

# Оглавление

<b>Введение</b>	3
<b>Глава 1. ОБЗОР АНАЛОГОВ</b>	5
1.1. Критерии сравнения . . . . .	5
1.2. Мобильные приложения . . . . .	6
1.3. BQ Ciclop DIY 3D Scanner . . . . .	7
1.4. hesamh DIY 3D Scanner . . . . .	9
1.5. Вывод . . . . .	10
<b>Глава 2. ВЫБОР ФИЗИЧЕСКИХ ПАРАМЕТРОВ СКАНЕРА</b>	11
2.1. Выбор метода сканирования . . . . .	11
2.2. Выбор компоновки . . . . .	12
<b>Глава 3. ВЫВОД НЕОБХОДИМЫХ УРАВНЕНИЙ</b>	17
3.1. Конфигурации модели . . . . .	17
3.1.1. Условные обозначения . . . . .	18
3.1.2. Модель в системе координат принтера . . . . .	18
3.1.3. Модель в системе координат камеры . . . . .	22
3.1.4. Выбор модели расчётов . . . . .	26
3.2. Оценка дискретности . . . . .	26
3.3. Калибровка сканера . . . . .	29
<b>Глава 4. АЛГОРИТМ МОДУЛЯ СКАНИРОВАНИЯ</b>	32
4.1. Этапы работы алгоритма . . . . .	32
4.2. Блок обработки видео . . . . .	33
4.2.1. Предобработка . . . . .	34
4.2.2. Выделение лазера . . . . .	35
4.2.3. Привязка координат . . . . .	41

4.3. Блок поиска объектов . . . . .	42
4.4. Блок генерации gcode . . . . .	45
<b>Глава 5. ОПИСАНИЕ РАБОТЫ СИСТЕМЫ . . . . .</b>	<b>50</b>
<b>Заключение . . . . .</b>	<b>53</b>
<b>Список литературы . . . . .</b>	<b>54</b>

## Введение

В настоящее время существует множество задач, требующих бесконтактного метода измерений: ориентирование в пространстве, измерение объектов, реконструкция объектов, сбор биометрических данных, реверс-инжиниринг, а также дизайн и творчество. Подобных задач с каждым годом становится всё больше, и таким образом растёт важность 3D-сканирования и следовательно необходимость эффективных алгоритмов, решающих конкретную задачу. Кроме того, некоторые задачи требуют уникальных встроенных решений, что представляет собой как правило ещё и конструкторскую задачу.

Одним из конкретных применений такой технологии является фигурное нанесение глазури на различные кондитерские изделия. На данный момент такая операция как правило выполняется вручную, что означает высокую стоимость, длительное производство и низкую повторяемость. Автоматизация этого процесса происходит обычно на крупных предприятиях, где производят большие партии однотипных кондитерских изделий. Получается, что на рынке нет промежуточного решения которое позволяло бы выпускать средние партии украшенной индивидуальным образом кондитерской продукции.

Относительно недавно стали появляться автоматизированные комплексы для нанесения рисунков на различные кондитерские изделия, однако они всё равно ограничены конкретными видами изделий (например, плоские крекеры). Такие комплексы работают по принципу 3D-принтера или ЧПУ-станка. Они печатают в одной плоскости, не имея возможности наносить материал с учётом отклонений формы, не говоря уже об изделиях сложной формы, таких как овсяное печенье, кексы, торты и т.д., у которых форма в общем случае не только не плоская, но и обладает множеством искривлений, выпукостей и впадин. Данное обстоятельство мешает автоматизации нанесения произвольных рисунков на кондитерские изделия различных форм и размеров.

Данная работа посвящена разработке модуля 3D-сканирования для конди-

терского принтера, что позволит наносить рисунки на произвольные изделия. Модуль рассчитан на прототип системы автоматического фигурного нанесения пищевой пасты, который представляет из себя принтер закрытого типа с подвижным печатающим узлом и статичным столом, на котором располагаются кондитерские изделия.

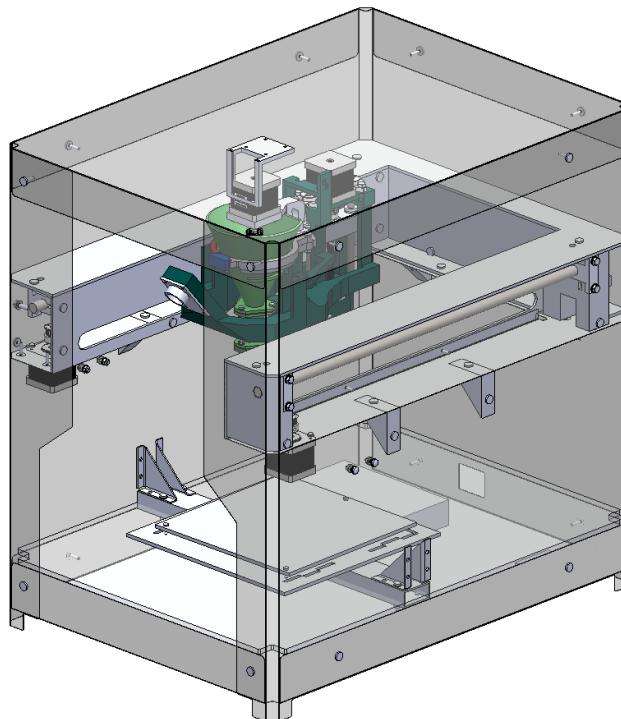


Рисунок 1 – Прототип системы автоматического фигурного нанесения пищевой пасты

Алгоритм должен решать следующие задачи:

- получение облака точек рабочей зоны
- обнаружение объектов и их положений в облаке точек
- генерация gcode произвольного рисунка с учётом облака точек

Новый модуль должен удовлетворять следующим требованиям:

- рабочая зона  $200 \times 200$  мм
- точность  $\pm 0.5$  мм
- время обработки данных не более 30 секунд

## Глава 1

# ОБЗОР АНАЛОГОВ

Как уже было отмечено ранее, 3D-сканеры находят применение во множестве областей и решают самые разные задачи. Поэтому на данный момент уже есть готовые устройства, позволяющие производить сканирование, и реализовано множество методов сканирования, каждый из которых требует разные компоненты и ресурсы.

Рассмотрим существующие аналоги таких устройств и соответствующие им методы. Ограничим выбор 3D-сканерами распространяющиеся по свободной модели, поскольку в нашем проекте важна низкая стоимость модуля.

### 1.1. Критерии сравнения

Определим критерии для сравнения устройств:

Точность — насколько результаты измерения отклоняются от реальных. Один из наиболее важных параметров сканеров. В зависимости от метода колеблется от единиц миллиметров до микрометров.

Скорость — насколько быстро производятся вычисления. Многие задачи требуют расчётов в реальном времени и этот критерий является критическим. Как правило Высокая скорость обработки сказывается на точности алгоритма или ограничивает применимость системы (универсальность), поскольку использует специфичные упрощения и аппроксимации. В зависимости от способа получения информации об объекте может измеряться в  $\frac{\text{мм}^2}{\text{с}}$ ,  $\frac{\text{кадры}}{\text{с}}$ .

Диапазон измерений — диапазон расстояний (глубины) на которой устройство способно обеспечить заданную точность в миллиметрах или метрах. Как правило определяется физическими ограничениями метода измерения и конструкцией устройства. Например размеры матрицы камеры и угол обзора задают верхний порог измерений, а для методов основанных на регистрации свето-

вых импульсов существует нижний предел измерений обусловленный высокой скоростью света.

Габариты — используемые технологии измерения оказывают существенное влияние на габариты устройства. В настоящее время существуют ручные сканеры довольно малых размеров, но также есть и стационарные, более габаритные устройства.

## 1.2. Мобильные приложения

В настоящее время 3D-технологии довольно распространены, поэтому существуют приложения на смартфон (например, Sony 3D Creator[1; 2]), которые позволяют проводить сканирование любому человеку при наличии устройства поддерживающего необходимые технологии. Такие приложения, как правило, распространяются бесплатно, что делает их самыми доступными (если не учитывать стоимость самого смартфона).

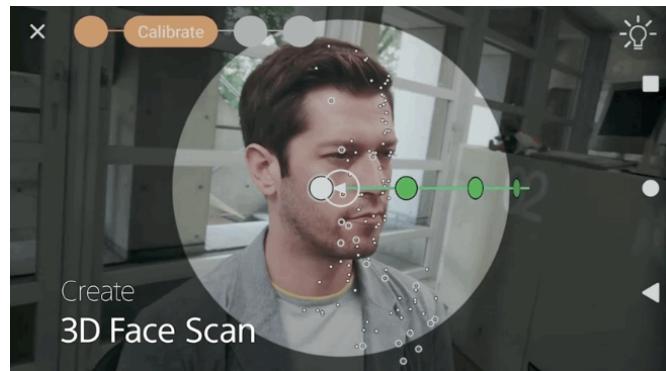


Рисунок 1.1 – Интерфейс приложения Sony 3D creator

Подобные приложения используют для измерений метод фотограмметрии. Суть данного метода заключается в том, что, имея несколько изображений одного объекта, полученных с разных точек обзора, можно сопоставить особые точки (features) этих изображений после чего восстановить модель объекта по каждому пикселию снимков[3].

Метод фотограмметрии, и стереоскопия в частности, как правило, имеют

сравнительно низкую точность, но высокую скорость сканирования. Особенно низкая точность свойственна мобильным приложениям ввиду ограниченных вычислительных ресурсов и качества используемых камер. Измерять таким методом можно объекты на расстоянии порядка метра от точки обзора. Габариты и стоимость ограничены используемым смартфоном, однако этот же метод можно использовать с несколькими фиксированными камерами.

Таблица 1.1 – Характеристики приложения Sony 3D creator

Метод	Фотограмметрия\стереозрение
Диапазон	$\approx 1$ м
Скорость	По завершению съёмки
Точность	$\approx 1$ мм
Габариты	Корпус смартфона
Стоимость	Бесплатно (стоимость смартфона)

### 1.3. BQ Ciclop DIY 3D Scanner

Это сканер с открытым исходным кодом, 3D-модели корпуса которого доступны для загрузки с целью последующей самостоятельной печати[4]. Представляет из себя поворотную платформу с закреплёнными к ней веб-камерой и двумя лазерными модулями. Данное устройство использует метод лазерной триангуляции для расчёта координат в сцене.

Суть данного метода в том, что лазер излучает на исследуемую поверхность, которую снимает камера. Камера и лазер при этом должны находиться под углом друг к другу. Таким образом проекция лазера, видимая в кадре, искажается согласно форме исследуемой поверхности. Зная угол и расстояние между камерой и лазером, а так же фокусное расстояние, можно рассчитать координаты засвеченных точек используя отклонение проекции в кадре, так как эти величины формируют подобные треугольники с одной неизвестной величи-



Рисунок 1.2 – Сканер BQ Ciclop

ной – координатой точки.

Данный метод, как правило, обладает высокой точностью. В дорогих сканерах (например, Faro ScanArm[5]) точность может быть порядка микрометров. Но сканирование этим методом может занять некоторое время, т.к. возможно исследовать только один «профиль» сцены за кадр. Таким образом необходимо провести лазером от одного конца сцены до другого, при этом плотность облака напрямую зависит от частоты кадров камеры и скорости движения вдоль сцены.

Недостатками этого аналога являются его конструкция с поворотным столиком, которую невозможно внедрить в имеющийся прототип. Также нельзя использовать его программное обеспечение, т.к. оно написано с учётом конструкции сканера.

Таблица 1.2 – Характеристики сканера BQ Ciclop 3D Scanner

Метод	Лазерная триангуляция
Диапазон	$\varnothing 250 \times 205$ мм
Скорость	2-8 минут на оборот
Точность	$\approx 0.5$ мм
Габариты	$500 \times 300 \times 230$ мм
Стоимость	6 000 р.

## 1.4. hesamh DIY 3D Scanner

Данный аналог это открытый проект за авторством пользователя hesamh, основанный на методе структурированного света[6]. Сканер собран из старого проектора и двух веб-камер, закреплённых на деревянном основании. Для этого проекта в открытом доступе находится инструкция для самостоятельной сборки и он использует свободное программное обеспечение 3DUNDERWORLD для обработки данных.



Рисунок 1.3 – Сканер hesamh

Метод структурированного света похож на метод лазерной триангуляции (триангуляция с проекцией линией является частным случаем), но использует специальный рисунок (паттерн), как правило чередующихся чёрных и белых полос. Этот рисунок также проецируется на объект, с помощью проектора, и искажения рисунка соответствуют форме объекта. Данный метод обладает, в общем случае, такой же точностью как метод лазерной триангуляции, но позволяет получить больше информации из одного кадра. Однако при этом необходим более сложный алгоритм для обработки данных с камеры. Другим существенным недостатком этого метода является необходимость использовать проектор, что значительно увеличивает его габариты и стоимость по сравнению с другими методами.

Таблица 1.3 – Характеристики сканера от hesamh

Метод	Структурированный свет
Диапазон	до 2 м от камеры
Скорость	-
Точность	$\approx 0.5$ мм
Габариты	$1000 \times 500 \times 300$ мм
Стоимость	$\sim 10000$ р.

## 1.5. Вывод

В результате обзора существующих решений стала очевидной невозможность использования готовых устройств ввиду ограниченного бюджета проекта и требований компактности. Таким образом необходимо разработать собственный модуль совместимый с прототипом. Для этого требуется с учётом особенностей принтера выбрать подходящий метод сканирования, компоновку модуля и его составляющие части.

## Глава 2

# ВЫБОР ФИЗИЧЕСКИХ ПАРАМЕТРОВ СКАНЕРА

### 2.1. Выбор метода сканирования

Для выбора метода сканирования необходимо определить условия работы и ограничения накладываемые существующей конструкцией прототипа на модуль сканирования и сопоставить их с требованиями и особенностями методов измерения, а также учесть требования технического задания.

Поскольку все методы оптические, они предъявляют требования к освещённости окружения. Освещение рабочего поля должно быть равномерным, не слишком ярким и не должно вызывать бликов. При этом сканируемые поверхности не должны быть зеркальными или поглощающими свет.

Параметры конструкции прототипа:

- габариты корпуса  $400 \times 550 \times 560$  мм
- размеры рабочей области  $200 \times 200$  мм
- контролируемое освещение
- закрытый корпус

Видно, что свободное пространство в принтере ограничено из-за наличия подвижного печатающего узла, габариты которого выходят за пределы рабочей зоны. Таким образом становится невозможна использование крупных компонентов в разрабатываемой системе сканирования.

Поэтому метод структурированного света в данном случае не подходит, так как требует использования проектора, который имеет большие размеры и

высокую стоимость. Существуют и миниатюрные проекторы, однако их стоимость ещё выше. Алгоритмы используемые в методе структурированного света являются сложными, что увеличивает требования к вычислительными мощностям прототипа и как следствие его стоимость.

Лазерная триангуляция накладывает дополнительные ограничения на возможные сканируемые поверхности, так как использует видимое излучение для работы. Это значит, что сканирование предметов такого-же цвета, что и свет излучаемый лазером, будет сильно затруднено. Такая ситуация может возникнуть, если изделие, например торт, будет обладать цветным покрытием. Однако это можно преодолеть если использовать сканирование в темноте. Это позволит сканировать такие объекты, тем не менее качество таких сканов будет хуже из-за явления самозасветки лазера. Так же, используя этот метод, модуль целиком или только лазер, необходимо перемещать вдоль сцены, чтобы захватить все профили исследуемой поверхности. Это снизит скорость сканирования в сравнении с остальными методами. Однако для реализации этого метода достаточно одной камеры и лазерного модуля, которые можно расположить компактно. Также это снижает стоимость прототипа, что важно в данном проекте.

Стереозрение в данной работе не рассматривается, так как этот метод был опробован ранее и не показал удовлетворительных результатов. В данном методе теряется большое количество данных сцены и точность оставляет желать лучшего. При этом для этого необходимы две камеры, что так же проблематично встроить в существующую систему[7].

## 2.2. Выбор компоновки

Общий план модуля сканирования представлен на рисунке 2.1 (подвижный узел не показан). Он демонстрирует относительное расположение элементов в пространстве. В данной компоновке лазер направлен перпендикулярно столу (параллельно оси  $Z$ ), линия проекции лазера параллельна оси  $Y$  стола, а ка-

мера под углом к вертикали не имея поворота относительно остальных осей. Движение модуля происходит параллельно оси  $X$  системы координат принтера в сторону уменьшения координаты.

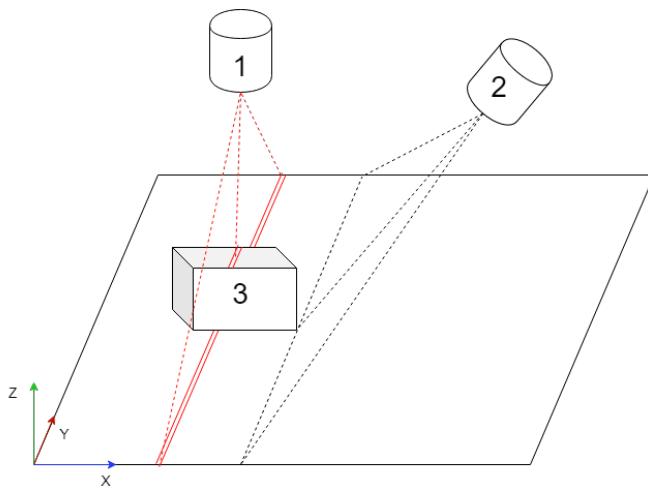


Рисунок 2.1 – Общий план модуля

Для обеспечения лучших показателей точности необходимо выбрать расстояние и угол между оптической осью камеры и осью лазера, а так же расстояние от камеры до стола. Этот этап важен, так как от этих параметров зависит разрешающая способность сканера и видимая рабочая зона. Чем ближе камера к столу, тем меньше дискретизация значений координат, но одновременно меньше видимая область. Угол наклона камеры также влияет на дискретизацию и видимую область сканера. С помощью расчётов, приведённых в разделе 3.2 были установлены следующие величины этих размеров:

Таблица 2.1 – Основные размеры модуля сканирования

Высота над столом	$140 \pm 10$ мм
Расстояние от камеры до лазера	$140 \pm 10$ мм
Угол между осями камеры и лазера	$45 \pm 15^\circ$

Данные величины позволяют достичь необходимой по ТЗ точности в  $\pm 0.5$  мм и ширины обзора 200 мм. Для модуля используется камера Microsoft Lifecam Cinema с максимальным разрешением  $1280 \times 720$  и максимальной частотой 30

кадров в секунду. Использование данной камеры позволяет упростить процесс калибровки, так как она обладает встроенной ректификацией изображения. Лазерный модуль S-2S с толщиной проецируемой линии в 3 мм. Точность измерений находится в обратной зависимости от толщины линии лазера.

Камера устанавливается в специальный держатель, в котором предусмотрена калибровка поворота камеры для более точного направления на интересующую зону и надёжного закрепления. Реализация крепления и установки модуля в систему рассматривается в работе[8].

К конструкции модуля предъявляются следующие требования:

- минимизация влияния вибрации на положение камеры
- размеры в пределах указанных в таблице 2.1
- допускаются посторонние объекты в верхней половине кадра занимающие не более четверти высоты кадра
- возможность калибровки поворота камеры
- возможность калибровки поворота лазера

Итоговый вид модуля в составе прототипа изображён на рисунке 2.3. Крепление камеры изображено на рисунке 2.4.

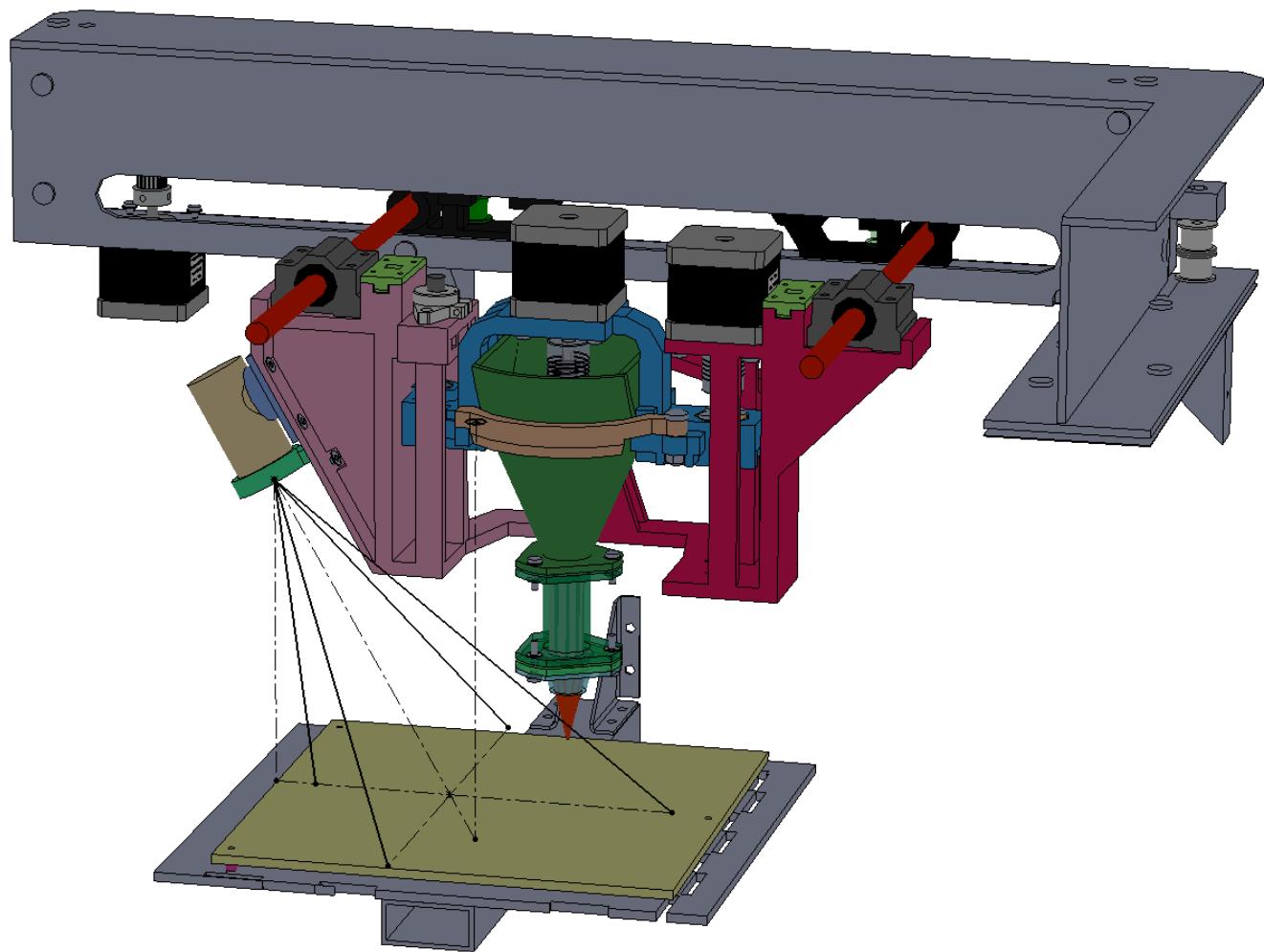


Рисунок 2.2 – Общий вид в принтере

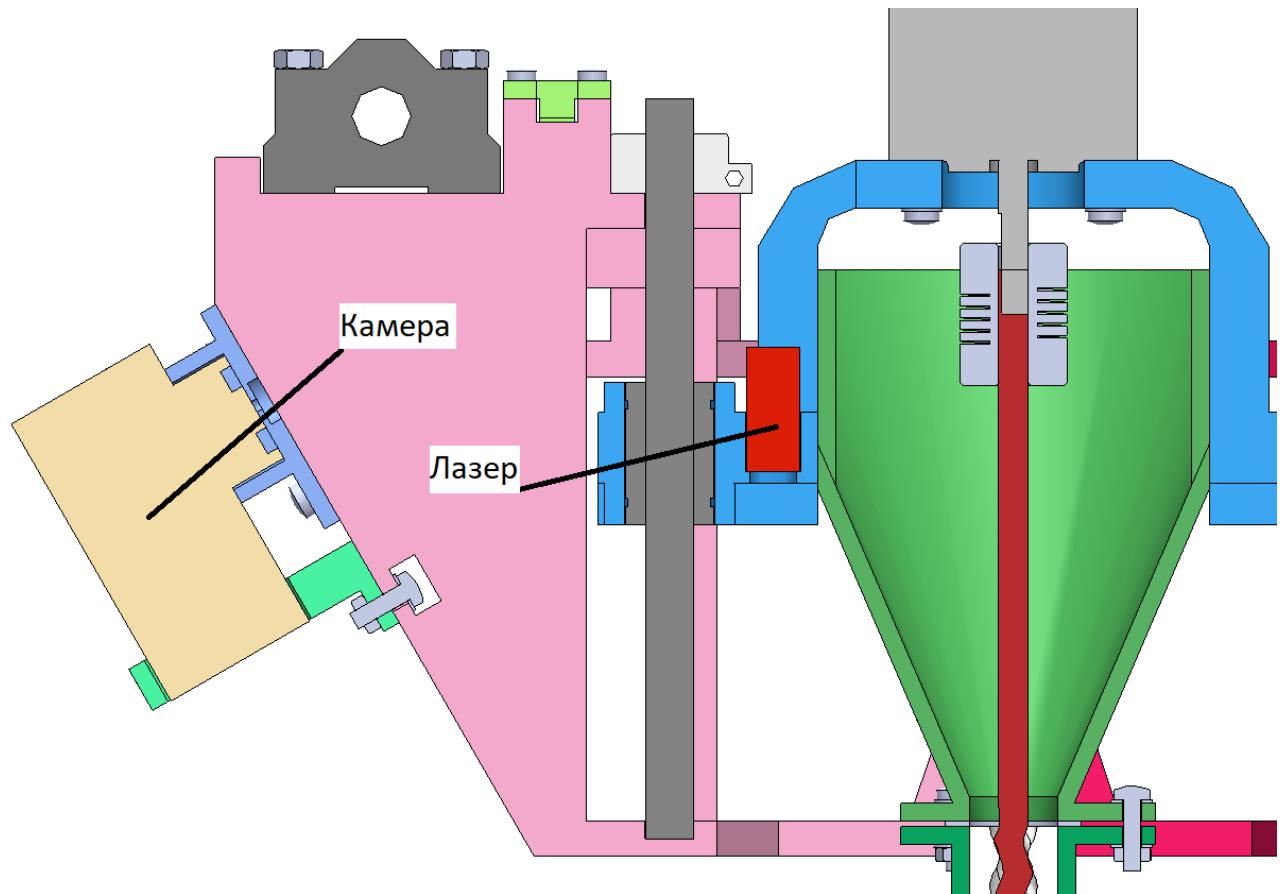


Рисунок 2.3 – Финальная компоновка модуля сканирования

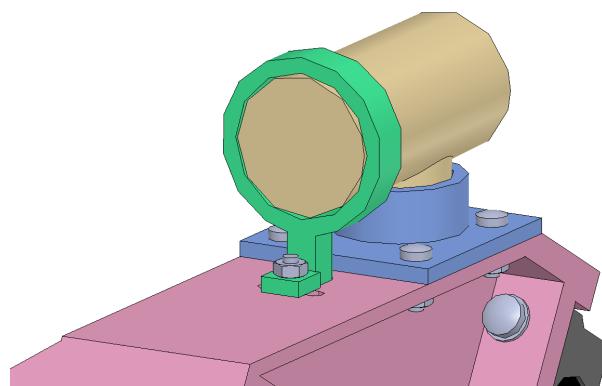


Рисунок 2.4 – Крепление камеры

## Глава 3

# ВЫВОД НЕОБХОДИМЫХ УРАВНЕНИЙ

### 3.1. Конфигурации модели

Для написания алгоритма сканирования в первую очередь необходимо составить математическую модель сканера, то есть вывести соответствующие уравнения, по которым будут рассчитываться координаты точек в сцене. При этом возможно несколько вариантов для вывода необходимых уравнений.

Для расчёта применяется модель перспективной проекции. Начало координат в данной модели располагается в центре проекций – точке где пересекаются линии проекций точек пространства.

Оси координат направлены следующим образом –  $Z$  от камеры перпендикулярно плоскости изображения,  $Y$  вниз параллельно короткой стороне кадра,  $X$  вправо параллельно длинной стороне. Таким образом получаем правую систему координат.

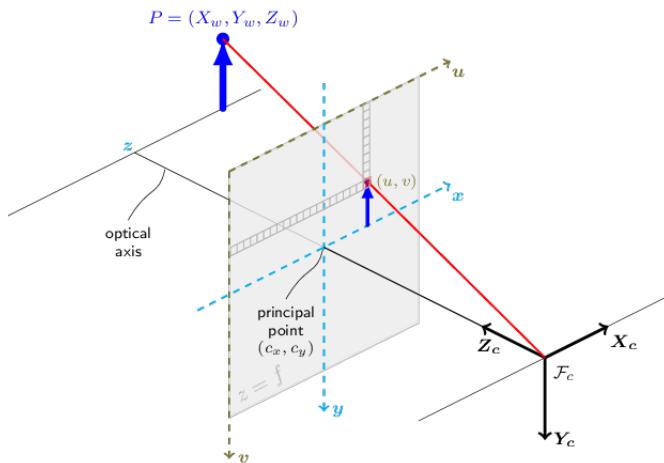


Рисунок 3.1 – Модель камеры; principal point – центр изображения, проекция начала координат на плоскость изображения

Экран в данной модели может располагаться как между объектом и центром проекции (как показано на рисунке 3.1), так и за центром проекции. В

последнем случае изображение оказывается перевёрнутым.

### 3.1.1. Условные обозначения

$f$  — фокусное расстояние камеры в пикселях

$u, v$  — горизонтальная и вертикальная координаты соответственно в плоскости изображения (пиксели)

$u_0, v_0$  — координата центра изображения

$(x, y, z)$  — координаты в СК прототипа

$({}^k x, {}^k y, {}^k z)$  — координаты в СК камеры

### 3.1.2. Модель в системе координат принтера

Первый расчёт производится в системе координат прототипа. Он основывается на реальных размерах сборки и отражает реальную конструкцию модуля сканирования. В данном варианте расчёта предполагается, что стол — рабочая поверхность — находится в плоскости  $XY$ , лазер излучает перпендикулярно столу и камера имеет поворот только вокруг оси  $Y$ . При этом считается, что лазер не имеет поворота вокруг своей оси.

Этот метод основан на работе [9], но учитывает смещение координат  $(x, y)$  с изменением высоты.

Необходимо найти уравнения описывающие зависимость координат  $(x_P, y_P, z_P)$  точки  $P$  от координат проекции этой точки на плоскость изображения  $(u_P, v_P)$ . Из  $\triangle PPM$  видно, что

$$z_P = \frac{P'M}{\cos \angle PPM} \quad (3.1)$$

Далее из  $\triangle CP'M$  очевидно

$$P'M = CP' \sin (\beta - \beta_0) \quad (3.2)$$

где  $\beta$  — произвольный угол падения луча лазера на матрицу,

$\beta_0$  — угол падения луча лазера отражённого от поверхности стола на матрицу.

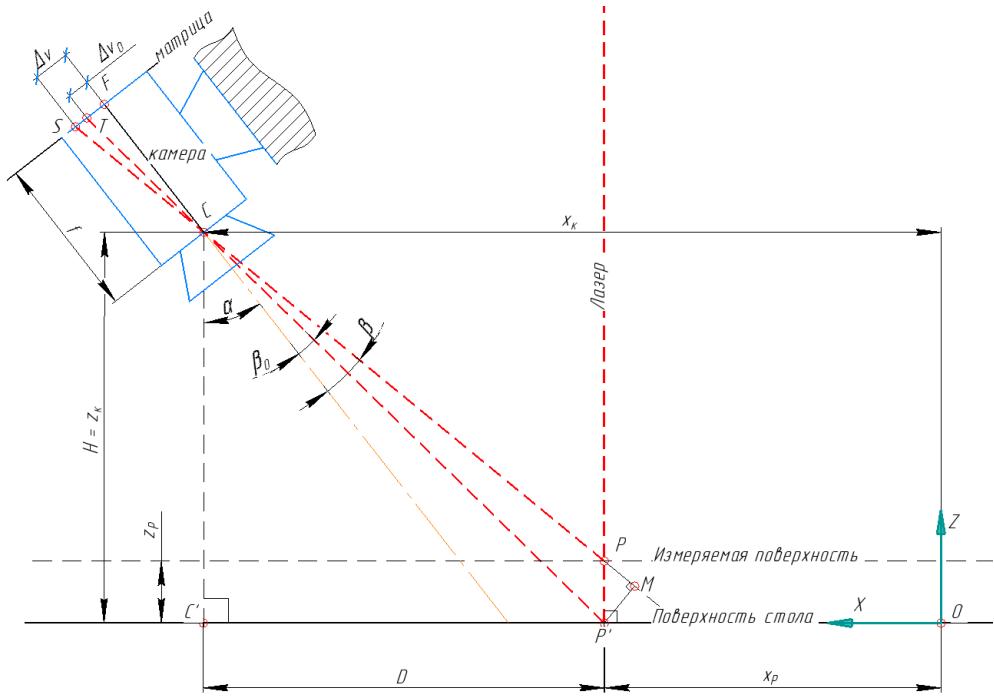


Рисунок 3.2 – Расчёт в системе координат прототипа

Рассмотрев  $\triangle CP'C'$  можно увидеть, что

$$CP' = \frac{H}{\cos(\alpha + \beta_0)} \quad (3.3)$$

Подставляя последовательно полученные равенства друг в друга получаем следующее уравнение

$$z_P = H \frac{\sin(\beta - \beta_0)}{\cos(\alpha + \beta_0) \cos \angle PP'M} \quad (3.4)$$

нетрудно показать, что

$$\cos \angle PP'M = \sin(\alpha + \beta) \quad (3.5)$$

В итоге приходим к уравнению координаты  $z$  точки  $P$ , выраженной через углы падения луча лазера на матрицу камеры.

$$z_P = H \frac{\sin(\beta - \beta_0)}{\cos(\alpha + \beta_0) \sin(\alpha + \beta)} \quad (3.6)$$

Расстояние точки  $P$  от камеры при данном методе расчёта константа на всём протяжении линии лазера в виду конструкции и равно расстоянию между камерой и лазером  $C'P' = D$

$$D = H \tan(\alpha + \beta_0) \quad (3.7)$$

тогда координата  $x_P$  выражается следующим соотношением

$$x_P = x_k - D = x_k - H \operatorname{tg}(\alpha + \beta_0) \quad (3.8)$$

Для вывода уравнения координаты  $y_P$  необходимо рассмотреть модуль в проекции на плоскость  $ZY$

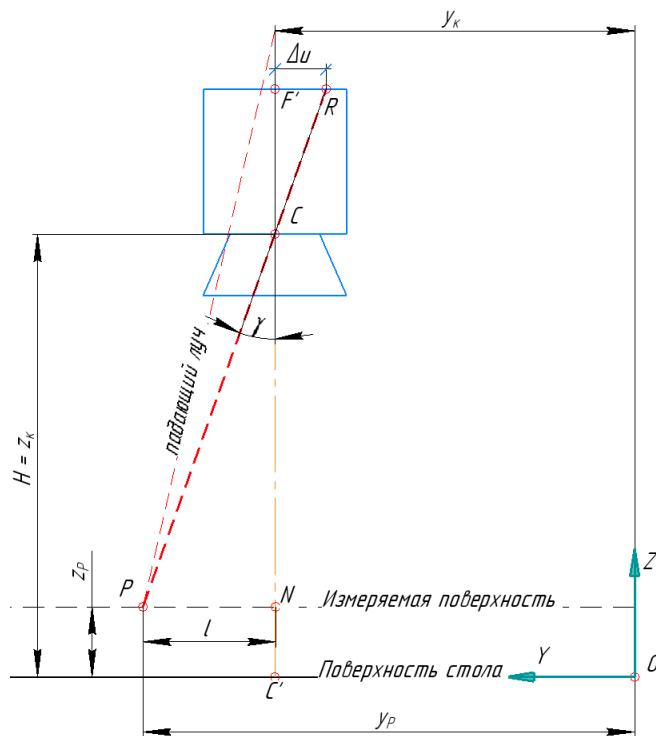


Рисунок 3.3 – Проекция на плоскость  $ZY$

При выводе важно помнить, что данный вид – проекция. Из чертежа очевидно, что

$$y_P = y_k + l \quad (3.9)$$

Из  $\triangle CPN$  легко вывести  $l$

$$l = CN \operatorname{tg} \gamma \quad (3.10)$$

где  $\gamma$  – угол падения луча лазера на матрицу в данной проекции. Поскольку  $CN = H - z_P$  можно записать

$$y_P = y_k + (H - z_P) \operatorname{tg} \gamma \quad (3.11)$$

Для вывода углов падения  $\beta$  и  $\beta_0$  рассмотрим  $\triangle CSF$  и  $\triangle CTF$ . Очевидно, что

$$\begin{aligned}\operatorname{tg} \beta &= \frac{\Delta v}{f} \\ \operatorname{tg} \beta_0 &= \frac{\Delta v_0}{f}\end{aligned}\tag{3.12}$$

где  $\Delta v = v_P - v_0$  – расстояние в пикселях от точки падения луча лазера на матрицу до центра изображения;

$\Delta v_0 = v_{P0} - v_0$  – аналогичное расстояние для луча лазера, отражённого от поверхности стола.

Рассмотрев  $\triangle CF'R$  получим значение угла падения в проекции  $ZY$

$$\operatorname{tg} \gamma = \frac{\Delta u}{CF'} = \frac{\Delta u}{f \cos \alpha}\tag{3.13}$$

где  $CF' = f \cos \alpha$  – проекция фокусного расстояния на плоскость  $ZY$ . Из полученных равенств легко найти значения углов, поскольку величины  $\Delta u$  и  $\Delta v$  известны из изображения с камеры.

Таким образом приходим к следующим уравнениям

$$\begin{cases} x_P = x_k - H \operatorname{tg}(\alpha + \beta_0) \\ y_P = y_k + (H - z_P) \frac{\Delta u}{f \cos \alpha} \\ z_P = H \frac{\sin(\beta - \beta_0)}{\cos(\alpha + \beta_0) \sin(\alpha + \beta)} \end{cases}\tag{3.14}$$

Преимущества этой модели:

- $H$  можно измерять напрямую
- рассчитанные значения координат соответствуют системе прототипа
- отражает реальную конструкцию сканера
- координата  $x$  – константа относительно камеры

Недостатки:

- много тригонометрических преобразований

- при изменении высоты камеры необходимы новые замеры
- большое количество допущений, как следствие много возможностей для ошибок

### **3.1.3. Модель в системе координат камеры**

Второй расчёт производится в системе координат связанной с камерой. Он основывается на теоретических величинах и происходит в два этапа. Первый – расчёт координат относительно камеры, второй – преобразование полученных координат в систему прототипа. В этом расчёте принято допущение, что лазер не имеет поворота вокруг своей оси, все остальные компоненты могут располагаться произвольно.

Для преобразования между двумя системами координат необходимо знать матрицу поворота системы камеры относительно системы прототипа. С помощью пакета opencv можно полностью определить положение камеры в пространстве используя шахматный паттерн[10].

Введём следующие определения:

*Рабочая плоскость* — плоскость перпендикулярная оптической оси камеры и проходящая через точку пересечения оптической оси камеры и луча лазера

*Измеряемая плоскость* — плоскость, до которой измеряется расстояние

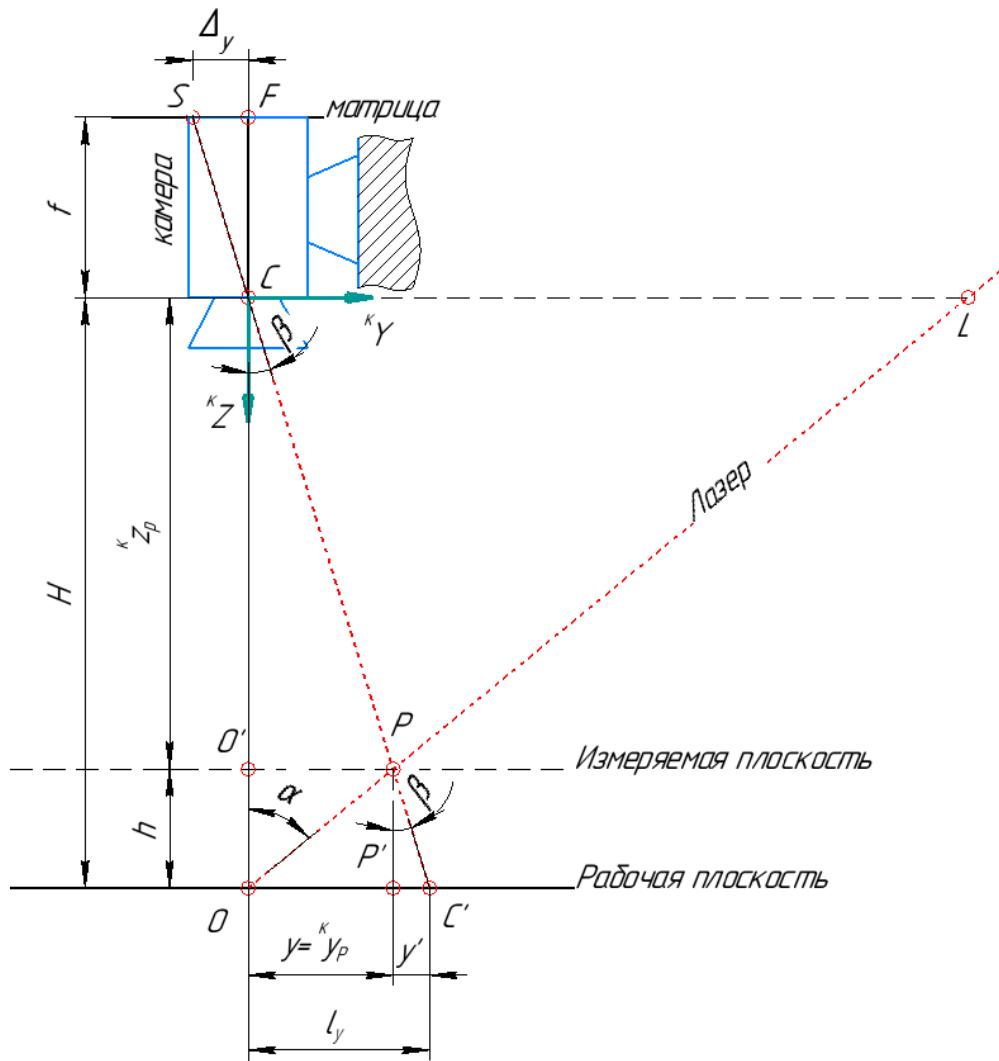


Рисунок 3.4 – Расчёт в системе координат камеры;  $O$  – точка пересечения оптической оси камеры и луча лазера (плоскости свечения лазера);  $C$  – оптический центр камеры, начало координат СК камеры;  $P$  – точка пересечения луча лазера и измеряемой плоскости;  $P'$  – проекция точки  $P$  на рабочую плоскость;  $S$  – точка падения луча лазера на матрицу;

На рисунке 3.4 обозначены следующие размеры:

$H$  – рабочая высота, расстояние от оптического центра камеры до рабочей плоскости

$\alpha$  – угол между осями камеры и лазера

$l_x, l_y$  – кажущиеся координаты измеряемой точки

$x, y$  – реальные координаты измеряемой точки

$x', y'$  – разность кажущейся и реальной координаты измеряемой точки

$h$  – высота измеряемой плоскости над рабочей плоскостью

Необходимо найти уравнения описывающие зависимость координат  $({}^kx_P, {}^ky_P, {}^kz_P)$  точки  $P$  в системе координат камеры от координат проекции этой точки на плоскость изображения  $(u_P, v_P)$ . Из чертежа очевидно, что

$${}^kz_P = H - h, \quad (3.15)$$

где  $h$  – высота измеряемой плоскости над рабочей. Рассмотрим треугольники  $\triangle OPC$ ,  $\triangle OPP'$ ,  $\triangle PCP'$ . Видно что

$$l_y = y + y', \quad (3.16)$$

где  $l_y$  – кажущаяся координата точки  $P$ ;

$y = {}^ky_P$  – реальная координата точки  $P$ ;

$y'$  – разность реальной и кажущейся координаты. Так как  $PP' = h$  можно записать следующее:

$$\begin{aligned} y &= h \operatorname{tg} \alpha \\ y' &= h \operatorname{tg} \beta \end{aligned} \quad (3.17)$$

где  $\beta$  – угол при вершине  $P$  в треугольнике  $\triangle PCP'$ ,  $\alpha$  – угол между оптической осью камеры и осью лазера. Также из треугольника  $\triangle SOC'$  ясно

$$l = H \operatorname{tg} \beta \quad (3.18)$$

следовательно

$$H \operatorname{tg} \beta = h (\operatorname{tg} \alpha + \operatorname{tg} \beta) \quad (3.19)$$

Зная, что  $h = H - {}^kz_P$  получаем

$${}^kz_P = H \frac{\operatorname{tg} \alpha}{\operatorname{tg} \alpha + \operatorname{tg} \beta} \quad (3.20)$$

Из треугольника  $\triangle CFS$  находим

$$\operatorname{tg} \beta = \frac{\Delta v}{f}, \quad (3.21)$$

где  $\Delta v = v_P - v_0$ ,  $f$  – фокусное расстояние камеры. В итоге получаем

$${}^kz_P = H \frac{\operatorname{tg} \alpha}{\operatorname{tg} \alpha + \frac{\Delta v}{f}} \quad (3.22)$$

Далее из  $\triangle CO'P$  очевидно, что

$${}^k y_P = {}^k z_P \operatorname{tg} \beta = H \frac{\operatorname{tg} \alpha \frac{\Delta v}{f}}{\operatorname{tg} \alpha + \frac{\Delta v}{f}} \quad (3.23)$$

аналогично для  ${}^k x_P$

$${}^k x_P = {}^k z_P \operatorname{tg} \gamma = H \frac{\operatorname{tg} \alpha \frac{\Delta u}{f}}{\operatorname{tg} \alpha + \frac{\Delta u}{f}} \quad (3.24)$$

где  $\operatorname{tg} \gamma = \frac{\Delta u}{f}$  – угол падения луча лазера на матрицу в плоскости  $XZ$ ,  $\Delta u = u_P - u_0$ .

В результате для системы координат камеры получаем:

$$\left\{ \begin{array}{l} {}^k x_P = H \frac{\operatorname{tg} \alpha \frac{\Delta u}{f}}{\operatorname{tg} \alpha + \frac{\Delta v}{f}} \\ {}^k y_P = H \frac{\operatorname{tg} \alpha \frac{\Delta v}{f}}{\operatorname{tg} \alpha + \frac{\Delta v}{f}} \\ {}^k z_P = H \frac{\operatorname{tg} \alpha}{\operatorname{tg} \alpha + \frac{\Delta v}{f}} \end{array} \right. \quad (3.25)$$

Для перевода в систему прототипа:

$$\begin{pmatrix} x_P \\ y_P \\ z_P \end{pmatrix} = R \begin{pmatrix} {}^k x_P \\ {}^k y_P \\ {}^k z_P \end{pmatrix} + \begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix} \quad (3.26)$$

Преимущества модели:

- отсутствие тригонометрии в расчётах координат
- параметры  $H$  и  $\alpha$  фиксированы
- меньшее количество допущений

Недостатки модели:

- сложно прямо измерить  $H$  и  $\alpha$ , необходим теоретический расчёт
- необходима матрица поворота камеры

### 3.1.4. Выбор модели расчётов

Сравнивая расчёты, видим, что недостатки второго менее серьёзны чем первого, а также что второй расчёт не имеет недостатков первого. В частности расчёт в системе координат камеры упрощает эксплуатацию модуля, т.к. отсутствует необходимость повторных измерений параметров при изменении положения внешних, по отношению к модулю, элементов (например, перемещение стола). Также меньшее количество допущений для второй модели позволяет добиться более точных результатов не имеющих пространственных искажений (например, растяжение сцены), которые могут возникать в следствии дополнительного неучтённого наклона камеры. Таким образом более выгодным вариантом расчёта является вторая модель.

Использование этой модели требует дополнительных предварительных расчётов, для использования, а именно – расчёт положения камеры относительно системы координат принтера и расчёт параметров модуля  $H$  и  $\alpha$ . Расчёт положения камеры – матрицы поворота  $R$  и координат  $(x_k, y_k, z_k)$  – можно произвести средствами библиотеки OpenCV используя специальный шахматный паттерн (подробнее в главе 4). Для расчёта параметров модуля необходимо вывести соответствующий расчёт на основе уравнений 3.25 3.26, что делается в разделе 3.3.

## 3.2. Оценка дискретности

Для удовлетворения характеристик, заданных ТЗ необходимо выбрать подходящие значения параметров  $H$  и  $\alpha$ . Эти параметры определяют разрешающую способность сканера и ширину обзора камеры. Разрешающая способность камеры – разность определяемых высот между соседними пикселями.

Для простоты примем, что рабочая плоскость совпадает со столом и лазер излучает перпендикулярно столу. Таким образом  $\alpha$  становится углом наклона камеры от вертикали. Для данной оценки и последующего применения резуль-

татов примем  $H$  равной высоте камеры над столом. Также, поскольку основным изделием является печенье, нас в первую очередь интересует диапазон высот объектов 5-20 мм.

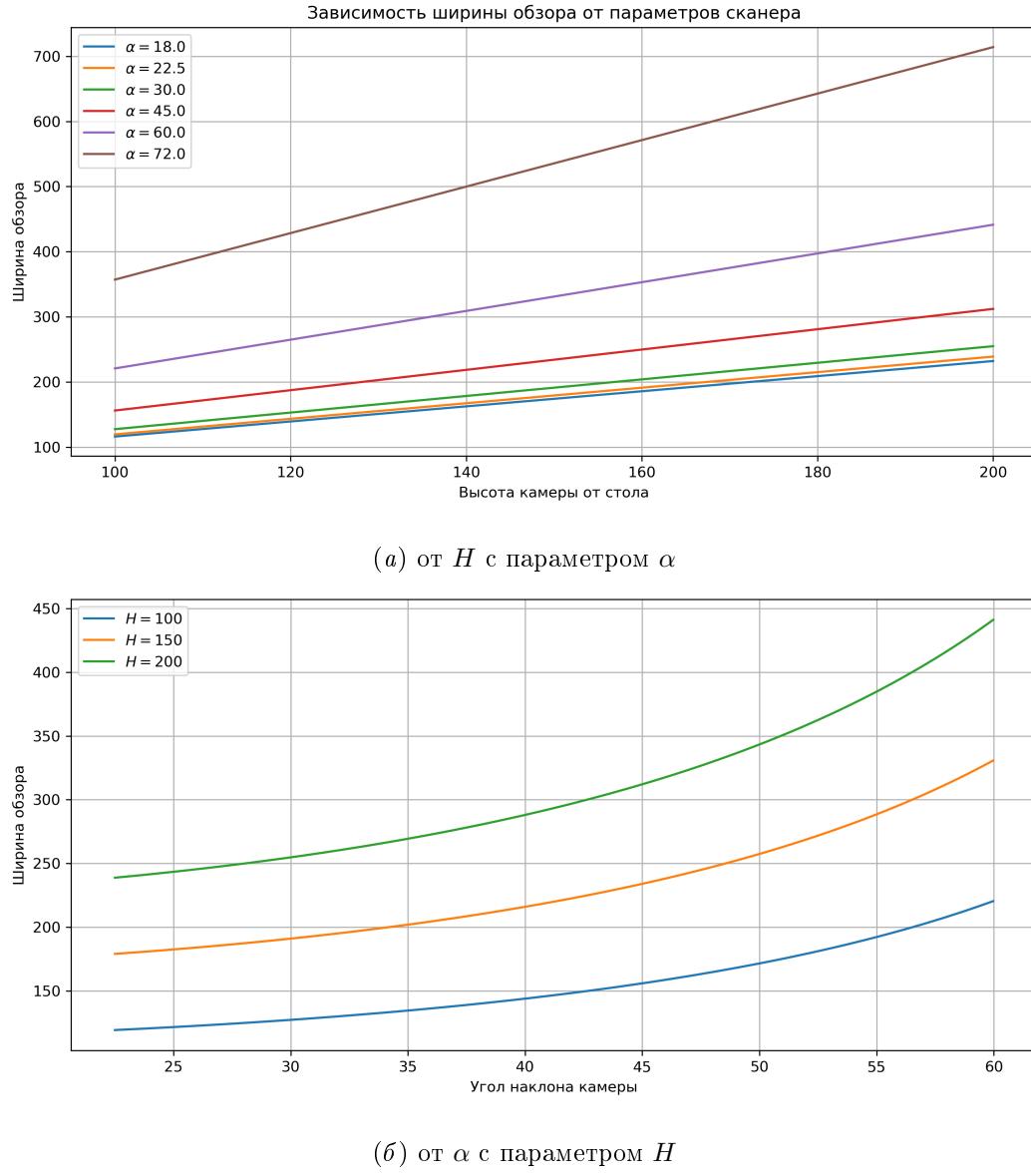


Рисунок 3.5 – Зависимость ширины обзора камеры от параметров сканера

Сначала рассмотрим рисунок 3.5 на котором изображена зависимость ширины обзора при разных параметрах сканера. На графике линии показывают ширину обзора по центру кадра, так как предполагается, что рабочая плоскость располагается в плоскости стола. Видно, что чем больше высота над столом  $H$  и угол от вертикали  $\alpha$ , тем больше ширина обзора. Необходимая ширина обзора по ТЗ равна 200 мм. Этот рисунок поможет выбрать оптимальное значение при

исследовании графиков изображённых на следующем рисунке 3.6.

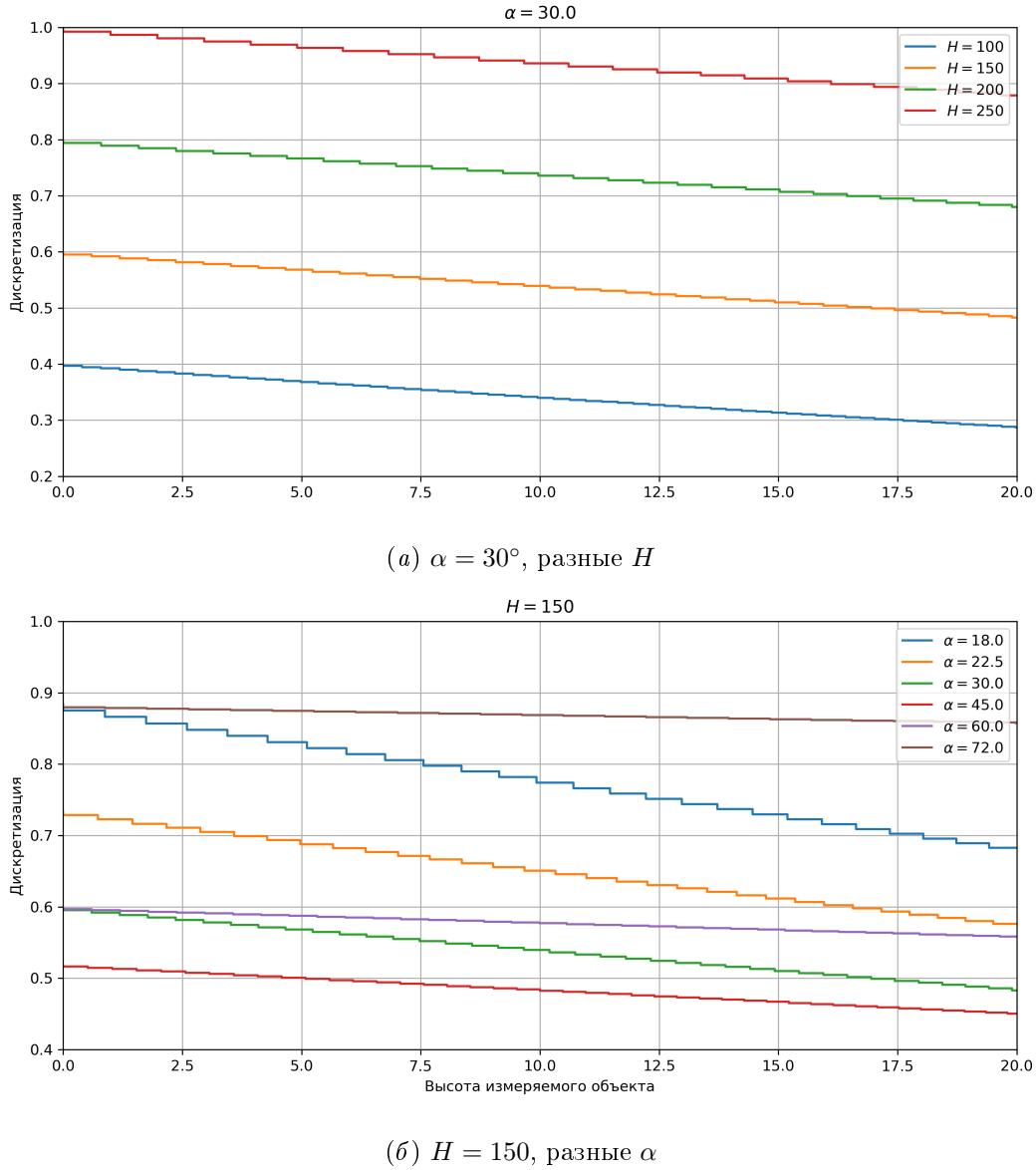


Рисунок 3.6 – Зависимость дискретизации от параметров сканера

По графикам на рисунке 3.6 видно, что дискретизация растёт с увеличением высоты камеры над столом, что логично т.к. при этом на один пиксель приходится большее расстояние. От угла же дискретизация зависит иначе - чем дальше угол от  $45^\circ$ , тем она больше. Требуемая точность системы  $\pm 0.5$  мм, таким образом дискретность должна быть не больше этого значения. Видно, что наиболее выгодные значения параметров лежат в интервалах  $130 \text{ мм} \leq H \leq 150 \text{ мм}$ ;  $30^\circ \leq \alpha \leq 60^\circ$ . При параметрах из данного диапазона одновременно достигается заданная точность и ширина обзора. При этом здесь рассматривается

случай, когда лазер на изображении находится с точностью до пикселя, однако итоговый модуль использует алгоритм нахождения лазера с субпиксельной точностью, что ещё больше снижает дискретизацию и позволяет компенсировать ошибки возникающие при сборке модуля или невозможности использовать идеальные параметры.

### 3.3. Калибровка сканера

Для калибровки сканера – расчёта параметров  $H$  и  $\alpha$  – необходимо вывести уравнения, в которых используя известные высоты эталонов можно рассчитать искомые величины. Для этого воспользуемся уравнением координаты  ${}^k z_P$  из 3.25

$${}^k z_P = H \frac{\operatorname{tg} \alpha}{\operatorname{tg} \alpha + \frac{\Delta v}{f}} \quad (3.27)$$

Имея два таких уравнения для разных высот  ${}^k z_i$  и  ${}^k z_k$  и соответствующих им разностей пикселей  $\Delta v_i$  и  $\Delta v_k$  получаем следующую систему уравнений

$$\begin{cases} {}^k z_i = H \frac{\operatorname{tg} \alpha}{\operatorname{tg} \alpha + \frac{\Delta v_i}{f}} \\ {}^k z_k = H \frac{\operatorname{tg} \alpha}{\operatorname{tg} \alpha + \frac{\Delta v_k}{f}} \end{cases} \quad (3.28)$$

разделив  ${}^k z_i$  на  ${}^k z_k$  получаем уравнение с одной неизвестной  $\operatorname{tg} \alpha$

$$\frac{{}^k z_i}{{}^k z_k} = \frac{\operatorname{tg} \alpha + \frac{\Delta v_k}{f}}{\operatorname{tg} \alpha + \frac{\Delta v_i}{f}} \quad (3.29)$$

проделав небольшие преобразования приходим к окончательному уравнению для  $\operatorname{tg} \alpha$

$$\operatorname{tg} \alpha = \frac{{}^k z_k \frac{\Delta v_k}{f} - {}^k z_i \frac{\Delta v_i}{f}}{{}^k z_i - {}^k z_k} \quad (3.30)$$

Конкретное значение угла в данном случае неважно поскольку в расчётах используется именно  $\operatorname{tg} \alpha$ , поэтому можно оставить так. Подставляя полученное

равенство в любое уравнение в системе после преобразований получаем уравнение для  $H$

$$H = {}^k z_i {}^k z_k \frac{\Delta v_k - \Delta v_i}{{}^k z_k \Delta v_k - {}^k z_i \Delta v_i} \quad (3.31)$$

Таким образом получаем систему уравнений, позволяющую рассчитать параметры сканера зная координаты двух точек  ${}^k z_i$  и  ${}^k z_k$  в системе камеры и соответствующих им разностей координат в плоскости изображения.

$$\begin{cases} \tan \alpha = \frac{{}^k z_k \frac{\Delta v_k}{f} - {}^k z_i \frac{\Delta v_i}{f}}{{}^k z_i - {}^k z_k} \\ H = {}^k z_i {}^k z_k \frac{\Delta v_k - \Delta v_i}{{}^k z_k \Delta v_k - {}^k z_i \Delta v_i} \end{cases} \quad (3.32)$$

Или можно переписать данную систему через углы падения луча лазера на матрицу  $\beta$  и получить следующую систему

$$\begin{cases} \tan \alpha = \frac{{}^k z_k \tan \beta_k - {}^k z_i \tan \beta_i}{{}^k z_i - {}^k z_k} \\ H = {}^k z_i {}^k z_k \frac{\tan \beta_k - \tan \beta_i}{{}^k z_k \tan \beta_k - {}^k z_i \tan \beta_i} \end{cases} \quad (3.33)$$

Однако полученная система уравнений зависит от координат в системе камеры, которые сложно измерить на практике. Необходимо привести уравнения к использованию координат из системы принтера. Данные координаты гораздо легче получить, т.к. возможно произвести прямые измерения этих величин. Для перевода координат из системы камеры в систему принтера воспользуемся уравнением 3.26. Согласно данному уравнению координата  $z_P$  точки  $P$  в системе принтера выражается следующим образом

$$z_P = r_{31} {}^k x_P + r_{32} {}^k y_P + r_{33} {}^k z_P + z_k \quad (3.34)$$

подставляя сюда уравнения 3.24 и 3.23 получаем

$$z_P = r_{31} {}^k z_P \tan \gamma + r_{32} {}^k z_P \tan \beta + r_{33} {}^k z_P + z_k \quad (3.35)$$

соответственно  ${}^k z_P$  равняется

$${}^k z_P = \frac{z_p - z_k}{r_{31} \tan \gamma + r_{32} \tan \beta + r_{33}} \quad (3.36)$$

Таким образом для калибровки необходимо выполнить следующее

1. Измерить координату  $z$  минимум пары эталонов в системе принтера
2. Для каждого эталона определить  $\Delta v$  или  $\operatorname{tg} \beta$
3. Перевести координаты в систему камеры подставив в уравнение [3.36](#)
4. Подставить полученные координаты и соответствующие углы в уравнение [3.32](#) или [3.33](#)

Для лучшей точности рекомендуется выбирать пары эталонов с наибольшей разницей в высоте, а также проводить множество расчётов с различными парами эталонов для усреднения результатов и минимизации ошибки измерений.

## Глава 4

# АЛГОРИТМ МОДУЛЯ СКАНИРОВАНИЯ

### 4.1. Этапы работы алгоритма

Задача алгоритма заключается в преобразовании входных данных – видео процесса сканирования и dxf файла с заданным контуром – в gcode файл. Gcode файл это последовательность инструкций для принтера написанных особым образом. Файл формата dxf это векторное изображение некоторого произвольного рисунка, которое необходимо нанести на изделия в сцене. В целом алгоритм модуля состоит из четырёх операций:

1. Обработка видео/Получение облака точек
2. Поиск объектов в облаке
3. Подготовка dxf файла
4. Генерация gcode файла

Последовательность операций отражена в блок-схеме на рисунке 4.1. При запуске алгоритма на вход передаются видео со сканера и dxf файл. После этого запускаются этапы обработки видео и dxf файла. Эти этапы независимы друг от друга и могут выполняться параллельно. Этап обработки видео включает в себя предобработку каждого кадра, затем выделение лазера в кадре, расчёт координат по формулам из главы 3. Результатом этого этапа является облако точек сцены. Оно передаётся в следующий блок для поиска объектов. Цель этого этапа в получении положений и ориентаций объектов в сцене. Затем данные об объектах и подготовленный рисунок передаются в последний блок алгоритма, где из них генерируется файл инструкций для принтера которые задают последовательность печати рисунка из dxf файла на каждый найденный объект в сцене.

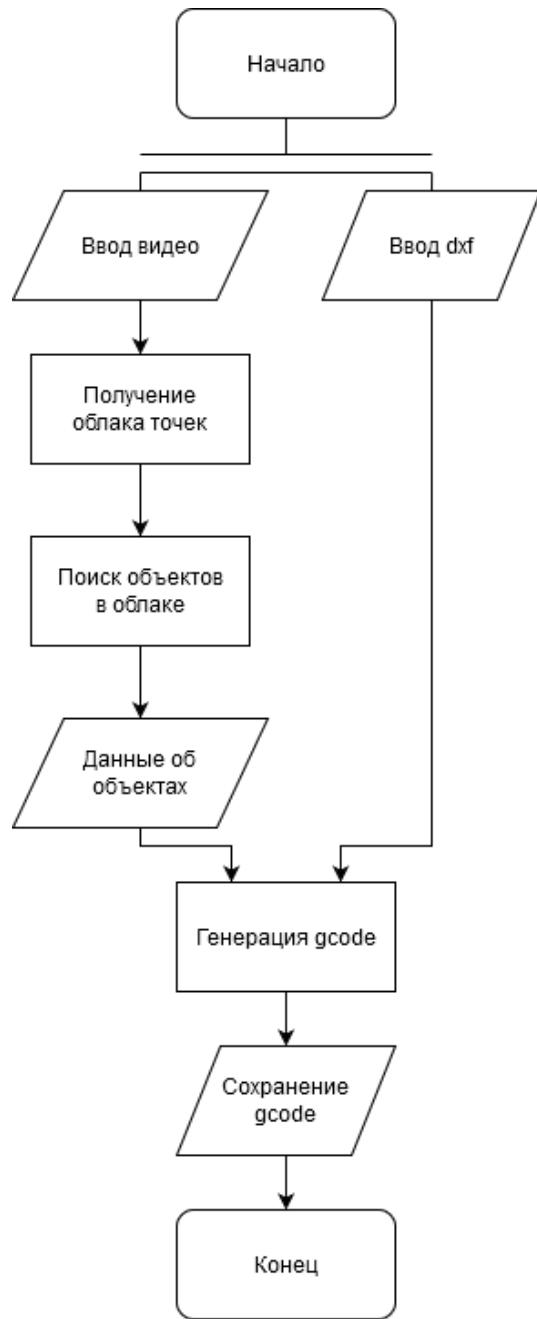


Рисунок 4.1 – Обобщённая блок схема алгоритма модуля

## 4.2. Блок обработки видео

Как уже было сказано, блок обработки видео включает в себя следующие шаги, которые выполняются для каждого кадра видео:

1. Предобработка кадра
2. Выделение лазера
3. Расчёт координат

После этого облако точек сохраняется в памяти и производится привязка координат к системе принтера. Этот этап необходим, потому что ввиду аппаратных ограничений сканирование в общем случае начинается в произвольном месте до рабочей поверхности и получить точную координату не представляется возможным. Результатом работы алгоритма становится облако точек, представляющее из себя матрицу размерностью (*количество кадров × ширина кадра*) где в каждой ячейке записана координата соответствующая этой точке. Таким образом облако точек одновременно является картой глубины сцены.

#### 4.2.1. Предобработка

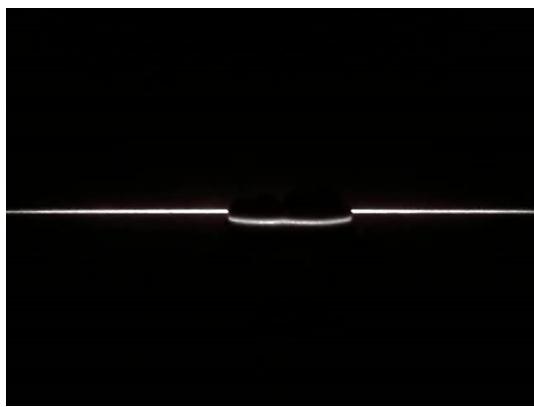


Рисунок 4.3 – Пример кадра из видео

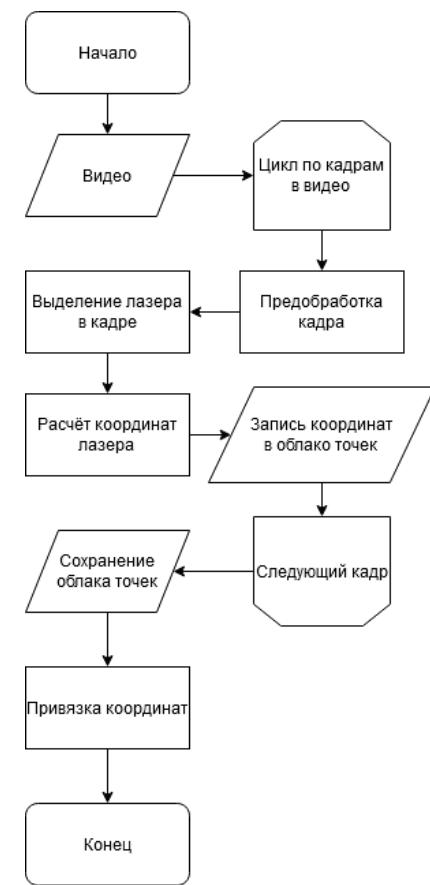


Рисунок 4.2 – Обобщённая блок схема обработки видео

Перед тем как выделить на изображении лазер, необходимо провести предобработку, которая сгладит шумы в кадре и уберёт большинство лишней информации. Первым шагом необходимо обрезать видео по заранее установленной области интереса. Вне этой области заранее гарантировано отсутствие полезной информации, т.е. вне этой области нет рабочей зоны.

Затем к изображению применяется фильтр гаусса. Он представляет со-

бой операцию свёртки гауссовой функцией на изображении. Применение фильтра гаусса позволяет снизить влияние шумов на изображении, что поможет на следующем шаге. Побочным эффектом фильтра является размытие изображения[11].

После этого с полученного изображения снимается «маска» – то есть изображение бинаризируется. В таком изображении единицами обозначена область, которую необходимо оставить, а нулями – отбросить. Для получения маски предусмотрено два подхода на выбор – бинаризация по заданному пороговому значению и бинаризация с порогом по Оцу. Последняя хороша тем, что позволяет избежать необходимости ручного подбора порогового значения.

Метод Оцу применим к изображениям на гистограмме которых есть только два пика. Хорошее пороговое значение для таких изображений находится между этими двумя пиками[12].

Это идеально подходит для нашего случая. Такая маска единицами помечает области на изображении, в которых с наибольшей вероятностью находится лазер, а нулями области, где лазера почти наверняка нет.

Последним шагом является применение маски к обрезанному изображению, что представляет собой простое поэлементное умножение. Таким образом получается изображение, в котором область, где почти наверняка лазера нет, состоит из пикселей значение которых равно нулю, остальная же часть остаётся нетронутой.

Это изображение затем передаётся в следующий блок алгоритма для выделения лазера.

#### 4.2.2. Выделение лазера

Для постановки задачи выделения лазера в первую очередь необходимо определить, что из себя представляет линия лазера на изображении. В нашем случае линия лазера в кадре располагается параллельно оси  $X$  изображения. Интенсивность в профиле лазера, т.е. сечении в перпендикулярном линии на-

правлении, соответствует Гауссовому распределению[13; 14] с некоторым шумом. Пик интенсивности, без учета шума, соответствует центру полосы лазера.

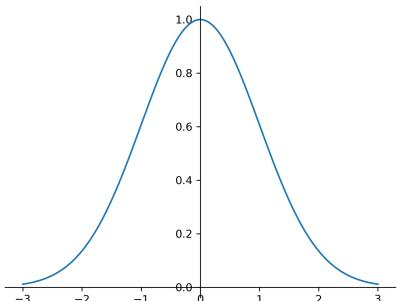


Рисунок 4.4 – Распределение Гаусса

Таким образом задача выделения лазера на изображении в первом приближении сводится к определению пика интенсивности для каждого профиля лазера, то есть в нашем случае для каждой колонки пикселей изображения.

Однако задача усложняется наличием шумов на изображении, которые постоянно исказывают профиль интенсивности лазера. Причинами шумов могут являться светочувствительный сенсор в камере, нежелательные отражения света лазера от поверхности, неподходящие условия окружения, программная обработка изображения на этапе кодирования-декодирования и так далее.

Помимо этого в общем случае линия лазера на изображении не является непрерывной. Разрывы могут возникать в следствии недостаточной контрастности лазера или попадания линии лазера в слепую зону. Некоторые методы достраивают недостающие участки лазера с помощью аппроксимации, другие не обрабатывают эти области.

Рассмотрим некоторые из существующих алгоритмов для выделения лазера и следующие их характеристики.

**Точность** – с какой точностью определяется положение лазера. Может быть **пикセルная** – положение определяется с точностью до одного пикселя – и **субпикセルная** – положение определяется с точностью в доли пикселя

**Восприимчивость к шуму** – критично ли наличие шумов на изображении для работы алгоритма

**Требования к вычислительным мощностям** – зависит от сложности операций и их количества.

**Метод максимумов** Данный метод находит максимум интенсивности в каждой колонке изображения и считает его центром лазера. Этот метод обладает пиксельной точностью. На работу данного метода влияют значительные шумы, как правило, связанные с окружением и поверхностью сканирования. Не требует серьёзных вычислительных мощностей, поскольку операция поиска максимума в колонке одна из простейших.

**Laplacian of Gaussian** На реальных изображениях пик профиля лазера как правило срезан, поэтому метод максимумов в данном случае не даёт максимальной точности[14].

Чтобы это преодолеть, к изображению можно использовать LoG преобразование, то есть применить свёртку гауссовой функцией на изображение, после чего вычислить вторую производную по изображению. Полученный результат инвертировать. На полученном новом изображении пики соответствуют центру лазера.

Данный метод в общем случае используется для выделения краёв и линий на изображении[15]. Для использования этого метода необходимо выбрать дисперсию  $\sigma$  гауссового распределения и окно свёртки. При малых  $\sigma$  таким образом возможной найти линии малой толщины, при больших  $\sigma$  наоборот, но одновременно за одно преобразование найти линии сильно отличающейся толщины невозможно. Параметр  $\sigma$  можно найти эмпирически, но можно воспользоваться оценкой  $\sigma > \frac{w}{\sqrt{3}}$  из работы [15], где  $w$  – ширина линии лазера в пикселях.

Данный метод также обладает пиксельной точностью. Более устойчив к шумам на изображении из-за использования гауссового размытия, что, однако,

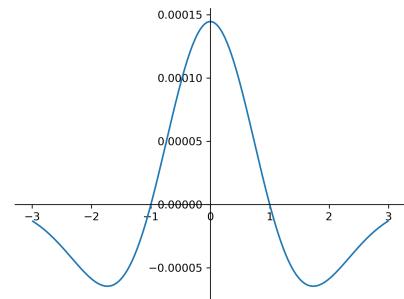


Рисунок 4.5 – Профиль интенсивности лазера после LoG преобразования

из работы [15], где  $w$  – ширина линии лазера в пикселях.

ухудшает выделение лазера на краях объектов. Из-за LoG преобразования данный метод требует больших вычислительных ресурсов, чем метод максимумов и, соответственно, выполняется медленнее.

**Квадратичная аппроксимация** Предыдущие два метода возможно улучшить и привести точность к субпиксельной[14]. Если рассмотреть профиль интенсивности лазера по пикселям, то в общем случае действительный максимум приходится на значение между пикселей. Также можно заметить что в окрестности максимума профиль представляет собой некоторую параболу  $y = ax^2 + bx + c$ . Взяв три точки – точку перед максимумом, точку максимума, точку после максимума – легко найти коэффициенты этой параболы и, соответственно, вершину, которая будет являться действительным максимумом.

Данный метод лучше применим к LoG трансформации, т.к. уже было сказано, что изначальный профиль лазера как правило обрезан при вершине. Однако в определённых условиях можно применять данный подход и для метода максимумов.

Применение этого метода увеличивает требование к вычислительным мощностям, однако незначительно, так как вычисления относительно просты и легко оптимизируются.

**Метод центра масс (Gray-Gravity Method)** По данному методу центр лазера  $(i_c, j_c)$  рассчитывается как «центр масс» интенсивностей в колонке  $j_c$  изображения по формуле

$$i_c = \left( \sum_{i=1}^H I(i, j_c) \cdot i \right) \cdot \left( \sum_{i=1}^H I(i, j_c) \right)^{-1} \quad (4.1)$$

где  $H$  – высота кадра в пикселях,  $I(i, j_c)$  интенсивность  $i$ -го пикселя в колонке.

Данный метод обладает субпиксельной точностью[16]. Однако, он крайне чувствителен к шумам, поэтому несмотря на субпиксельную точность его применение в задачах требующих высокой точности нежелательно. При этом данный

метод один из самых быстрых и нетребовательных к вычислительным мощностям в виду тривиальности вычислений.

### **Улучшенный метод центра масс (Improved Gray-Gravity Method)**

Улучшенный вариант предыдущего метода[16]. Предполагает два основных шага – предварительное определение центра лазера методом центра масс, затем уточнение координаты.

Уточнение координаты происходит следующим образом: С помощью методом скользящих наименьших квадратов находится касательная, нормаль и кривизна линии в точке предварительного центра. На основе этих значений подбирается прямоугольная область ограничивающая множество точек для расчётов. Затем рассчитывается отклонение действительного центра лазера от предварительного на основе интенсивности точек в регионе и их расстояния до линии пересекающей предварительный центр по направлению касательной в этой точке.

Достоинством этого метода является то, что он учитывает направление линии лазера в кадре и кривизну линии. Предыдущие методы предполагают что лазер параллелен оси  $x$  изображения, из-за чего теряют в точности ввиду искажения распределения интенсивности.

Данный метод обладает субпиксельной точностью и не настолько чувствителен к шумам как обычный метод центра масс. Также предварительным шагом можно применять любой другой метод. Однако данный метод сложнее реализовать и он более требователен к вычислительным мощностям, чем предыдущие методы.

**Другие методы** Кроме упомянутых методов существуют множество других. В частности *Gaussian Fitting*[13] и *Usamentiaga Method*[17].

*Gaussian Fitting* основан на методе Штегера, описанном в работе [15]. В работе [13] выдвигается математический аппарат для определения параметров

распределения лазера на основе данных с изображения, в частности соотношения сигнал/шум.

Суть данного метода заключается в том, чтобы отыскать реальное распределение интенсивности лазера на изображении без влияния шума и таким образом найти реальный центр лазера.

Достоинствами этого метода являются высокая устойчивость к шумам и высокая точность определения центра лазера.

Недостатком является большое время выполнения даже для одного изображения, что не позволяет использовать этот метод в задачах требующих быстрого выполнения, а также значительная сложность реализации алгоритма.

Usamentiaga method решает задачу выделения лазера в сложных индустриальных условиях и представляет алгоритмы устойчивые к шумам и нежелательным отражениям лазера. Также метод интересен тем, что предлагает алгоритм заполнения зон, где выделение лазера не удалось. В своей основе данный метод представляет комбинацию метода максимумов и метод центра масс, применённый в окрестности максимума.

Данный метод относится к субпиксельным и очень устойчив к шумам различного характера. При этом точность определения лазера не уступает gaussian fitting при значительно меньшем требовании к вычислительным мощностям. По данным работы [16] данный метод в 90 раз быстрее gaussian fitting, но в 3 раза медленнее метода центра масс. Это позволяет использовать его в задачах требующих малого времени выполнения. Однако он также как и gaussian fitting сложен в реализации.

**Выбранный метод** В конечном итоге основным был выбран метод Laplacian of Gaussian в комбинации с квадратичной аппроксимации описанный в статье [14], поскольку он обладает наиболее оптимальными характеристиками работы и время выполнения сканирования для видео из 1240 кадров (длительность 40 секунд с частотой 30 кадров в секунду) составляет около 9 секунд, что удовле-

творяет техническому заданию.

#### **4.2.3. Привязка координат**

Ранее было сказано, что ввиду аппаратных ограничений видео сканирования в общем случае начинается в неопределённый момент перед рабочей зоной. При этом нет возможности узнать координату сканера в момент начала записи видео. Это порождает проблему несоответствия рассчитанных в процессе обработки координат и реальных. Таким образом необходимо разработать алгоритм, который позволял бы по данным с видео определить необходимое смещение координат.

Основная идея такого алгоритма состоит в том, что необходима уникальная метка-идентификатор, имеющая заранее известную координату. Такая метка и алгоритм должны удовлетворять следующим требованиям:

- Лёгкое определение по входному видео
- Устойчивость к ложноположительным срабатываниям
- Лёгкое внедрение в конструкцию
- Минимальное влияние на общее время работы алгоритма

Предлагается использовать рельефные идентификаторы на столе, которые будут образовывать некоторый паттерн схожий с идеей штрихкода. Таким образом после сканирования в облаке точек сцены будет видна данная метка, которую возможно распознать программными методами и рассчитать необходимое смещение.

В конечном итоге идентификатор представляет собой ряд меток (рисунок 4.6) расположенных на некотором заданном расстоянии друг от друга. Метки же в свою очередь представляют некоторую прямоугольную поверхность заданной ширины на определённой высоте от стола. Высота от стола для всех меток одинаковы, остальные параметры произвольны.

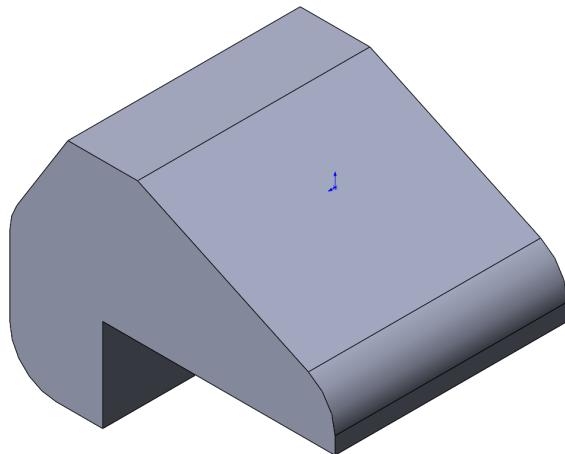


Рисунок 4.6 – Рельефная метка

Алгоритм перебирает профили сцены в облаке точек и сравнивает их на соответствие заданному идентификатору. Как только алгоритм находит подходящий профиль рассчитывается необходимое смещение, которое применяется ко всем координатам в облаке.

### 4.3. Блок поиска объектов

Задача поиска объектов в произвольном облаке точек одна из сложнейших. Такие задачи в настоящее время пытаются решить с использованием нейронных сетей и «искусственного интеллекта». На данный момент нет стандартных подходов к решению этой задачи.

Однако в нашем случае задача значительно упрощается по следующим причинам:

- имеется карта глубины с прямым соответствием пикселей и реальных координат
- объекты относительно простые
- необходимо распознать только верхнюю часть объектов

Поэтому возможно перевести задачу из распознавания объектов в пространстве в распознавание замкнутых контуров на изображении.

На вход алгоритму подаётся карта глубины сцены, на которой видны изделия. Применив к изображению алгоритм watershed найдём контур для каждого объекта, даже в случае их прямого соприкосновения.

Данный алгоритм рассматривает изображение как топографическую карту, в которой высокая интенсивность обозначает пики, а низкая долины. Данный рельеф заполняется водой разных цветов (маркеры), начиная с изолированных долин. В местах смешивания выстраиваются барьеры. Этот процесс происходит до тех пор, пока все пики не окажутся под водой. Полученные таким образом водоёмы разных цветов обозначают различные контуры изображения[18].

Поскольку карта глубины даёт прямое соответствие между пикселями и координатами, то наш контур задаётся некоторым пространственным многоугольником. Отбросив  $z$  координату, получим проекцию контура на плоскость. Затем можно рассчитать моменты этого контура и получить его центроиду. Так же через моменты можно найти ориентацию контура в пространстве (поворот вокруг центра), это эквивалентно вписыванию эллипса в контур и нахождения угла наклона главной диагонали.

Момент  $m_{pq}$  контура можно посчитать по следующей формуле

$$m_{pq} = \sum_k^n i_k^p j_k^q, \quad (4.2)$$

где  $p$  и  $q$  – порядки момента;

$(i_k, j_k)$  – координаты вершины контура, пиксели или миллиметры;

$n$  – количество вершин контура;

Центр контура рассчитывается по формуле

$$(i_c, j_c) = \left( \frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (4.3)$$

Угол ориентации в пространстве рассчитывается по формуле

$$\theta = \frac{1}{2} \arctan \left( \frac{b}{a - c} \right) + (a \leq c) \frac{\pi}{2}, \quad (4.4)$$

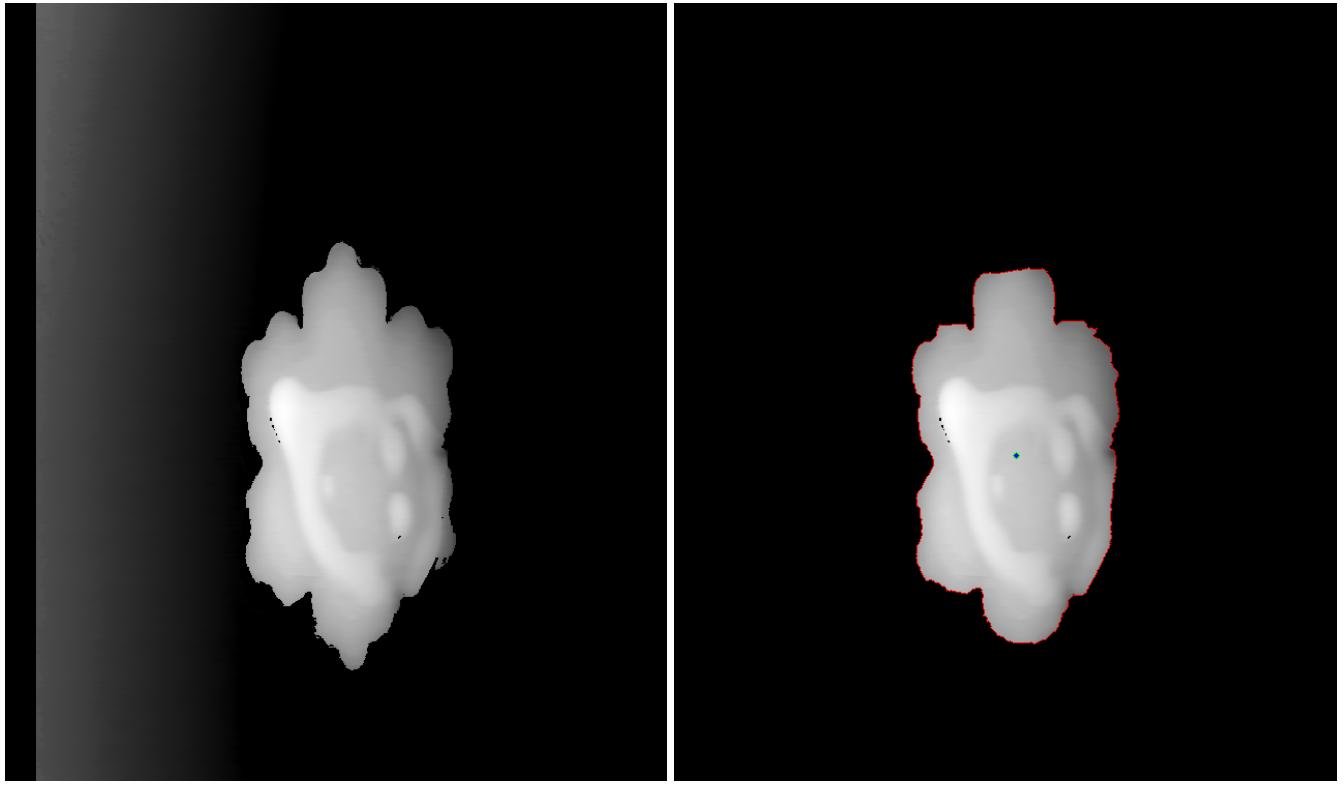
где  $a = \frac{m_{20}}{m_{00}} - i_c^2$ ;  $b = 2\frac{m_{11}}{m_{00}} - i_c j_c$ ;  $c = \frac{m_{02}}{m_{00}} - j_c^2$

Таким образом для каждого найденного контура программой формируется объект содержащий следующие данные:

- область в облаке точек, содержащую только этот контур
- координату центра объекта в плоскости
- ориентацию объекта в плоскости

Область в облаке сохраняется с целью оптимизации по времени алгоритма проекции рисунка на рельеф.

На рисунке 4.7 изображён пример работы алгоритма для случая одного объекта. Видно, что некоторые детали на краях изделия пропадают. Это происходит из-за использования морфологических операций при предобработке до применения алгоритма watershed. Однако, это не мешает определить необходимые параметры объекта, т.к это происходит равномерно и симметрично по периметру объекта.



(a) Карта глубины поданная на вход

(б) Обнаруженные объекты

Рисунок 4.7 – Пример работы алгоритма

#### 4.4. Блок генерации gcode

3D-принтеры, и прототип в частности, не способны описывать сложные кривые в пространстве. Возможно описание окружностей и, в некоторых случаях, простейших сплайнов только в горизонтальной плоскости. Это существенное препятствие в нашем проекте, так как сопло печатающего элемента должно огибать сложную поверхность. Так же dxf рисунок может задавать контуры сложной формы даже в плоскости. Для преодоления данных ограничений было решено реализовать слайсер-алгоритм решающий эту задачу.

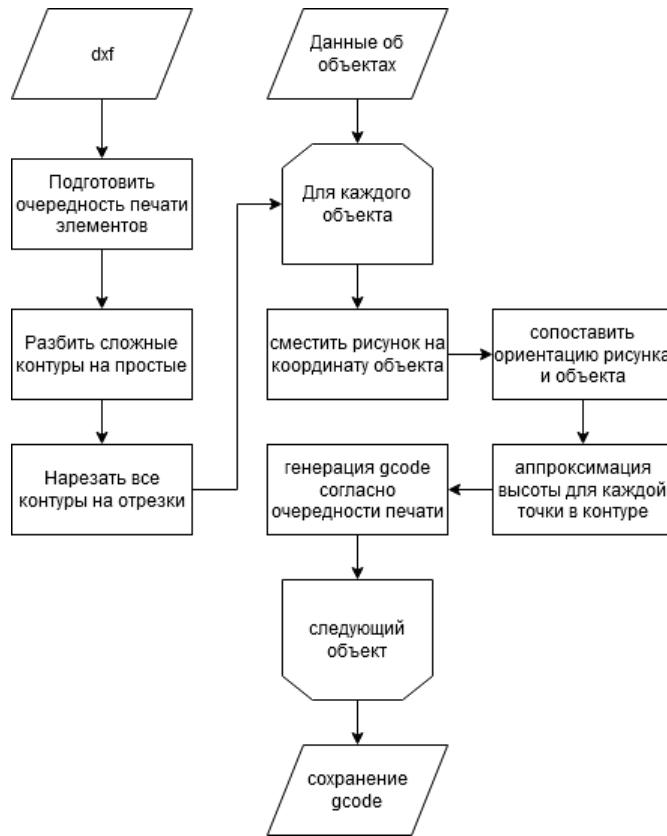


Рисунок 4.8 – Блок-схема генератора gcode

На вход алгоритма поступают данные об объектах и рисунок. В первую очередь происходит обработка dxf файла – формируется очередьность нанесения элементов рисунка в зависимости от их примыкания и расположения в структуре файла; сложные элементы (например, сплайны) разбиваются на более простые; все элементы «нарезаются» на отрезки определённой длины, заданной пользователем.

Затем обработанный файл комбинируется с данными о каждом объекте. Для каждой точки в контуре аппроксимируется соответствующая высота на основе данных облака. В процессе разработки были опробованы три варианта аппроксимации – выбор ближайшей точки, среднее  $k$ -ближайших соседей и метод скользящих средних квадратов. Проиллюстрируем работу алгоритмов на примере dxf-файла изображённом на рисунке 4.9.

**Ближайшая точка** По данному методу для заданной  $(x_p, y_p)$  координаты в облаке ищется аппроксимирующая точка  $(x_a, y_a, z_a)$  для которой  $(x_a - x_p)^2 +$

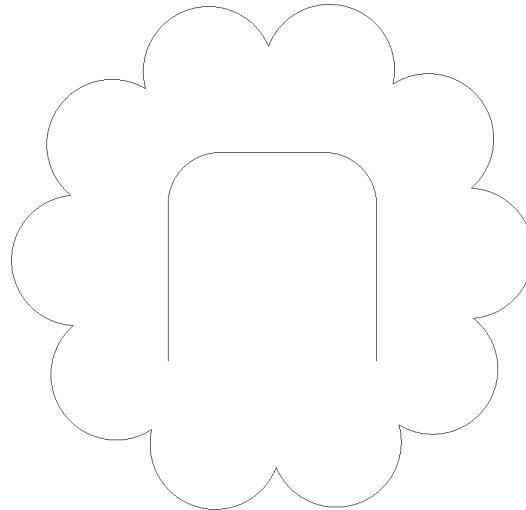


Рисунок 4.9 – Пример рисунка для нанесения

$(y_a - y_p)^2$  есть минимум среди всех точек в облаке.

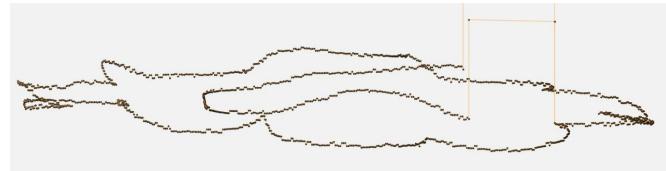


Рисунок 4.10 – Метод ближайшей точки

**$k$ -ближайших соседей** По данному методу выбирается  $k$  точек, расстояние до проекции которых от необходимой точки минимально. Затем вычисляются среднее координаты  $z$  этих точек. Полученное значение является аппроксимированной высотой искомой точки.

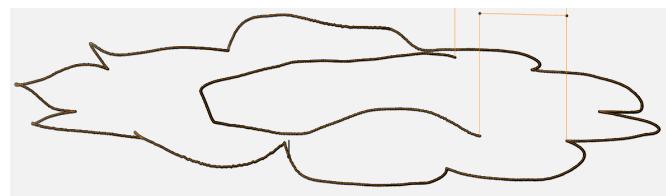


Рисунок 4.11 – Метод  $k$ -соседей;  $k = 30$

**Скользящие наименьшие квадраты** Это метод поиска некоторой полиномиальной функции, которая аппроксимирует данные с наименьшей квадратичной ошибкой в окрестности заданной точки [19]. То есть имея множество  $n$  точек

$\{M_i(x_i, y_i, z_i)\}_{i=1}^n$  и опорную точку  $M_p(x_p, y_p)$  необходимо найти коэффициенты  $c_{kl}$  следующей функции

$$f(x, y) = \sum_{k=0}^p \sum_{l=0}^q c_{kl} x^k y^l, \quad (4.5)$$

где  $p, q$  – максимальная степень  $x$  и  $y$  переменных соответственно. Для нахождения коэффициентов необходимо решить задачу минимизации следующей функции

$$E = \sum_i^n w_i(r_i) \left( \sum_{k=0}^p \sum_{l=0}^q c_{kl} x_i^k y_i^l - z_i \right), \quad (4.6)$$

где  $w_i$  – весовой коэффициент задающийся следующим образом

$$w_i(r_i) = \begin{cases} 1 - 6r_i^2 + 8r_i^3 - 3r_i^4, & r_i \leq 1 \\ 0, & r_i > 1 \end{cases} \quad (4.7)$$

где  $r_i$  – нормализованное расстояние между опорной точкой  $M_p$  и точками  $M_i$  из множества, рассчитываемое по формуле

$$r_i = \sqrt{(x_i - x_p)^2 + (y_i - y_p)^2} \cdot R^{-1}, \quad (4.8)$$

где  $R$  – радиус влияния, точки не входящие в этот радиус не имеют эффекта на аппроксимацию. Уравнения весовых коэффициентов выведены в работе [20] Минимизируя  $E$  и переписывая в матричном виде получаем следующее уравнение для коэффициентов функции  $f(x, y)$

$$c = [B^T W(M_p) B]^{-1} B^T W(M_p) z, \quad (4.9)$$

где  $z = [z_1 \dots z_n]^T$  – вектор значений координат точек множества,  $B$  – обобщённая матрица Вандермонда данных из  $(x, y)$  координат точек множества, в данном случае

$$B = \begin{bmatrix} 1 & y_1 & x_1 & \cdots & x_1^p y_1^q \\ 1 & y_2 & x_2 & \cdots & x_2^p y_2^q \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & y_n & x_n & \cdots & x_n^p y_n^q \end{bmatrix} \quad (4.10)$$

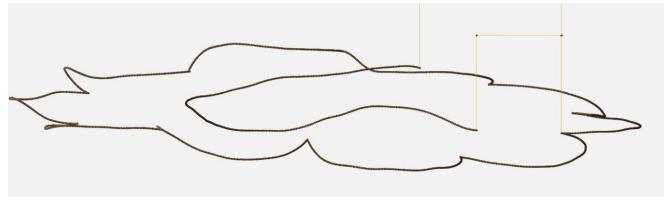


Рисунок 4.12 – Метод скользящих наименьших квадратов

Найдя таким образом аппроксимацию поверхности легко вычислить координату  $z$  искомой точки.

**Выбранный метод** Из трёх методов предпочтительным является метод скользящих наименьших квадратов. Данный метод, как правило, применяется для восстановления сложных поверхностей по облаку точек. Он является более плавным и точным, чем остальные методы, что легко видно по рисункам 4.10-4.12. Однако данный метод значительно требовательнее к вычислительным мощностям, поэтому его применение ограничено и рекомендовано при повышенных требованиях точности.

Метод ближайшей точки самый быстрый из рассмотренных и не требует к вычислениям. Однако при этом контуры получаются наименее плавными, что может повредить изделие при работе, где необходима точность.

Метод  $k$ -ближайших соседей является промежуточным вариантом указанных двух методов.

Для системы реализованы все три рассмотренных метода с возможностью пользователем выбрать подходящий. Для этого будут составлены специальные методические указания, описывающие в каких случаях лучше использовать ту или иную аппроксимацию.

## Глава 5

# ОПИСАНИЕ РАБОТЫ СИСТЕМЫ

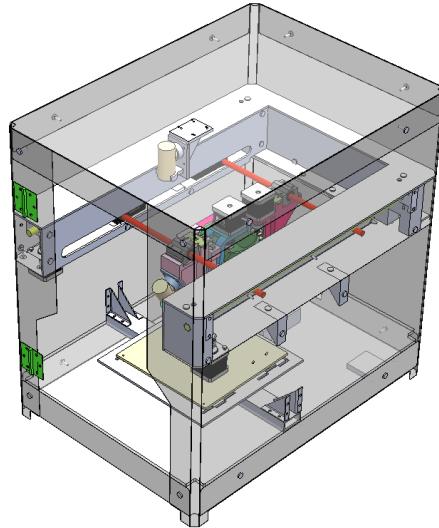


Рисунок 5.1 – Конечный вид прототипа системы автоматического фигурного нанесения пищевой пасты

Перед эксплуатацией модуля необходимо провести калибровку камеры и сканера в целом с помощью специального паттерна. Калибровка камеры позволяет получить матрицу её внешних и внутренних параметров – положение камеры в пространстве, фокусное расстояние и реальный центр изображения. Калибровка сканера позволяет получить параметры  $H$  и  $\alpha$  по формулам из раздела 3.3.

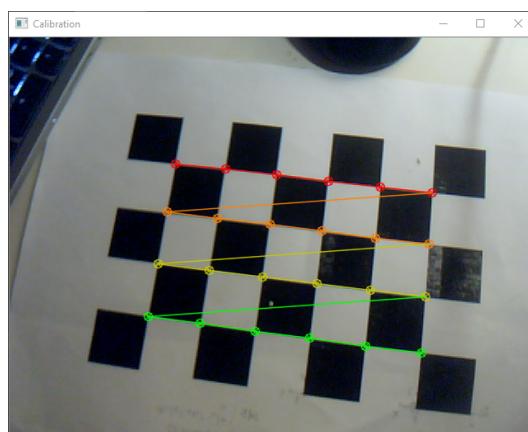


Рисунок 5.2 – Процесс калибровки камеры

Эксплуатация системы происходит следующим образом. В рабочую зону располагают объекты, закрывают дверцу прототипа и выключают освещение. По команде оператора начинается процесс съёмки видео для обработки. По окончанию съёмки файл отправляется на компьютер, который обрабатывает полученные данные по алгоритму описанному в главе 4.



Рисунок 5.3 – Процесс обработки видео

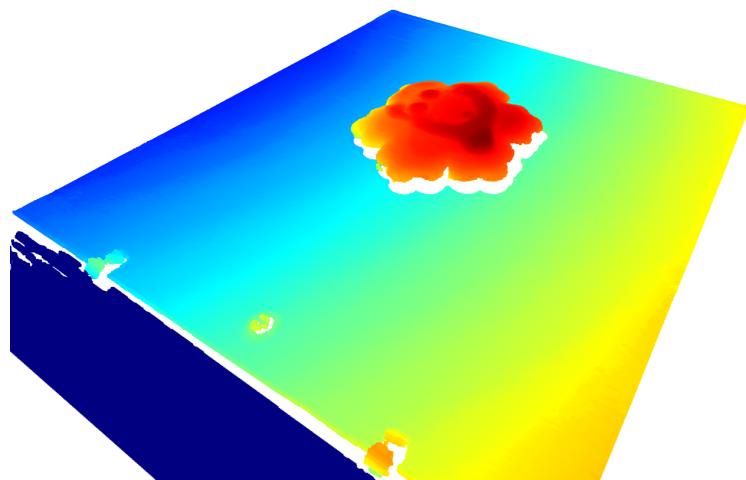


Рисунок 5.4 – Облако точек сцены

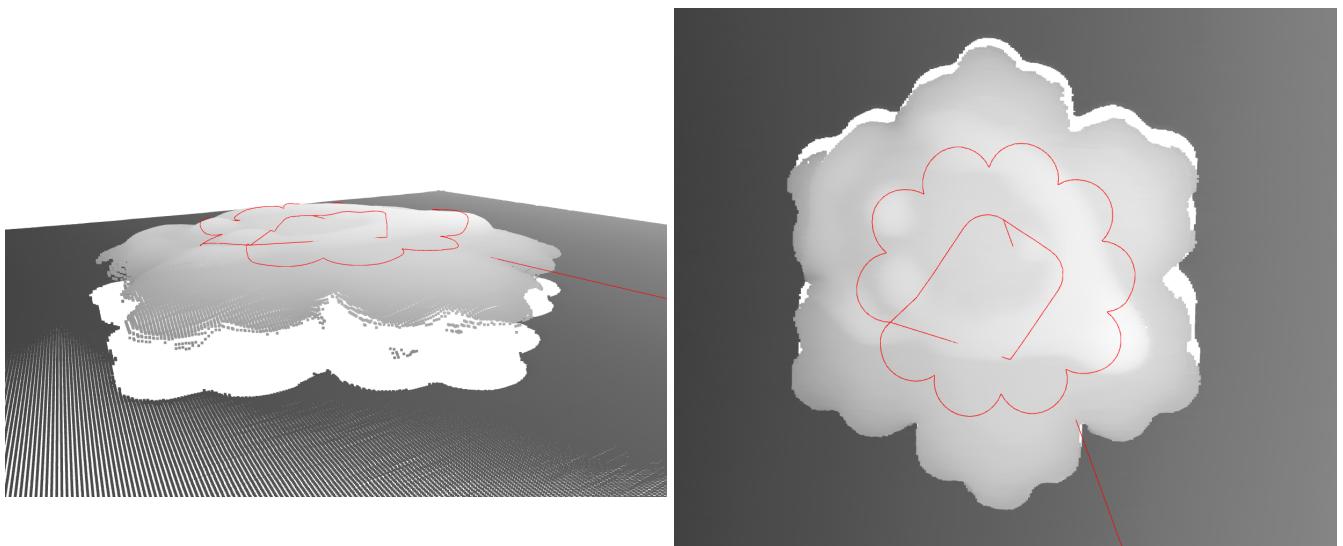


Рисунок 5.5 – Наложение контура из dxf-файла на объект

После обработки данных сгенерированный gcode отправляется обратно в прототип на исполнение. Затем изделия подвергаются проверке системой качества, описанной в работе [21].

## Заключение

В работе была разработана система 3D-сканирования для прототипа системы автоматического фигурного нанесения пищевой пасты. Разработанная система удовлетворяет требованиям технического задания.

Система строит облако точек рабочей зоны по видео файлу процесса сканирования, находит объекты в рабочей зоне и генерирует gcode на основе заданного dxf файла для нанесения пасты на объекты. Скорость обработки видео файла составляет примерно 9 с. Итоговая стоимость компонентов системы составляет 4850 рублей, что значительно дешевле представленных на рынке аналогов.

Дальнейшим развитием работы будет улучшение качества входных данных и реализации сканирования в цвете. Также улучшение алгоритма генерации gcode и реализация более совершенного алгоритма поиска объектов в облаке. Помимо этого, при условии успеха реализации сканирования в цвете, возможно переработать систему привязки координат, что позволит добиться большей надёжности. Написание методических рекомендаций по выбору метода аппроксимации будет являться отправной точкой при написании пользовательской документации.

## Список литературы

1. *Sony.* Xperia 3D Creator. — режим доступа: <https://3d-creator.sonymobile.com/>.
2. *Sembera C.* Sony 3D Creator Review: A 3D Scanner in Your Pocket. — режим доступа: <https://all3dp.com/1/sony-3d-creator-review/>.
3. *Гужов В. И.* Методы измерения 3D-профиля объектов. Контактные, триангуляционные системы и методы структурированного освещения: учеб. пособие. — 2015. — с. 1—82. — ISBN 9785778227279.
4. Ciclop 3D Scanner (BQ and Horus). — режим доступа: <https://www.instructables.com/id/Ciclop-3D-Scanner-BQ-Horus/>.
5. MA-RF-6306-EN - FARO ScanArm Tech Sheet-Feb2020. — режим доступа: <https://faro.app.box.com/s/kw8eigxhbd49vva5mffyy8p77nxfa17s/file/623707651582>.
6. *hesamh.* DIY 3D Scanner Based on Structured Light and Stereo Vision in Python Language. — режим доступа: <https://www.instructables.com/id/DIY-3D-scanner-based-on-structured-light-and-stere/>.
7. *Скрипко Р. Е.* Выпускная квалификационная работа «Разработка системы автоматического фигурного нанесения пищевых продуктов». — 2018.
8. *Мацуганов А. А.* (науч. рук. Монахов Ю. С.) Разработка печатающего узла кондитерского принтера. — 2020. — режим доступа: <https://kmu.itmo.ru/digests/article/4421>.
9. Machine vision system for curved surface inspection / M.-F. R. Lee [и др.] // Machine Vision and Applications. — 2000. — т. 12, № 4. — с. 177—188. — ISSN 0932-8092. — DOI: [10.1007/s001380000043](https://doi.org/10.1007/s001380000043).

10. Camera Calibration and 3D Reconstruction. — режим доступа: [https://docs.opencv.org/2.4/modules/calib3d/doc/camera\\_calibration\\_and\\_3d\\_reconstruction.html](https://docs.opencv.org/2.4/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html) (дата обр. 14.05.2020).
11. Corke P. Robotics, Vision and Control - Fundamental Algorithms In MATLAB ® Second, Completely Revised, Extended And Updated Edition. т. 75. — 2017. — с. 693. — ISBN 978-3-319-54413-7. — DOI: [10.1007/978-3-319-54413-7](https://doi.org/10.1007/978-3-319-54413-7). — режим доступа: <https://www.springer.com/gp/book/9783319544120>.
12. OpenCV: Image Thresholding. — режим доступа: [https://docs.opencv.org/3.4/d7/d4d/tutorial\\_py\\_thresholding.html](https://docs.opencv.org/3.4/d7/d4d/tutorial_py_thresholding.html) (дата обр. 14.05.2020).
13. Statistical behavior analysis and precision optimization for the laser stripe center detector based on Steger's algorithm / L. Qi [и др.] // Optics Express. — 2013. — т. 21, № 11. — с. 13442. — ISSN 1094-4087. — DOI: [10.1364/oe.21.013442](https://doi.org/10.1364/oe.21.013442).
14. Laser line detection with sub-pixel accuracy / A. Molder [и др.] // Elektronika ir Elektrotehnika. — 2014. — т. 20, № 5. — с. 132—135. — ISSN 13921215. — DOI: [10.5755/j01.eee.20.5.7114](https://doi.org/10.5755/j01.eee.20.5.7114).
15. Steger C. Subpixel-precise extraction of lines and edges // International Archives of Photogrammetry and Remote Sensing. — 2000. — т. XXXIII. — с. 141—156. — режим доступа: <http://ias.in.tum.de/people/steger/publications/2000/ISPRS-Congress-00-Steger.pdf>.
16. Sub-pixel extraction of laser stripe center using an improved gray-gravity method / Y. Li [и др.] // Sensors (Switzerland). — 2017. — т. 17, № 4. — ISSN 14248220. — DOI: [10.3390/s17040814](https://doi.org/10.3390/s17040814).
17. Usamentiaga R. n., Molleda J., Garcí a D. F. Fast and robust laser stripe extraction for 3D reconstruction in industrial environments // Machine Vision

and Applications. — 2012. — т. 23, № 1. — с. 179—196. — ISSN 09328092. — DOI: [10.1007/s00138-010-0288-6](https://doi.org/10.1007/s00138-010-0288-6).

18. Image Segmentation with Watershed Algorithm. — режим доступа: [https://docs.opencv.org/master/d3/db4/tutorial\\_py\\_watershed.html](https://docs.opencv.org/master/d3/db4/tutorial_py_watershed.html) (дата обр. 14.05.2020).
19. Veerabhadra R., Garimella R. V. A Simple Introduction to Moving Least Squares and Local Regression Estimation. — 2017.
20. Calculation of stratum surface principal curvature based on a moving least square method / G. qing LI [и др.] // Journal of China University of Mining and Technology. — 2008. — т. 18, № 1. — с. 59—63. — ISSN 10061266. — DOI: [10.1016/S1006-1266\(08\)60013-9](https://doi.org/10.1016/S1006-1266(08)60013-9).
21. Ма́йоров А. С. (науч. рук. Монахов Ю. С.) Система контроля качества продукции для кондитерского принтера. — 2020. — режим доступа: <https://kmu.itmo.ru/digests/article/4885>.