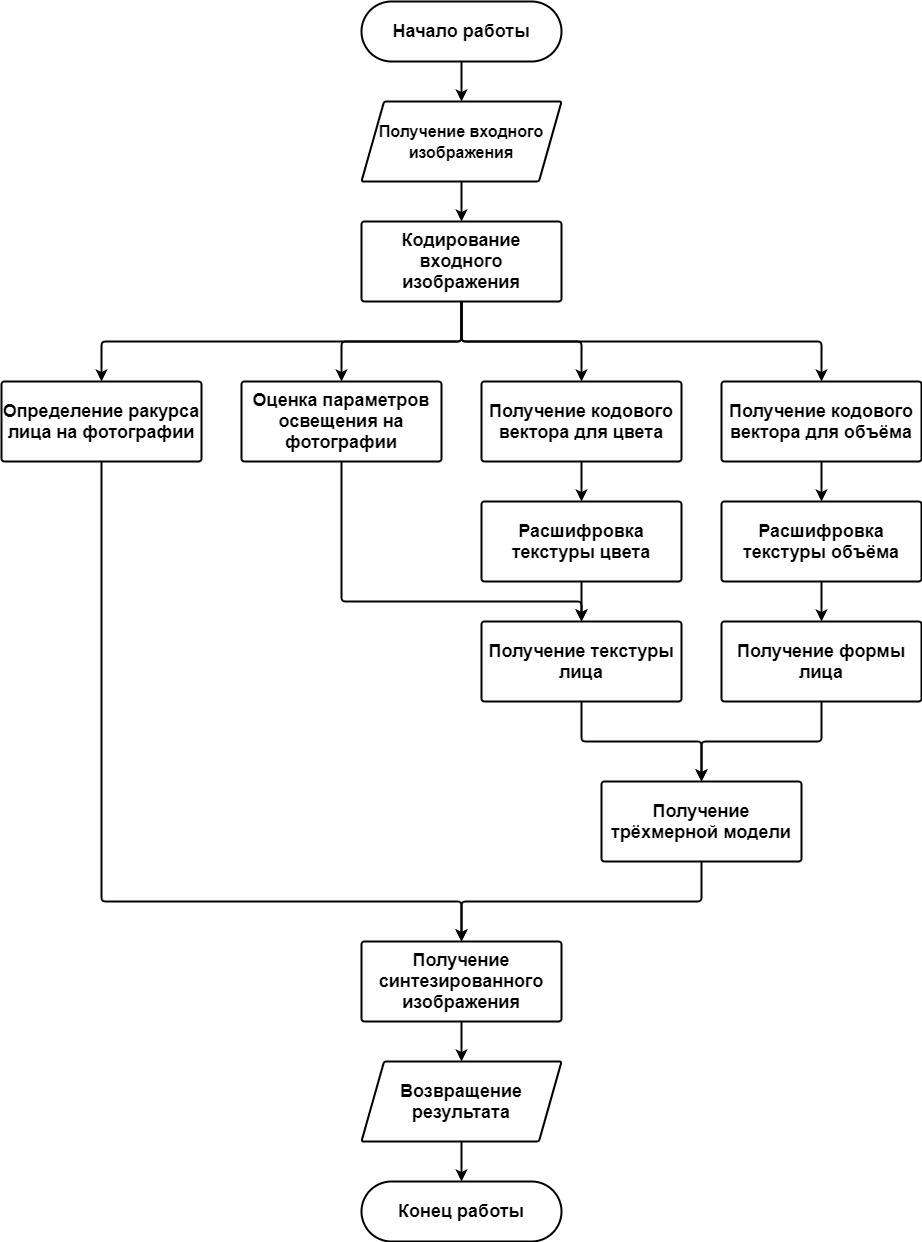
*Fa* – выходные данные о цвете (кодовый вектор),

*L* – коэффициенты разложения гармоник освещения,

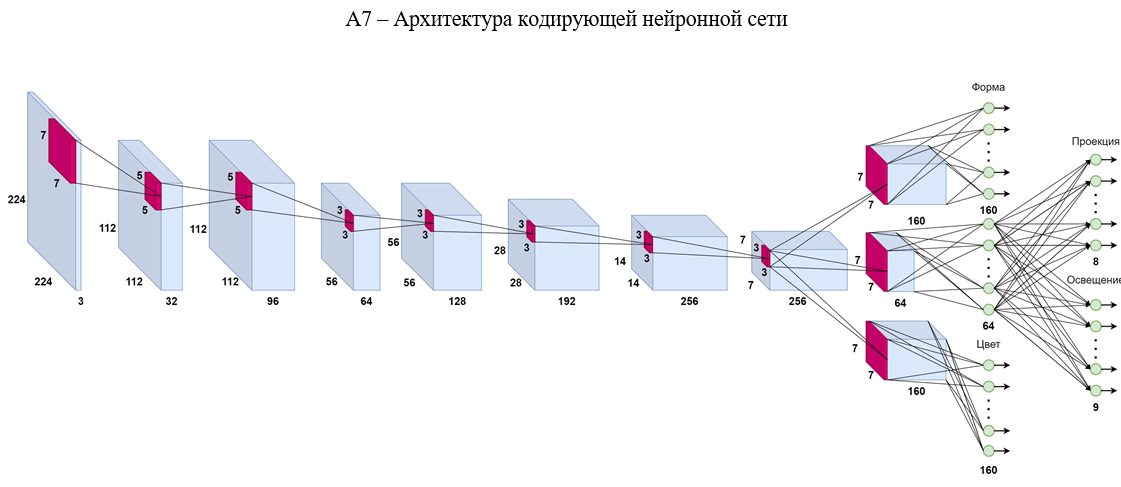
*S* – форма (*shape*), *A* – цвет (*albedo*), *T* – текстура (*texture*)

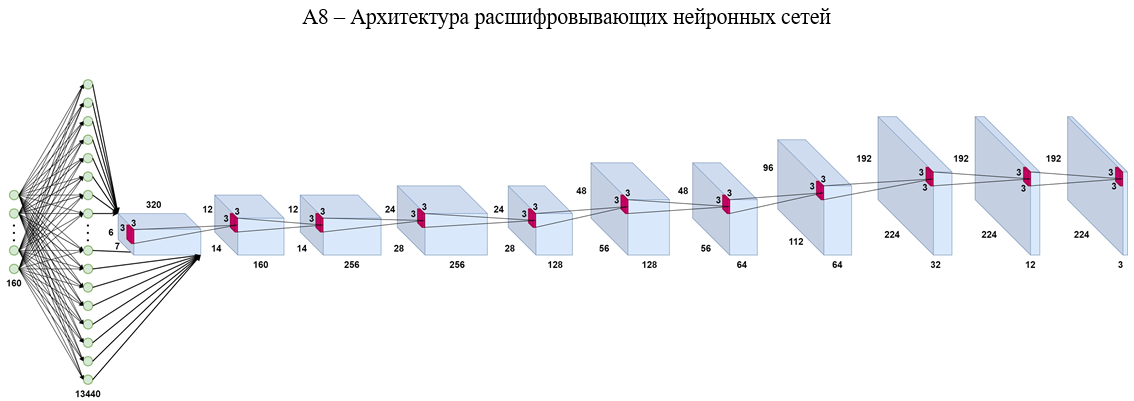
*Rendering Layer* – подсистема синтеза изображений

А5 – Обобщённая блок-схема реконструкции

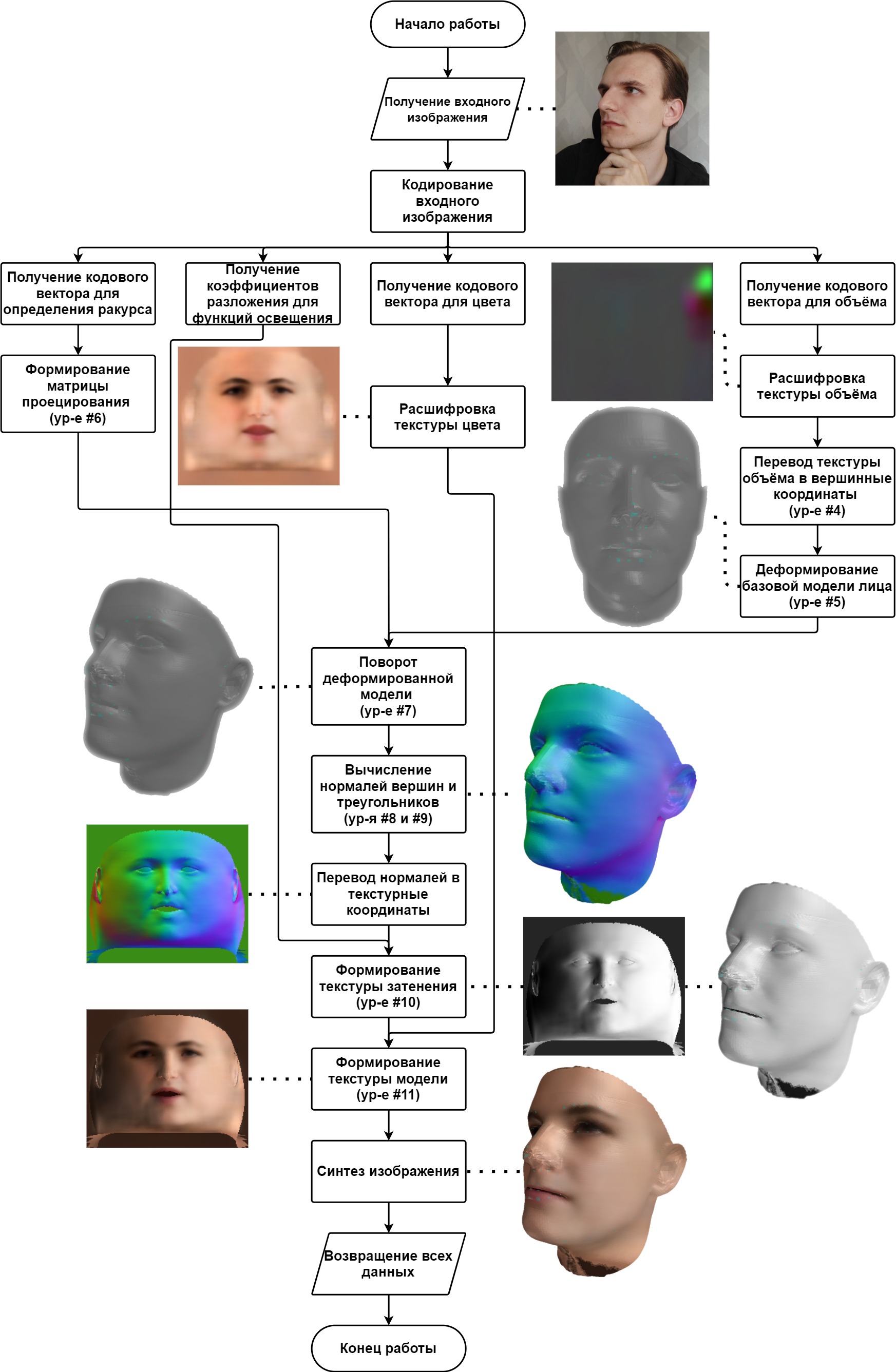


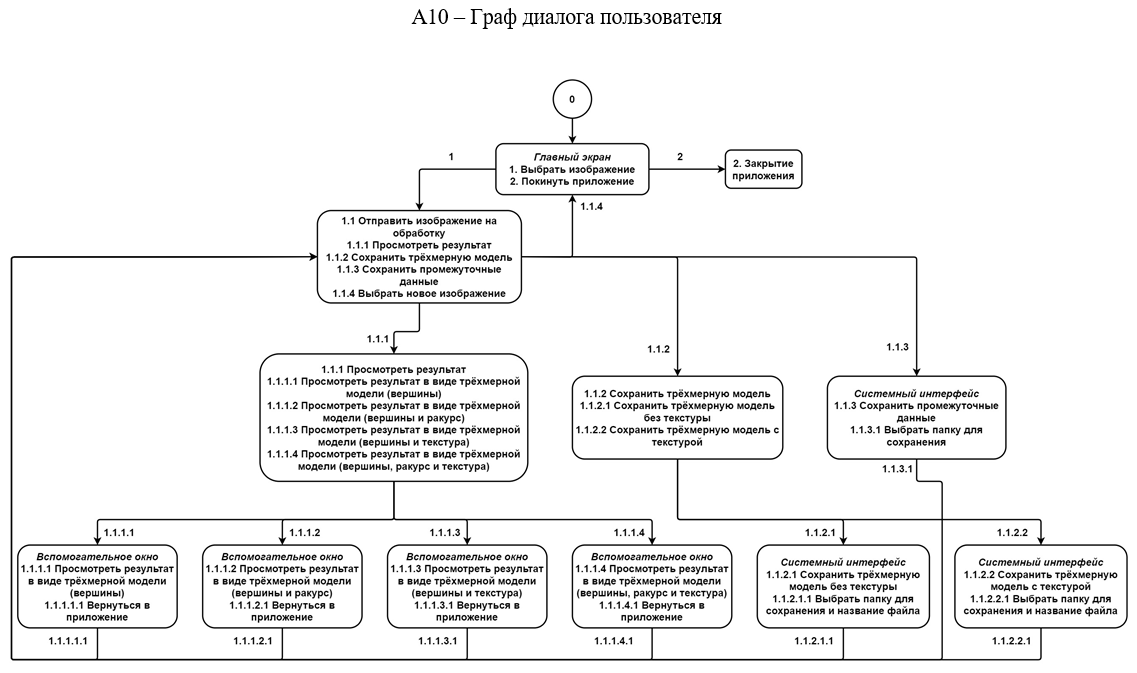




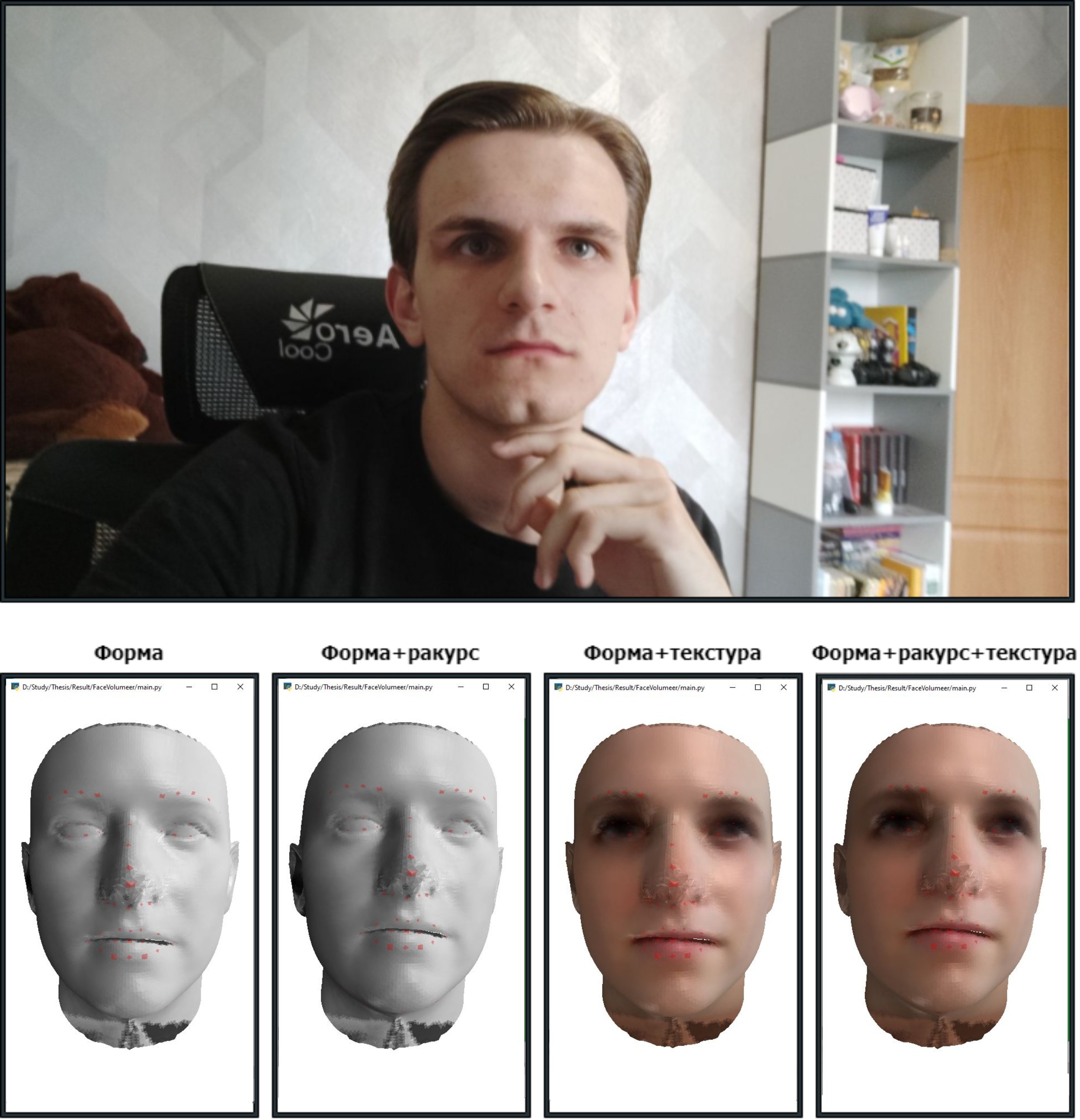


А9. Подробная блок-схема алгоритма реконструкции





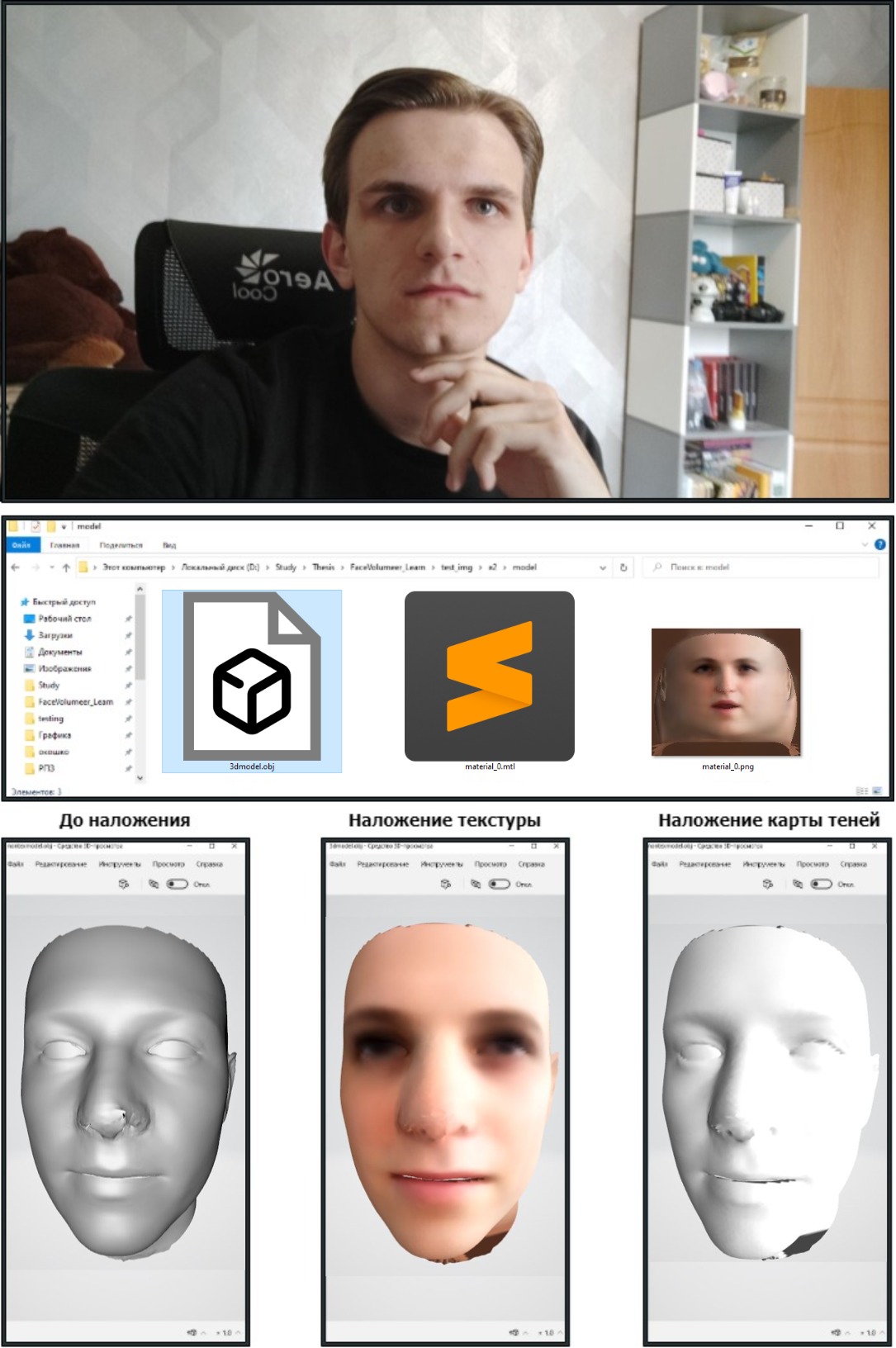
А11 – Просмотр результата реконструкции через интерфейс системы



Верхний ряд: исходное изображение;

Нижний ряд: результат нажатия на кнопку "Просмотреть результат" в приложении при различных настройках

А12 – Экспорт трёхмерной модели

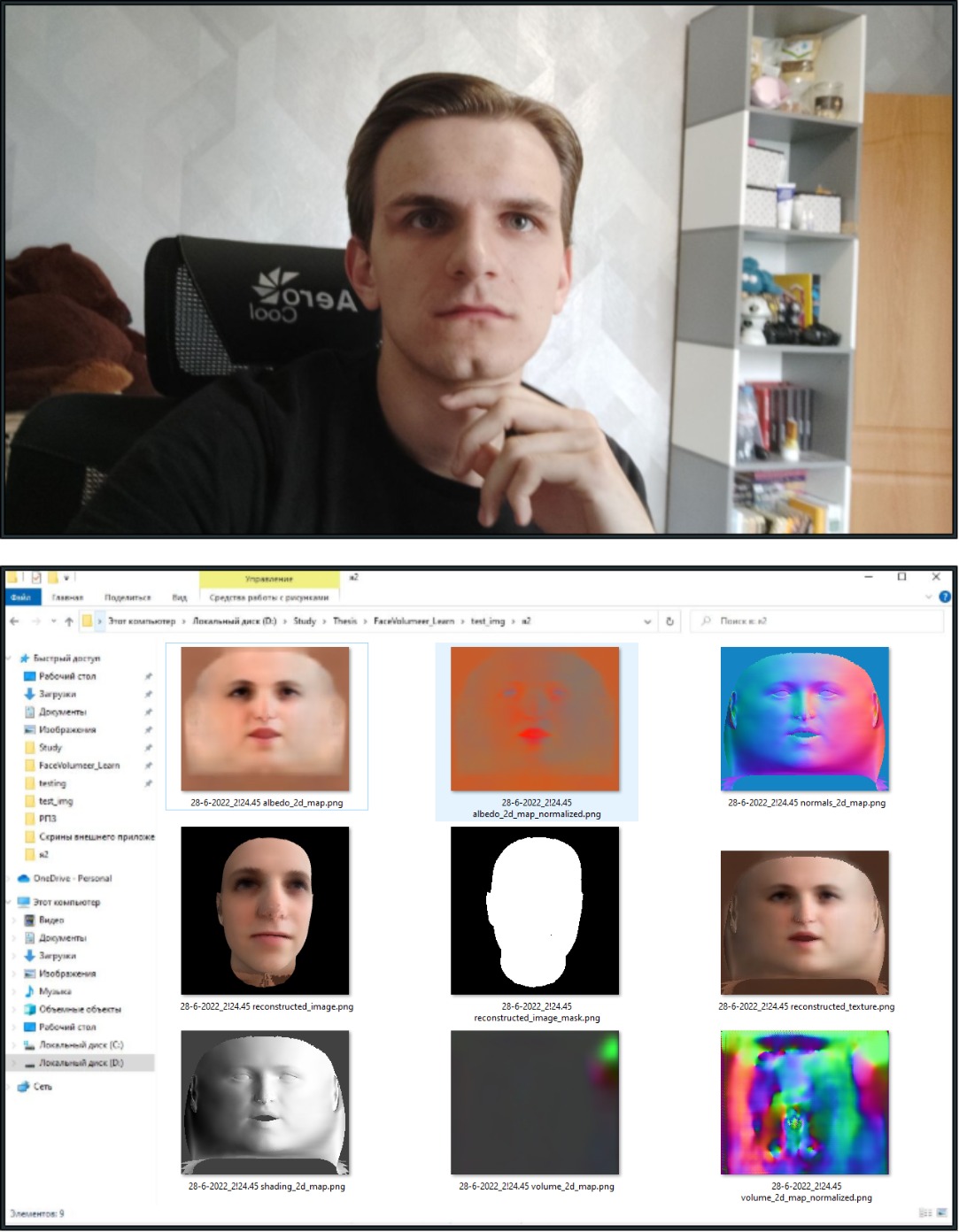


Верхний ряд: исходное изображение;

Средний ряд: результат экспорта трёхмерной модели в папку в формате .*obj*;

Нижний ряд: просмотр модели с помощью стандартной утилиты *Windows 10*

А13 – Экспорт промежуточных данных

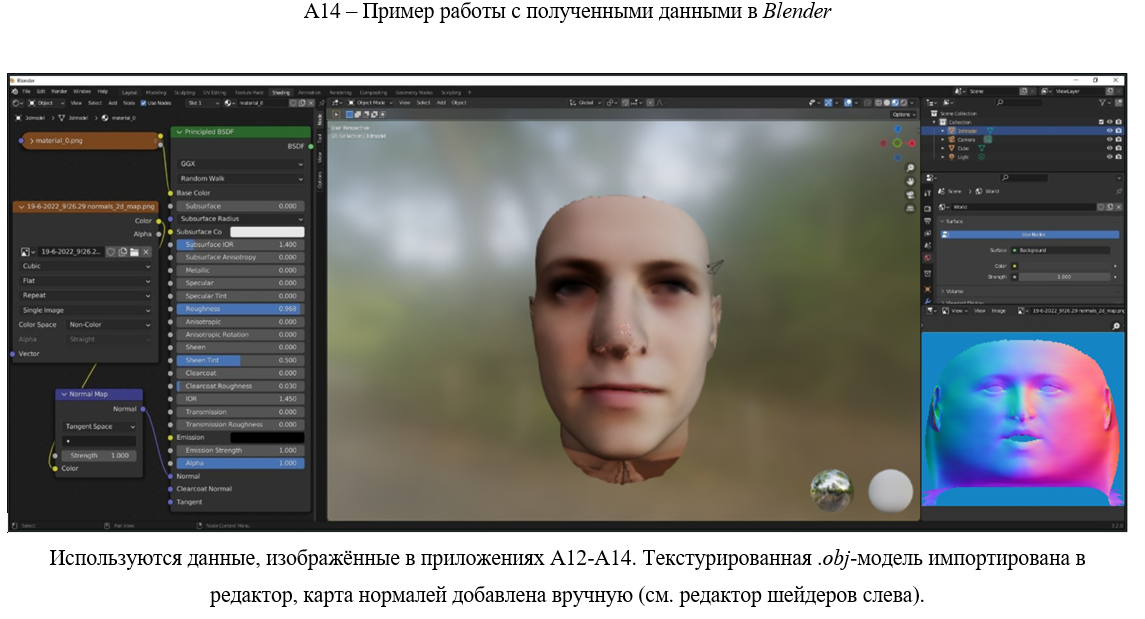


Верхний ряд: исходное изображение

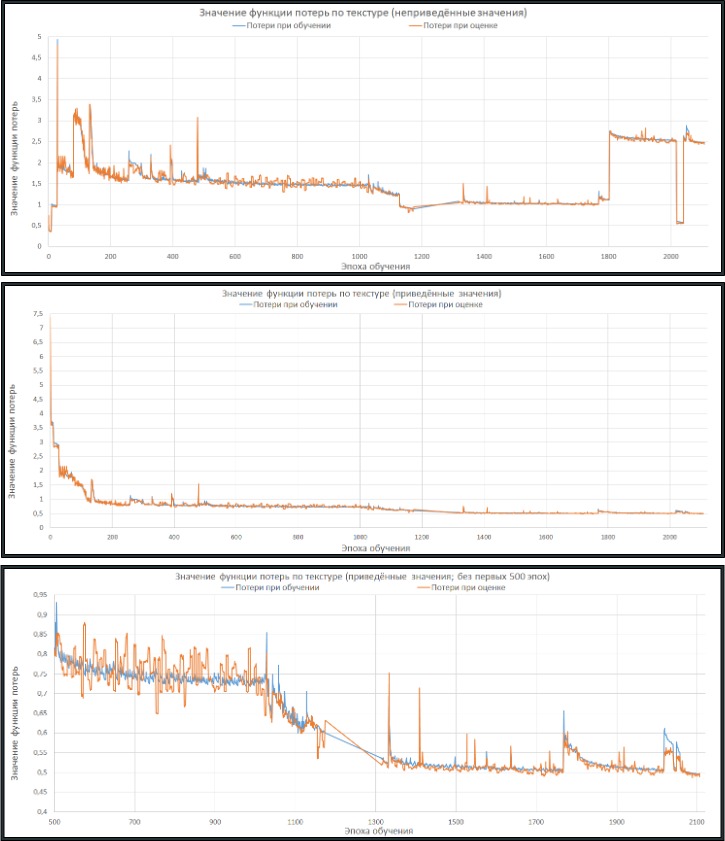
Нижний ряд: папка с промежуточными данными.

Слева направо, по строкам:

* Цветовая текстура, цветовая текстура после применения масштабирования в диапазон [0; 1], карта нормалей;
* Синтезированное изображение, маска синтезированного изображения, итоговая лицевая текстура;
* Карта теней, карта объёма, карта объёма после масштабирования в диапазон [0; 1].



А15 – Динамика обучения нейронных сетей



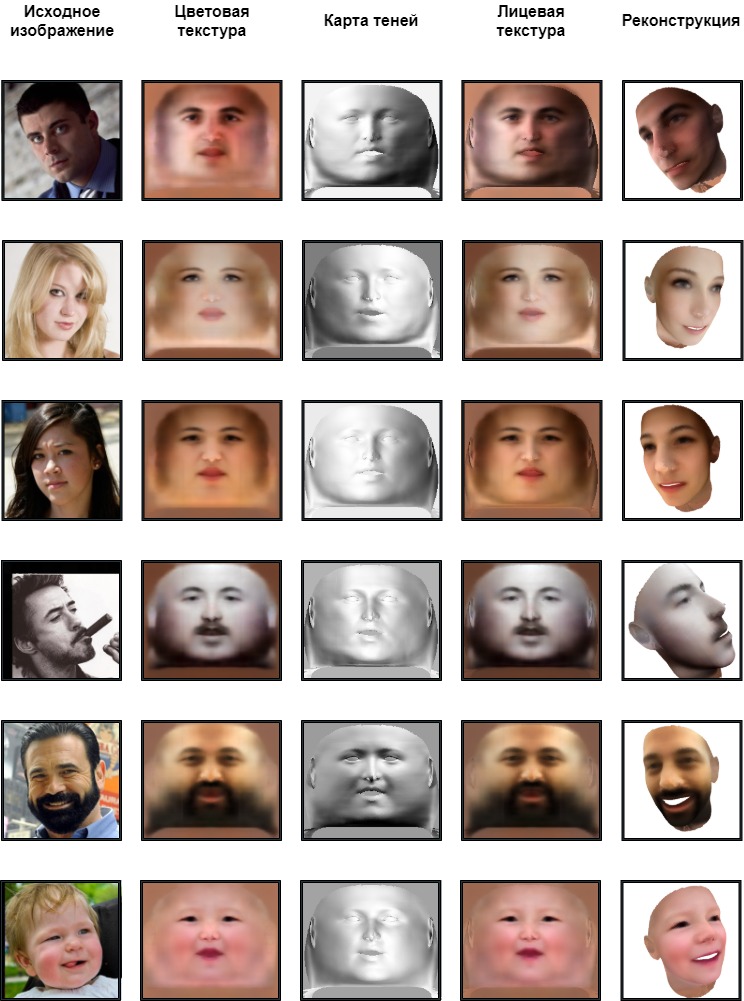
Графики представлены для потерь по текстуре; с остальными слагаемыми ситуация схожая (числа могут быть другие, но вид графика тот же)

Верхний ряд: исходные данные, на которым заметны резкие скачки (следствие изменений весовых коэффициентов);

Средний ряд: масштабированные до единого весового коэффициента данные

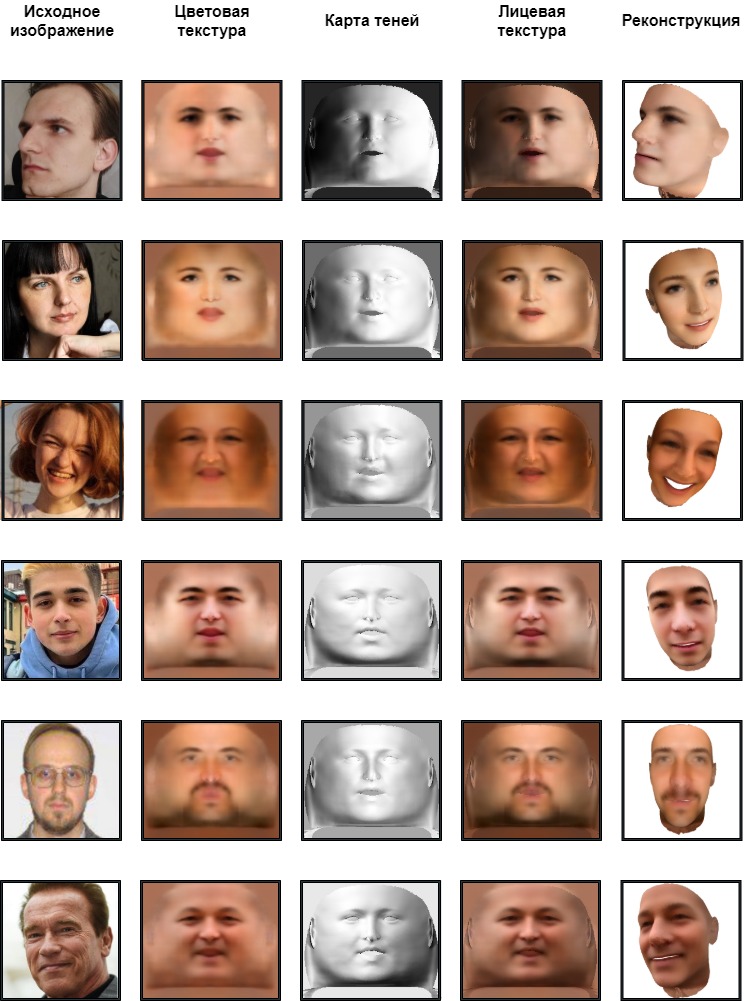
Нижний ряд: то же самое, но начиная с эпохи №500

А16 – Результаты работы для образцов из обучающей выборки

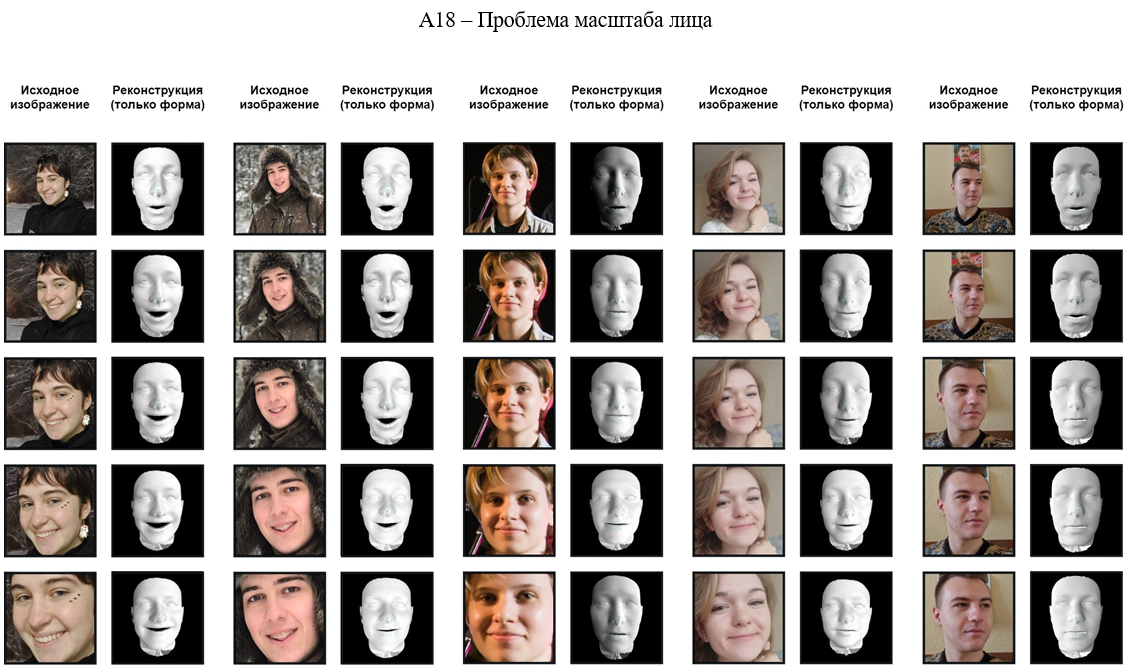


\* - пропорция лица может быть искажена из-за проблемного рендера, а не из-за некачественной работы системы

А17 – Результаты работы для произвольных образцов



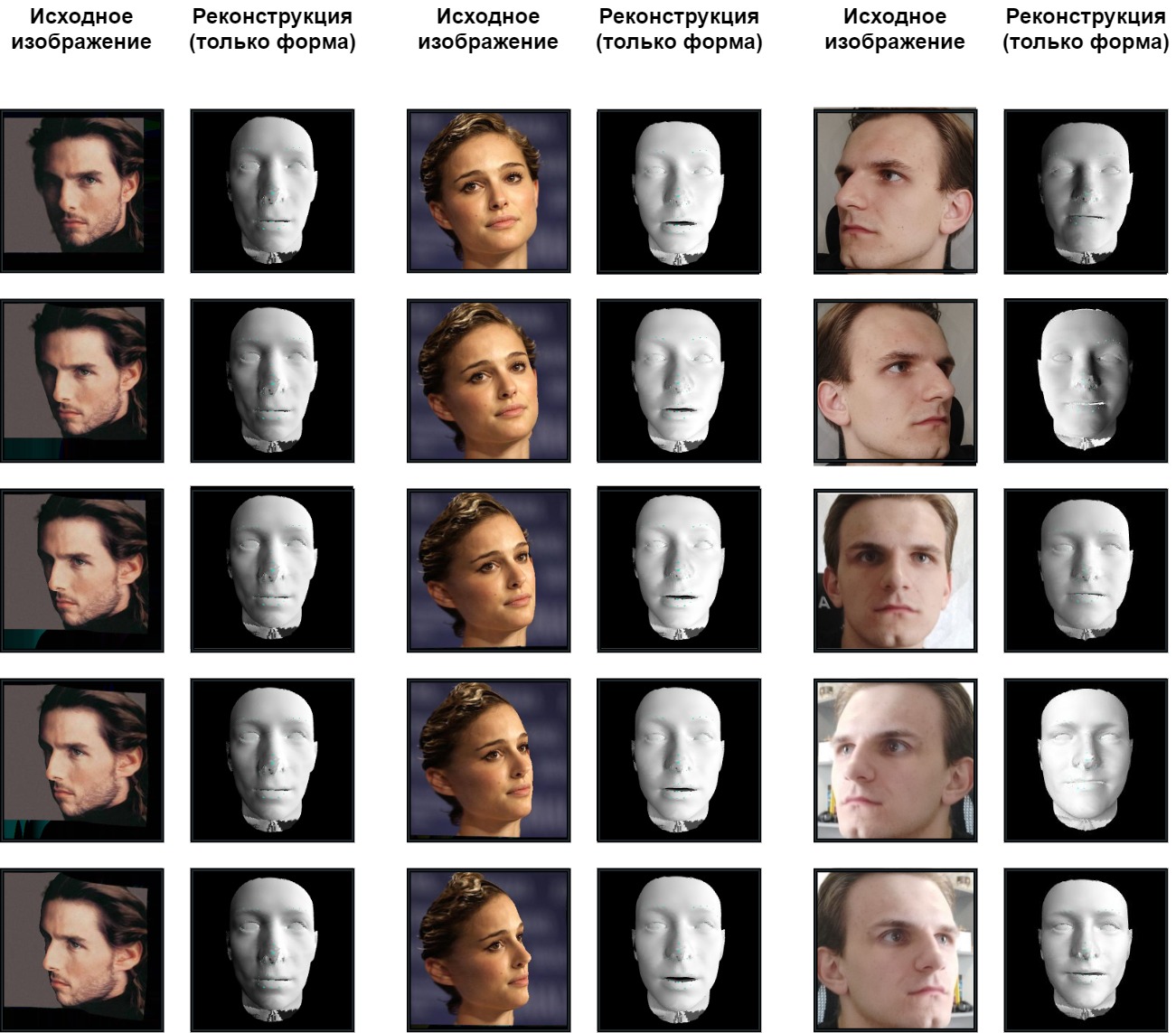
\* - пропорция лица может быть искажена из-за проблемного рендера, а не из-за некачественной работы системы



А19 – Проблема сдвига лица

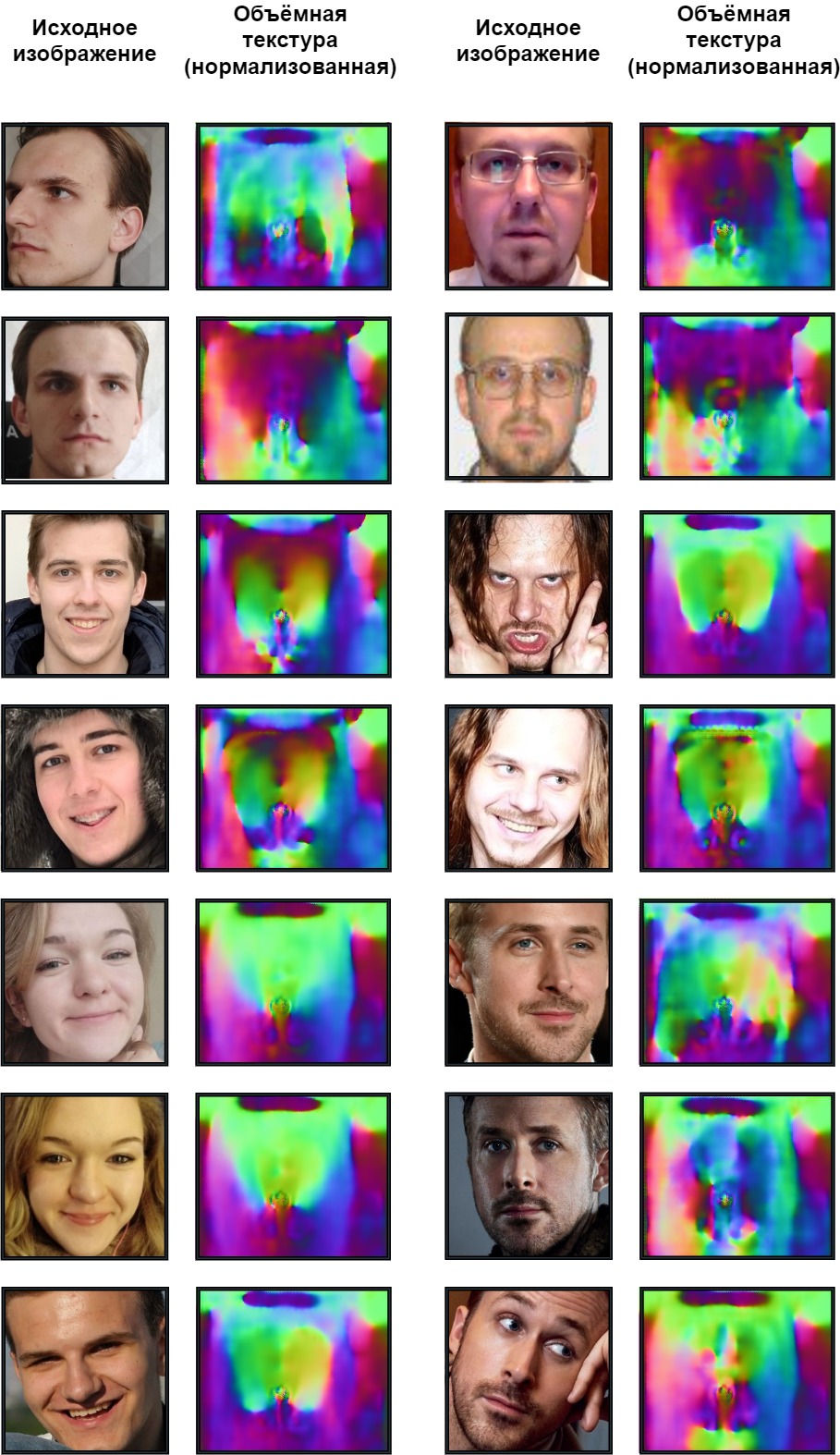


А20 – Проблема вращения лица



В правом столбце первое, третье и пятое изображения – это три разных снимка, а второе и четвёртое – соответствующие зеркальные отражения

А21 – Сравнение объёмных текстур



# ПРИЛОЖЕНИЕ B. ТЕХНИЧЕСКОЕ ЗАДАНИЕ

# 

|  |  |
| --- | --- |
|  | **Министерство образования и науки Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение высшего образования**  **«Московский государственный технический университет** **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

**Кафедра «Системы обработки информации и управления»**

Утверждаю:

Научный руководитель  
 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Семёнов Д. В.

(подпись) (ФИО)

\_\_ \_\_\_\_\_ 2022 г.

**Система реконструкции трёхмерной поверхности человеческого лица на основе фотографий**

Техническое задание

(вид документа)

писчая бумага

(вид носителя)

­5

(количество листов)

Выполнил:  
Забурунов Леонид Вячеславович, группа РТ5-81Б

1 марта 2022 г.

г. Москва

### 1. Наименование программы

«Система реконструкции трёхмерной поверхности человеческого лица на основе фотографий, сделанных без помощи узкоспециализированных устройств»

### 2. Основание для разработки

Основанием является задание на выполнение выпускной квалификационной работы бакалавра (ВКРБ), подписанное научным руководителем.

### 3. Назначение разработки

Результатом выполнения работы будет являться прикладная программа, позволяющая пользователю загружать фотографию человеческого лица и загружать созданную на её основе трёхмерную модель лица в одном из общепринятых форматов.

Созданный программный продукт позволит визуализировать человека в виртуальном пространстве и использовать получаемую модель во всех областях, связанных с трёхмерной компьютерной графикой: видеоигры, приложения для работы в виртуальной и дополненной реальностях, приложения для синтеза фотографий и прочие.

### 4. Требования к программе

### 4.1 Требования к архитектуре создаваемой программы

Программный продукт должен быть реализован в виде графического приложения с использованием платформы *Qt* для языка *Python 3*.

Специальные требования по внутренней структуре программных объектов не предъявляются.

### 4.2 Требования к функциональным характеристикам

Система должна работать по принципу системы реального времени, то есть с наложением временных ограничений на обработку одной фотографии.

Основные требования к возможностям, предоставляемым пользователю при пользовании программой:

1. Выбор исходной фотографии для загрузки в систему (допускается использование стандартного интерфейса *Windows*);
2. Преобразование фотографии в трёхмерную модель лица;
3. Просмотр результата преобразования в отдельном окне, предназначенном для визуализации модели. Это окно должно поддерживать базовые функции для рассмотрения модели: приближение/отдаление и смена ракурса;
4. Возможность получения сформированной трёхмерной модели в виде файла формата *.obj*.

### 4.3 Требования к входным и выходным данным

Входными данными является исходная фотография человеческого лица, загружаемая пользователем.

Выходными данными является созданная на основе находящегося на фотографии лица трёхмерная модель, представленная в одном из форматов, поддерживаемых различными программными пакетами, предназначенными для работы с трёхмерной графикой.

### 4.4 Требования к надёжности создаваемой программы

Программа не должна выдавать непредвиденных ошибок при выполнении, а создаваемая 3D-модель должна быть пригодной для чтения другими инструментами для работы с трёхмерной графикой.

### 4.5 Требования к составу технических средств

Для использования программы требуется персональный компьютер, состоящий из комплектующих, использующих микроархитектуру *AMD64* (*x64*). Рекомендуется:

1. Процессор, имеющий не менее 4 ядер, работающих на тактовой частоте не менее 2 ГГц;
2. Оперативная память объёмом не менее 6ГБ;
3. Дискретный видеоадаптер, способный обеспечить работу приложения (включая просмотр созданной 3D-модели);
4. Совместимые периферийные устройства.

### 4.6 Требования к составу программных средств

Для обеспечения работоспособности создаваемой программы требуется операционная система *Windows 7/8/10/11*.

### 5. Этапы работы

|  |  |  |
| --- | --- | --- |
| **#** | **Этап** | **Срок выполнения** |
| 1 | Исследование предметной области | Март 2022 г. |
| 2 | Утверждение технического задания | Март 2022 г. |
| 3 | Проектирование нейросетевой системы реконструкции | Март-Апрель 2022 г. |
| 4 | Обучение нейронных сетей | Март-Апрель 2022 г. |
| 5 | Создание первичного макета | Апрель 2022 г. |
| 6 | Тестирование и отладка программы | Апрель 2022 г. |
| 7 | Создание полноценного приложения | Апрель-Май 2022 г. |
| 8 | Оформление документации | Май-Июнь 2022 г. |
| 9 | Защита работы | Июнь 2022 г. |

### 6. Требования к программной документации

Требуется наличие следующих единиц технической документации:

1. Техническое задание;
2. Программа и методика испытаний;
3. Описание программы;
4. Руководство пользователя.

### 7. Исполнитель

Студент МГТУ им. Баумана, группы РТ5-81Б, Забурунов Леонид Вячеславович.

### 8. Порядок приема работы

Приём и контроль программного изделия осуществляется в соответствие с методикой испытаний (см. документ «Программа и методика испытаний»).

### 9. Дополнительные условия

Данное техническое задание может уточняться в установленном порядке.

# ПРИЛОЖЕНИЕ C.

### ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ

|  |  |
| --- | --- |
|  | **Министерство образования и науки Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение высшего образования**  **«Московский государственный технический университет** **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

**Кафедра «Системы обработки информации и управления»**

Утверждаю:

Научный руководитель  
 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Семёнов Д. В.

(подпись) (ФИО)

\_\_ \_\_\_\_\_ 2022 г.

**Система реконструкции трёхмерной поверхности человеческого лица на основе фотографий**

Программа и методика испытаний

(вид документа)

писчая бумага

(вид носителя)

­4

(количество листов)

Выполнил:  
Забурунов Леонид Вячеславович, группа РТ5-81Б

1 мая 2022 г.

г. Москва

### 1. Объект испытаний

Объектом испытаний является программа оценки объёмной поверхности человеческого лица на основе фотографий.

### 2. Цель испытаний

Испытания проводятся с целью проверки работоспособности различных элементов программы и соответствия выполняемых программой функций требованиям, предъявленным в техническом задании.

### 3. Состав предъявляемой документации

Перед проведением испытаний предъявляются следующие документы:

1. Техническое задание;
2. Программа и методика испытаний.

### 4. Технические требования

### 4.1 Требования к программной документации

Состав программной документации должен удовлетворять требованиям документа «Техническое задание».

Программная документация должна быть оформлена в соответствии с ГОСТ и ЕСПД по составлению и оформлению документов на программное изделие.

### 4.2 Требования к техническому обеспечению

### 4.2.1 Требования к аппаратному обеспечению

Испытания программы будут проводиться с использованием следующего аппаратного обеспечения:

1. Процессор *Intel Xeon E5645*, работающий на частоте 3,8 ГГц;
2. Видеокарта Palit JetStream Geforce GTX 1060 6GB;
3. Объём оперативной памяти – 12 ГБ;
4. Объём дисковой подсистемы – 360 ГБ.

### 4.2.2 Требования к программному обеспечению

Испытания программы будут проводиться с использованием следующего программного обеспечения:

1. Операционная система *Windows 10*;
2. Среда разработки *PyCharm 2022.1*.

### 5. Методы испытаний

Перечень испытаний приведён в таблице №1, расположенной ниже

Таблица №1 – Программа проведения испытаний

| **#** | **Действие** | **Требуемый результат** | **Номер соответствующего пункта ТЗ** |
| --- | --- | --- | --- |
| 1 | Запуск приложения | Успешная инициализация и переход к главному окну | – |
| 2 | Нажатие кнопки выбора фотографии | Появление стандартного интерфейса операционной системы для выбора файла | 4.2.1 |
| 3 | Нажатие кнопки оценки трёхмерной поверхности | Изначально – шкала прогресса, по завершению оценки – демонстрация соответствующего текста на главном экране, а также появление кнопки для перехода к быстрому просмотру полученной а также появление кнопки для перехода к быстрому просмотру полученной *3D*-модели | 4.2.2 |
| 4 | Нажатие кнопки перехода к предварительному просмотру результата | Переход к дополнительному окну, в котором появляется трёхмерная сцена с в котором появляется трёхмерная сцена с *3D*-моделью человеческой головы, лицо которой деформировано в соответствии с проведённой оценкой поверхности | 4.2.3 |
| 5 | Нажатие кнопки сохранения полученного результата в формате *.obj* | Появление стандартного интерфейса операционной системы для выбора места сохранения и имени будущего файла | 4.2.4 |
| 6 | Нажатие кнопки выхода из приложения | Успешное закрытие приложения, не приводящее к последствиям в работе операционной системы или других приложений | – |

### 

### 6. Результат испытаний

Испытания считаются пройденными успешно, если при демонстрации программа работала в соответствии с требуемыми для каждого действиями результатами, то есть пользователь смог:

* Успешно запустить программу;
* Успешно загрузить фотографию лица;
* Просмотреть результат работы программы во вспомогательном окне;
* Получить результат работы программы в виде файла *3D*-модели в формате *.obj*;
* Успешно завершить выполнение программы.