# 3. Экспериментальная проверка разработанного алгоритма

# 3.1 Организация вычислительного эксперимента

# 3.1.1 Проектирование и разработка модели в Unity

Реализация алгоритма включает использование датчиков изменения высоты, которые располагаются на определенном расстоянии друг от друга. Сигналы от этих датчиков передаются на центральный компьютер или контроллер для обработки данных.

Для эмуляции функциональности устройства был выбран игровой движок Unity, который является популярным и предоставляет возможность создания виртуальных сред и моделирования физических процессов. В рамках Unity была разработана симуляция, которая имитировала считывание показателей с датчиков. Таким образом, была осуществлена проверка алгоритма определения количества яиц без использования реальных яиц и физической модели. Геометрические показатели зависимости размера яиц от категории соответствует рисунку 1.

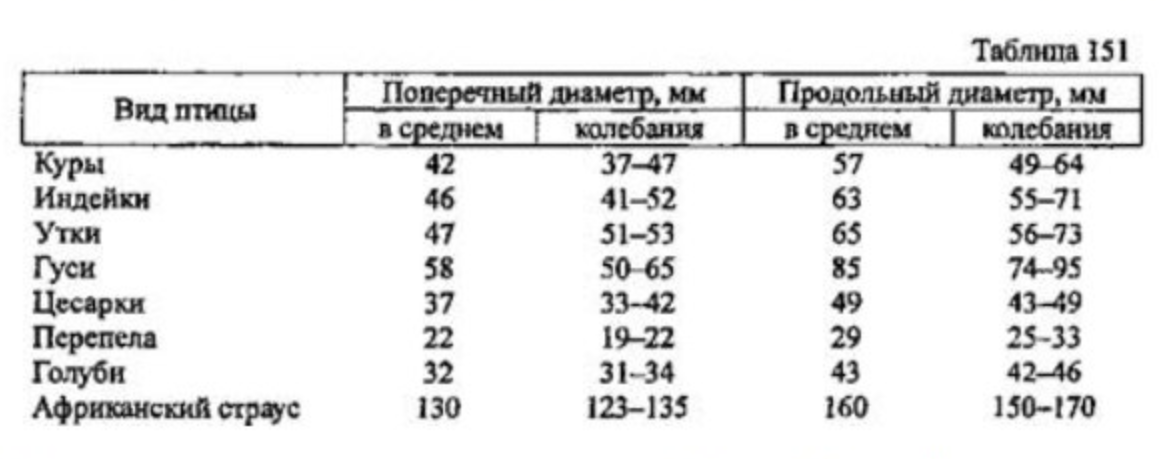


Рисунок 1 – Геометрические показатели зависимости размера яиц от категории

На рисунке 2 представлена генерация объектов, учитывая различные категории и положение яиц.

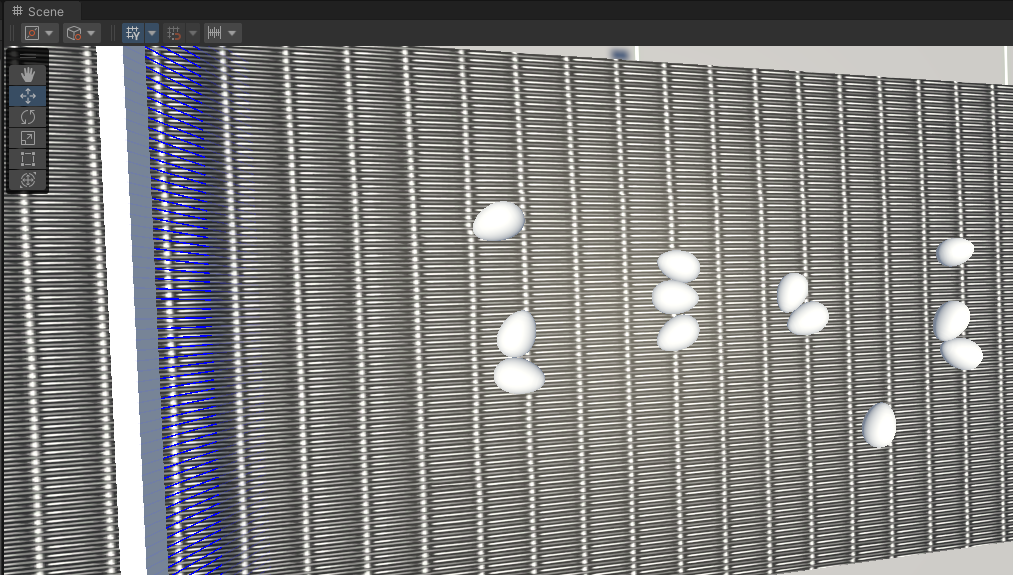


Рисунок 2 – Генерация яиц

Код генерации объектов представлен листингом 1.

Листинг 1 - Код генерации объектов

void Update()

{

float width = ConvWidthScript.getWidth();

if (Time.time > nextSpawn && GlobalVar.GetIsSpawn())

{

nextSpawn = Time.time + spawnDelay;

CalcEgg.addEggSpawnCol(spawnCol);

for (int i = 0; i < spawnCol; i++)

{

randomX = 0;

float rotateEgg = Random.Range(0.0f, 360.0f);

whereToSpawn = new Vector3(randomX, 2.0f, -7f);

GameObject Enemy = Instantiate(go, whereToSpawn, Quaternion.identity);

Enemy.transform.Rotate(0.0f, rotateEgg, 90f);

Enemy.transform.localScale = new Vector3(0.011f \* randomScale, 0.011f \* randomScale, 0.011f \* randomScale);//0.021

}

}

}

В Unity роль датчиков выполняют примитивные кубы, которые отправляют лучи (ray) через определенные промежутки времени. Основные характеристики шума и частоты обновления регулируются на основе настоящих датчиков VL53L0X. Внешний вид датчиков отображен на рисунке 3. Ширина датчиков, частота обновления и расстояние до конвейера могут быть настроены. Скорость движения конвейера можно задать в пределах от 1м/мин до 12 м/мин.

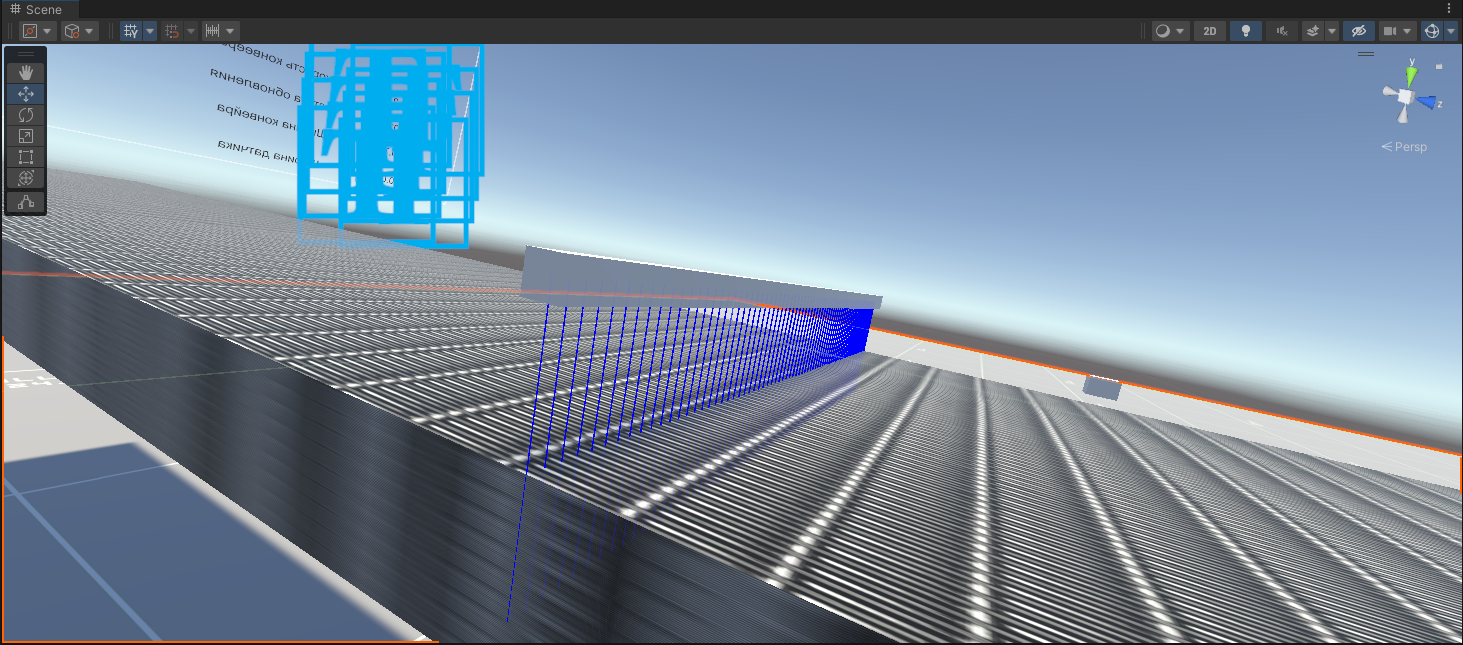


Рисунок 3 – Внешний вид датчиков

Код фиксирования показателей датчика представлен листингом 2.

Листинг 2 - Код фиксирования показателей датчика

void FixedUpdate()

{

nextUpdate = Time.time + GlobalVar.getSensorUpdateDelay();

Ray ray = new Ray(transform.position, transform.forward \* 0.15f);

Debug.DrawRay(transform.position, transform.forward \* 0.15f, Color.blue);

RaycastHit hit;

if (!Physics.Raycast(ray, out hit, 0.45f))

return;

float randomScale = Random.Range(-GlobalVar.getNoise(), GlobalVar.getNoise());

mas.Add(hit.distance + randomScale);

bool isIndexLast = (index == HeightMap.getMass().Count - 1);

if (isIndexLast)

{

HeightMap.CheckSensors();

}

if (hit.collider.gameObject.GetInstanceID() == -1268)

return;

float val = mas[mas.Count - 1];

if (val >= GlobalVar.getMinDistance())

return;

hit.collider.gameObject.GetComponent<Renderer>().material.color = Color.red;

Destroy(hit.collider.gameObject, 5f);

}

Записанные показания с датчиков сохраняются в бинарный файл. Структура бинарного файла представлена на рисунке 4.

Рисунок 4 – Структура бинарного файла

Общий вид сцены представлен на рисунке 5.

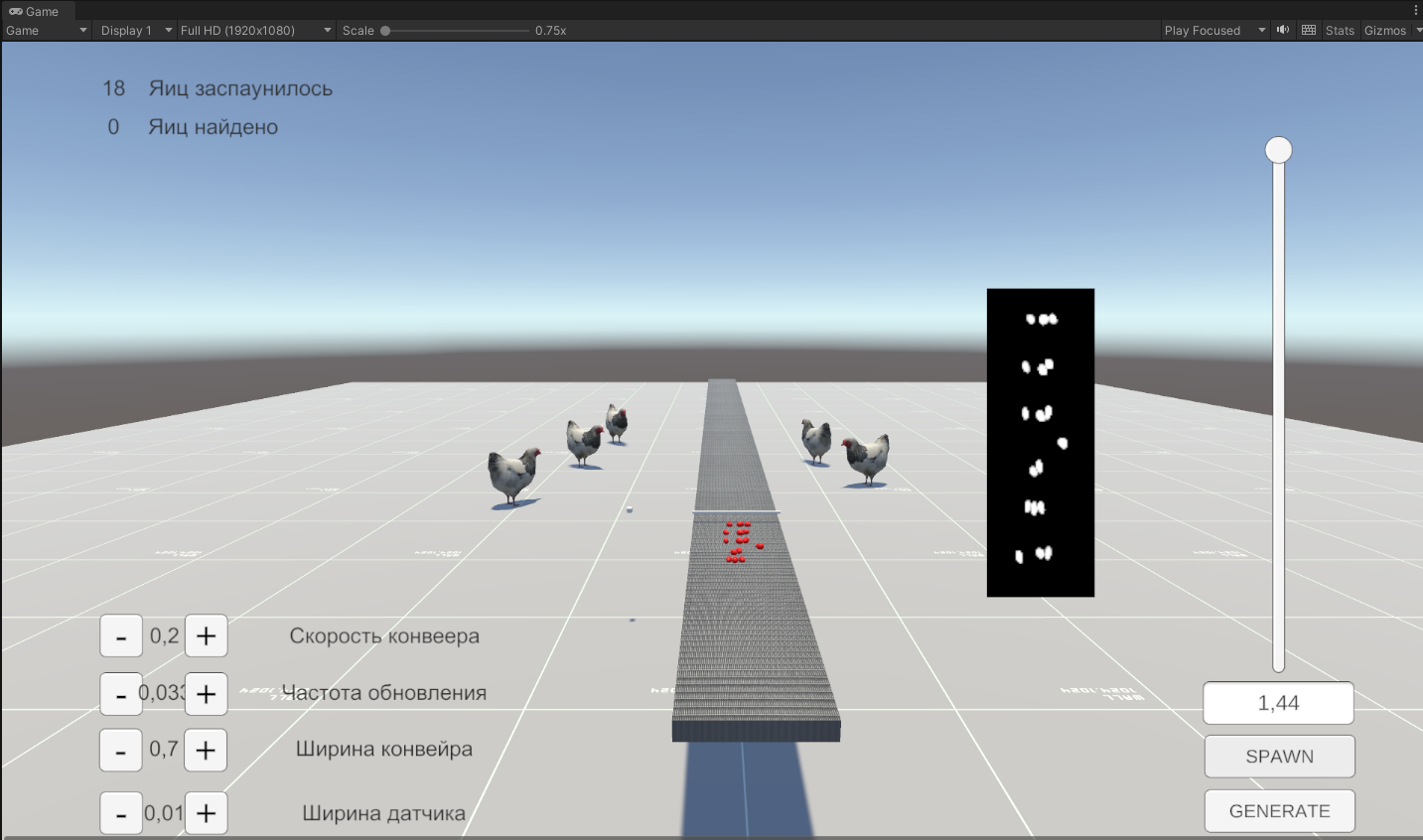


Рисунок 5 – Общий вид сцены

# 3.1.2 Проектирование и разработка приложения для оценки работы методов фильтрации

Приложение предназначено для оценки метода фильтрации. Блок-схема работы приложения представлена на рисунке 6.

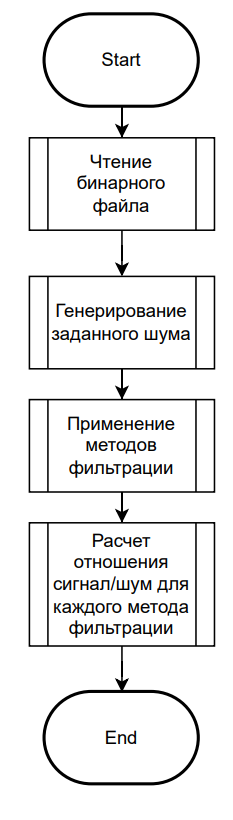
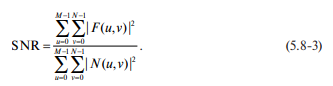


Рисунок 6 – Блок-схема работы приложения

Отношение сигнал/шум дает меру уровня информационного отношения мощности сигнала (т. е. исходного, неискаженного изображения) к уровню мощности шума. Изображения с низким шумом будут иметь большое значение SNR (1), а те же изображения с высоким уровнем шума — малое значение SNR. Само по себе данное соотношение имеет ограниченное значение, но оно является важной метрикой, используемой для описания характеристик алгоритмов восстановления.

(1)

где  - энергетический спектр неискаженного изображения;

 - энергетический спектр неискаженного изображения.

На основе результатов работы приложения по оценке пикового отношения, был выбран метод фильтрации данных - скользящее среднее. Полученные результаты показали, что этот метод обеспечивает достаточное снижение уровня шума и сохранение структуры данных на карте вершин с яйцами. Пример работы программы представлен не рисунке 6.

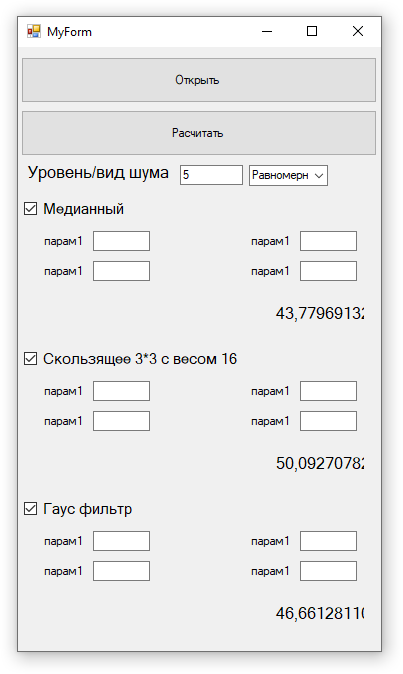


Рисунок 6 – результат работы приложения по оценке пикового отношения

Также следует отметить, что для каждой предметной области необходимо использовать свой метод фильтрации, чтобы обеспечить наилучший результат и оптимальное пиковое отношение сигнала к шуму в каждом конкретном случае.

# 3.1.3 Проектирование и разработка приложения для оценки работы алгоритма

Приложение предназначено для оценки работы алгоритма с заданным уровнем шума. Блок-схема работы приложения представлена на рисунке 7.

Рисунок 7 – Блок-схема работы алгоритма

Приложение Читает бинарный файл, структура которого представлена на рисунке 4. Для генерации шума было использовано равномерное распределение.

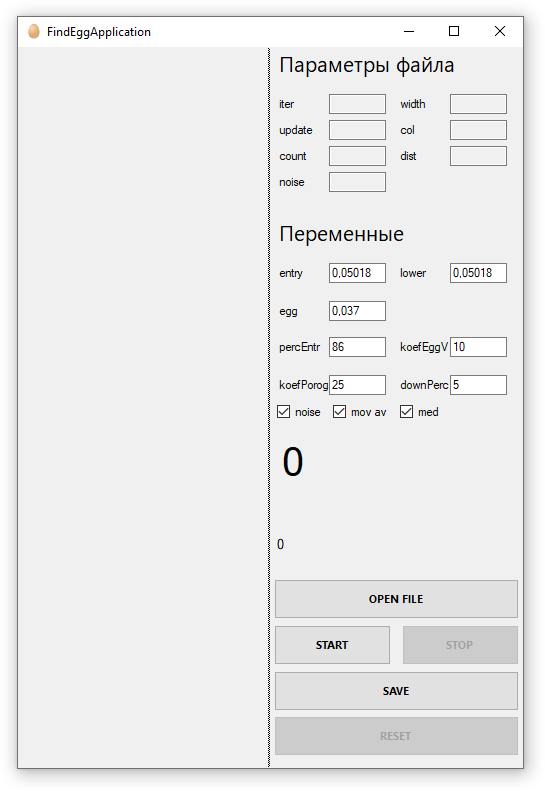


Рисунок N – Общий вид приложения

В блоке "Параметры файла" выводится общая информация из бинарного файла. пример открытого файла представлен на рисунке N.

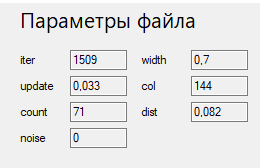


Рисунок N – Общая информация из бинарного файла

В блоке "Переменные" отображается порог обнаружения и переменные для работы алгоритма:

* entry - порог обнаружения;
* egg - базовый размер яйца на основе которого будет расчитываться порог обнаружения;
* percEntr – процент от базового размера яйца для расчета порога обнаружения;
* koefEggV - количество объектов будет основываться статистика по формуле N;
* koefPorog - коэффициент деления по формуле N.

Пример работы программы программы представлен на рисунке N.

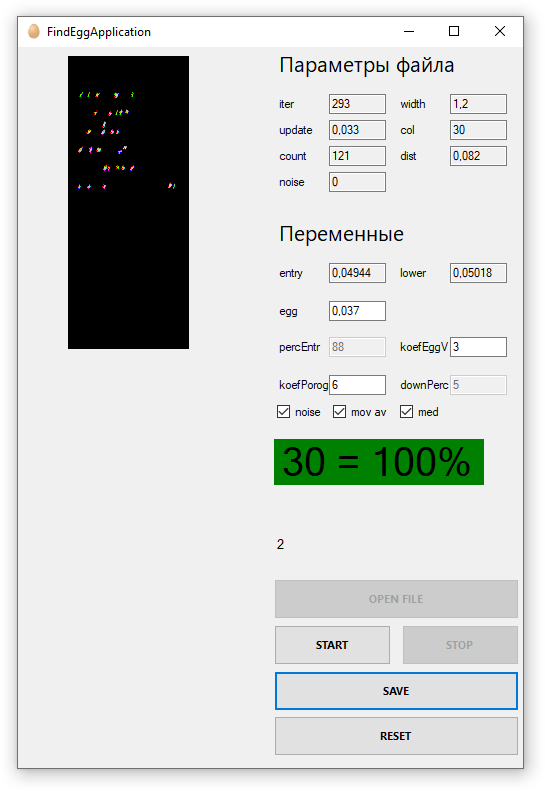


Рисунок N – Пример работы программы

- чтение данных

- вывод изображения

- метод фильтрации

- все классы и методы

- небольшой результат

# 3.2 Анализ результатов эксперимента

# 3.3 Формирование рекомендаций

# 3.4 Выводы по главе.