

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский
Нижегородский государственный университет им. Н.И. Лобачевского»
(ННГУ)

Институт информационных технологий, математики и механики
Кафедра математического обеспечения и суперкомпьютерных технологий

Направление подготовки:
«Фундаментальная информатика и информационные технологии»
Профиль подготовки:
«Компьютерная графика»

КУРСОВАЯ РАБОТА

Тема:
«Исследование ХАI-возможностей для ранней
диагностики стресса растений»

Выполнила:
студентка группы 381706-1
Максимова Ирина Игоревна

(подпись)

Научный руководитель:
проф. каф. МОСТ, д-р техн. наук
Турлапов Вадим Евгеньевич

(подпись)

Нижний Новгород
2021

Оглавление

1. Введение.....	4
2. Постановка задачи.....	7
3. Теоретическая часть.....	8
3.1 Матрица GLCM.....	8
3.2 Описание классификаторов.....	9
3.2.1 Линейный дискриминант Фишера.....	10
3.2.2 Дерево решений.....	10
3.2.3 Случайный лес.....	11
3.2.4 Метод опорных векторов.....	11
3.2.5 Метод k-ближайших соседей.....	12
3.2.6 Одноуровневый персептрон.....	12
3.3 Метрики качества алгоритмов классификации.....	13
4. Исследовательская часть.....	14
4.1 Обзор статистико-текстурного аппарата на RGB изображениях.....	14
4.1.1 Описание базы данных.....	15
4.1.2 Обзор изображений в красном канале.....	15
4.1.3 Исследование статистических признаков.....	17
4.1.4 Метод квантования изображений.....	19
4.1.4 Исследование текстурных признаков.....	21
4.1.5 Описание полного вектора признаков.....	22
4.1.6 Алгоритм извлечения вектора локальных признаков.....	23
4.1.7 Выбор оптимальных параметров классификаторов.....	24
4.1.8 Классификация на глобальных и локальных данных.....	25
4.2 Исследование применения NDVI _G аналога.....	27
4.2.1 Расчет NDVI _G образов изображений.....	28
4.2.2 Результаты использования вектора NDVI _G признаков.....	29
4.3 Построение XAI-блока в эксперименте на реальных условиях.....	30
4.3.1 Подготовка набора данных.....	31
4.3.2 Предварительная обработка изображений.....	32
4.3.3 Анализ результатов эксперимента.....	34
4.3.4 Модель регрессии SLP и ее анализ.....	37
4.3.5 Результаты исследования.....	41
4.4 Инструменты исследования.....	42
4.5 Описание структуры программы.....	42
5. Заключение.....	43
6. Список литературы.....	45

7. Приложение	46
Приложение А. Визуализация локальных признаков.....	46
Приложение В. Визуализация локальных признаков.....	47
Приложение С. Подбор параметров классификаторов.....	48
Приложение D. Показатели F-Score в задаче обнаружения болезни на основе локальных, глобальных и комбинированных признаков в красном канале	49
Приложение Е. Показатели F-Score в задаче классификации болезни на основе локальных, глобальных и комбинированных признаков в красном канале	49
Приложение F. Показатели в задаче обнаружения болезни на основе локальных, глобальных и комбинированных NDVI признаков	50
Приложение G. Показатели F-Score в задаче классификации болезни на основе локальных, глобальных и комбинированных NDVI признаков	50
Приложение Н. Листинг кода извлечения полного вектора признаков	51

1. Введение

Болезни растений являются одной из главных угроз глобальной продовольственной безопасности. Ежегодно болезни растений приводят к потере 10–16% мирового урожая сельскохозяйственных культур, стоимость которого оценивается в 220 миллиардов долларов [1]. Одновременно с потерей урожая происходит прирост населения мира. Согласно прогнозу ООН, численность населения мира достигнет 9,8 млрд. человек в 2050 году и 10,7 млрд. в 2100 году [2]. Чтобы удовлетворить потребности в продовольствии постоянно растущего населения, необходимо свести к минимуму потери мирового урожая.

Одним из главных шагов на пути к сохранению урожая является использование эффективных методов для выявления заболеваний растений.

Мониторинг сельскохозяйственных культур играет ключевую роль в успешном выращивании урожая. В настоящее время основным подходом, используемым на практике, является наблюдение невооруженным глазом эксперта. Обращение за консультацией к практикующим врачам, диагностика заболевания и получение заключения эксперта - долгая, дорогая и трудоемкая практика. Поэтому в настоящее время возникает острая необходимость в создании быстрых и недорогих современных методов диагностики стрессовых состояний растений.

Актуальность данной работы заключается в автоматизации процесса мониторинга сельскохозяйственных угодий. **Проблематика** данной работы заключается в возможности методов объяснимого искусственного интеллекта (XAI, eXplainable Artificial Intelligence) выполнять раннюю диагностику стрессовых состояний растений.

Основными возбудителями болезней культурных растений являются паразитические бактерии, грибы и вирусы. Данные патогены вызывают отклонения жизненных процессах поражённого растения и приводят к значительным изменениям не только его внутреннего состояния, но и внешнего вида. Внешними признаками болезней растений являются: увядание, разрушение, деформация органов растения, отмирание, изменение цвета тканей, появление гнили, пятен, налета, и др.

Поскольку появление побочных внешних эффектов сигнализирует о реакции растения на заболевание, то диагностика растений может проводиться путем выявления зараженной области. С точки зрения анализа изображения всякое внешнее проявление болезни листа сопровождается изменением его статистических и текстурных признаков. В настоящее время для анализа таких признаков используется машинное обучение (ML, Machine Learning), поскольку оно обладает высоким показателем точности и большим потенциалом.

Чтобы обеспечить применение методов ML, а затем глубокого обучения (DL, Deep Learning), были созданы наборы данных для различных растений и их заболеваний. Самые интересные из них, как правило, сопровождаются публикацией прикладных методов ИИ. Так, в публикации [3] 2013 анализируется применение методов ML к данным гиперспектральной визуализации (HSI) листьев сахарной свеклы (нормальных и пораженных 3 типами болезней, по 200–400 проб каждый; гиперспектрометр с диапазоном 400–1000 нм и разрешение до 2,8 нм). Обучение проводится на изображениях по спектральному индексу растительности (SVI). Достигнутая точность для 1-4 классов: 89%, 92%, 87%, 85%.

В статье [4] 2018 уже анализируется применение 5 различных моделей DL: AlexNet, AlexNetOWTBn, GoogLeNet, Overfeat, VGG. Набор данных содержит 87848 изображений RGB для 25 типов растений в 58 комбинациях [растение, болезнь], включая здоровые растения. Успешность классификации составляет 99,53%.

Последние годы также отмечены открытием новых журналов по применению искусственного интеллекта в сельском хозяйстве, таких как китайский журнал «Искусственный интеллект в сельском хозяйстве» (с 2019 г.). С высоким импакт-фактором первые выпуски журнала содержат подробные обзоры высокого уровня ведущих авторов по использованию ИИ (Искусственный интеллект) в сельском хозяйстве, такие как [5], [6], [7].

Однако практически отсутствуют публикации по раннему выявлению болезней или стресса растений, что наиболее важно для своевременного реагирования и, как следствие, сохранения урожая. Тем более актуальным является использование объяснимого искусственного интеллекта для этой цели.

Методы ХАИ начали постепенно вызывать интерес в связи с обнаружением крупных дефектов в диагностике, построенной на современных методах глубокого обучения. Основными дефектами являются: сложность обработки ошибок ИИ - обработка должна быть быстрой, надежной и без разрушения имеющихся навыков; необъяснимость решений ИИ - сложность перевода неявной логики ИИ в логически явную форму. Методы ХАИ разрабатываются с целью устранения этих дефектов.

Целью данной работы является создание простых ХАИ-блоков для ранней диагностики стрессовых состояний растений. **Объектом** исследования являются методы объяснимого искусственного интеллекта, а **предметом** исследования диагностика стрессовых состояний растений.

В рамках преследуемой цели предлагается разделить исследование на 2 части. Первая часть должна содержать обзор общих и/или проблемно-ориентированных методов извлечения признаков из изображений и моделей машинного обучения на примере задачи классификации болезней растений. Во второй части предлагается протестировать лучшую модель и набор

признаков в задаче ранней диагностики заболевания на примере эксперимента, приближенного к реальности.

Первая часть исследования подразумевает выполнение следующих этапов:

Подготовка данных. Необходимо найти базу данных, на которой будет выполняться тренировка и тестирование классификаторов. Данные должны содержать размеченные изображения как больных, так и здоровых листьев исследуемого сорта растения. Выборка должна быть достаточно большой для получения надежных результатов.

Извлечение и анализ статистических и текстовых признаков. На этом шаге мы уменьшаем размер данных путем получения необходимой информации из изображения. Предполагается, что детектировать заболевание возможно с помощью статистических признаков, но для классификации болезни их может оказаться недостаточно. Поверхность листьев растений достаточно репрезентативна, чтобы различать их заболевания с высокой точностью. Главной характеристикой поверхности является текстура. Наиболее распространенным вариантом получения характеристик текстуры является матрица совместной встречаемости уровней серого тона (GLCM). Для ее вычисления задаются 2 параметра: расстояние d и направление φ . Определение лучших значений параметров d , φ является отдельной подзадачей в этом исследовании.

Тестирование алгоритмов машинного обучения. Для выбора лучшей модели классификации в данной работе будут рассмотрены следующие алгоритмы: линейный дискриминант Фишера, дерево решений, случайный лес, мультиклассовый метод опорных векторов, k -ближайших соседей и одноуровневый персептрон.

Вышеизложенные этапы являются базовыми и обязательными в задаче классификации болезней листьев растений. Качественная реализация каждого этапа позволит получить надежную оценку предлагаемого статистико-текстурного аппарата.

Этапы проведения второй части исследования:

Подготовка данных. База данных должна содержать изображения как больных, так и здоровых растений. Изображения должны быть максимально приближенными к реальности, словно они были получены с беспилотного летательного аппарата или датчиков, установленных на сельскохозяйственных наземных транспортных средствах.

Построение ХАИ-блока. На основе лучшей модели, выбранной в первой части исследования, и предложенного статистико-текстурного аппарата построить ХАИ-блок, оснащенный средствами анализа и визуализации результатов его работы.

Результатом второй части исследования и работы в целом всей работы ожидается эффективный и надежный ХАИ-блок для ранней диагностики стрессовых состояний растений.

2. Постановка задачи

С учетом подхода, описанного в предыдущей главе, для достижения поставленной цели необходимо:

1. Освоить теоретический минимум. А именно:
 - 1.1 Изучить статистический метод исследования текстур на базе GLCM.
 - 1.2 Ознакомиться с наиболее популярными алгоритмами машинного обучения:
 - Линейный дискриминант Фишера
 - Дерево решений
 - Случайный лес
 - Мультиклассовый метод опорных векторов
 - К-ближайших соседей
 - Одноуровневый персептрон
 - 1.3 Ознакомиться с основными метриками в задачах машинного обучения.
2. Подготовить базу данных для обучения и тестирования классификаторов.
3. Определить из каких каналов будут извлекаться признаки.
4. Извлечь статистические и текстурные признаки для всех изображений, на основе которых будет выполняться классификация. Для текстурных признаков определить оптимальное значение параметров GLCM.
5. Исследовать качество работы различных классификаторов на извлеченных данных.
6. Сделать выводы по возможности применения каждого из рассматриваемых классификаторов и признаков для классификации болезней.
7. Подготовить набор изображений, максимально имитирующих реальные условия.
8. На основе лучшей модели, выбранной в п.4 и предложенного статистико-текстурного аппарата в п.3 построить ХАИ-блок, оснащенный средствами анализа и визуализации результатов его работы.
9. Оценить возможности ХАИ-блока в ранней диагностике стрессовых состояний растений на наборе данных п.6.

3. Теоретическая часть

3.1 Матрица GLCM

Любую текстуру можно описать как пространственное распределение значений яркости локальной области изображения с ростом расстояния между оцениваемыми точками. Такое описание реализует матрица совместной встречаемости уровней серого тона - Grey Level Co-occurrence Matrix (GLCM) [8].

Матрица GLCM представляет собой оценку плотности распределения вероятностей второго порядка, полученную по изображению в предположении, что плотность вероятности зависит лишь от расположения двух пикселей. Обозначим эту матрицу $P(i, j, d, \varphi)$, где i и j – яркости соседних точек на изображении, расположенных на расстоянии d друг от друга, при угловом направлении φ . Размер матрицы определяется количеством градаций яркости изображения. Матрица GLCM обычно приводится к одному из двух видов:

- Симметричная матрица:

$$S(i, j, d, \varphi) = P(i, j, d, \varphi) + P(j, i, d, \varphi)^T$$

- Нормализованная матрица (матрица условной вероятности):

$$N(i, j, d, \varphi) = \frac{P(i, j, d, \varphi)}{\sum_{m,k} P(m, k, d, \varphi)}$$

Как только матрицы вычислены, из них извлекаются текстурные признаки. Для этой цели Haralick, Shanmugam и Dinstein [8] предложили 14 мер. Позже Connors и Harlow [9] отметили, что многие из них коррелируют друг с другом и только 5 из этих 14 мер являются достаточными. Далее приведем описание этих мер.

- *Энтропия* (ENT, entropy). Измеряет беспорядок или сложность изображения. Диапазон $(0, +\infty]$. Энтропия велика, когда изображение не является текстурно однородным, а многие элементы GLCM имеют очень маленькие значения. Сложные текстуры, как правило, имеют высокую энтропию. Также энтропия отражает количество информации в изображении: чем больше энтропия, тем больше информации в изображении.

$$ENT = \sum_{ij} P_{ij} * (-\ln P_{ij}) \quad (1)$$

- *Однородность* (НОМ, homogeneity). Измеряет близость распределения элементов GLCM к диагонали. Диапазон $(0, 1]$. Принимает наибольшее значение, если наибольшие записи в GLCM находятся вдоль диагонали матрицы.

$$HOM = \sum_{i,j} \frac{P_{ij}}{1 + (i - j)^2} \quad (2)$$

- *Корреляция* (COR, correlation). Измеряет линейную зависимость интенсивностей пикселей относительно друг друга. Диапазон $[-1, 1]$. Корреляция принимает высокие значения, когда значения в GLCM распределены равномерно, и низкие значения в противном случае. Высокая корреляция текстуры означает высокую предсказуемость отношений пикселей. Корреляция равна NaN для однотонного изображения.

$$COR = \sum_{i,j} P_{ij} * \frac{(i - \mu_i) * (j - \mu_j)}{\sigma_i * \sigma_j} \quad (3)$$

$$\mu_i = \sum_{i,j} i * P_{ij}, \quad \mu_j = \sum_{i,j} j * P_{ij}$$

$$\sigma_i^2 = \sum_{i,j} P_{ij} * (i - \mu_i)^2, \quad \sigma_j^2 = \sum_{i,j} P_{ij} * (j - \mu_j)^2$$

- *Энергия* (ENG, energy). Измеряет текстурную однородность. Диапазон $[0, 1]$. Высокие значения энергии возникают, когда распределение уровня серого имеет постоянную или периодическую форму. GLCM менее однородного изображения будет иметь большое количество маленьких записей. Энергия равна 1 для однотонного изображения.

$$ENG = \sqrt{\sum_{i,j} P_{ij}^2} \quad (4)$$

- *Контраст* (CON, contrast). Определяет локальные изменения интенсивности. Диапазон $[0, (size(GLCM[0])-1)^2]$. Контраст равен 0 для однотонного изображения. Концентрация ненулевых элементов в GLCM вокруг главной диагонали представляет собой низкоконтрастное изображение. При увеличении числа локальных вариаций интенсивностей контраст возрастает.

$$CON = \sum_{i,j} P_{ij} * (i - j)^2 \quad (5)$$

3.2 Описание классификаторов

В работе рассматривается несколько алгоритмов классификации. В данном разделе кратко опишем каждую модель классификатора, переходя от самой простой, к наиболее сложной.

3.2.1 Линейный дискриминант Фишера

Линейный дискриминант Фишера (LDF, Fisher's Linear Discriminant) — это метод машинного обучения, применяемый для нахождения линейных комбинаций признаков, наилучшим образом разделяющих объекты на классы. Дискриминант Фишера предполагает проецирование входных данных на линейное подпространство, возможно меньшей размерности, состоящее из направлений, которые максимизируют разделение между классами.

Само преобразование основано на максимизации соотношения «межклассовой дисперсии» к «внутриклассовой дисперсии» с целью уменьшения вариации данных в одном и том же классе и увеличения разделения между классами. Например, целевая функция для двух классов будет выглядеть так:

$$J(\theta) = \frac{(\hat{u}_2 - \hat{u}_1)^2}{\hat{s}_1^2 + \hat{s}_2^2},$$

где θ – искомая поверхность, \hat{u}_i – проекция среднего вектора i -го класса на направление θ , \hat{s}_i^2 – дисперсия объектов i -го класса после проекции на направление θ . После проецирования вероятность принадлежности образца $x = (x_1, \dots, x_n)$ классу y могут быть получены с использованием наивного правила Байеса:

$$P(y|x_1, \dots, x_n) = \frac{P(y) \prod_{i=1}^n P(x_i|y)}{P(x_1, \dots, x_n)}$$

В случае мультиклассовой классификации y_i для объекта x назначается класс i с максимальной апостериорной вероятностью $P(y_i|x_1, \dots, x_n)$.

3.2.2 Дерево решений

Дерево решений (DTC, Decision Tree Classifier) – модель прогнозирования. На основе входных данных дерево решений прогнозирует значение целевой переменной путем изучения простых правил принятия решений if-then-else. Такие деревья очень схожи с бинарными деревьями поиска.

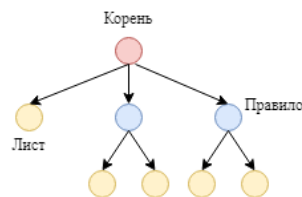


Рисунок 1. Структура дерева решений

В корневом узле располагается множество входных данных, в листовых узлах – значение целевой функции, в остальных узлах – правило перехода, определяющее по какому из ребер идти. Главными параметрами алгоритма принцип формирования правил перехода и критерий остановки. Правило перехода, или правило разбиения данных на подмножества на текущем узле, зачастую определяются из условия минимизации среднего значения энтропии внутри этих подмножеств. То есть атрибут и правило разбиения в узле, выбираются так, чтобы результирующие подмножества включали образцы с одинаковыми метками. Остановка алгоритма как правило задается глубиной дерева.

3.2.3 Случайный лес

В основе метода *случайного леса* (RF, Random Forest) лежит использование ансамбля деревьев решений. На вход каждому дереву поступает некоторое подмножество тренировочных данных, причем эти подмножества для разных деревьев не пересекаются. После чего, на полученной выборке, происходит построение дерева решений. Результатом прогноза ансамбля деревьев решений будет класс, который набрал наибольшее количество «голосов». Главными параметрами алгоритма являются число деревьев, принцип формирования правил перехода и критерий остановки для каждого дерева.

3.2.4 Метод опорных векторов

Основная цель *метода опорных векторов* (SVM, Support Vector Machine) состоит в том, чтобы найти разделяющую гиперплоскость в N-мерном пространстве признаков, которая четко классифицирует эти признаки.

В классической версии SVM – бинарный классификатор. Чтобы разделить два класса точек данных, может быть выбрано множество гиперплоскостей. Искомой разделяющей гиперплоскостью будет гиперплоскость, максимизирующая расстояние до точек данных обоих классов. Чем больше это расстояние, тем меньше будет средняя ошибка классификатора.

Так как в данной работе необходимо классифицировать на несколько классов, то будет использоваться модифицированная версия SVM – *мультиклассовый метод опорных векторов*. Главными параметрами алгоритма являются функция ядра и параметр регуляризации. Функция ядра позволяет строить модели с использованием нелинейных разделяющих гиперплоскостей. Параметр регуляризации позволяет отрегулировать гладкость

классификации объектов обучающей выборки. Низкая гладкость приведет к большой ошибке на тренировочной выборке, высокая к переобучению.

3.2.5 Метод k-ближайших соседей

Метод *K-ближайших соседей* (KNN, K-Nearest Neighbor) - метрический алгоритм для автоматической классификации объектов. Объект присваивается тому классу, который является наиболее распространённым среди k-соседей данного элемента, классы которых уже известны.

Для классификации каждого из объектов тестовой выборки необходимо последовательно выполнить следующие операции:

1. Вычислить расстояние до каждого из объектов обучающей выборки;
2. Отобрать k объектов обучающей выборки, расстояние до которых минимально;
3. Класс классифицируемого объекта — это класс, наиболее часто встречающийся среди k ближайших соседей.

Основными параметрами модели являются выбор числа соседей k, метрика и весовая функция учета близости соседа.

3.2.6 Одноуровневый персептрон

Одноуровневый персептрон (SLP, Single Layer Perceptron) представляет собой полносвязную нейронную сеть с одним скрытым слоем. Структура SLP:

1. Входной слой. Во входной слой поступают исходные данные, и передаются дальнейшим слоям. Размер входного слоя задается размером входного вектора признаков.
2. Один скрытый слой. В нем происходит взвешенное суммирование выходных сигналов предыдущего слоя и формирование выхода посредством нелинейной функции активации. Выбор размера скрытого слоя является одним из главных параметров алгоритма.
3. Выходной слой. Вывод результата. В данной работе выходом будет распределения вероятностей принадлежности к определённому классу. Размер выходного слоя определяется из задачи. Для детектирования на выходе будет 2 узла, для классификации число узлов равно числу классов.

Сигналы между нейронами разных слоев сети передаются через соединения синапсов. У каждого синапса есть 1 параметр — вес. Благодаря ему, входная информация изменяется, когда передается от одного нейрона к другому.

Процесс обучения персептрона состоит в подборе весов модели путем минимизации функции ошибки. В задаче классификации в качестве функции ошибки используется кросс-энтропия. Наиболее эффективным оптимизационным алгоритмом для поиска минимума функции ошибки сейчас является Adam (adaptive moment estimation).

3.3 Метрики качества алгоритмов классификации

Для оценки качества работы классификатора на тестовой выборке, используются различные численные оценки. В простейшем случае такой метрикой может быть доля верных предсказаний *Accuracy*, которая определяется как отношение числа верных предсказаний к числу объектов в тестовой выборке. Однако такая метрика присваивает всем ответам одинаковый вес, вследствие чего ее можно применять только в случае сбалансированных наборов данных.

Для определения качества работы классификатора на несбалансированных данных необходимо использовать точность (*Precision*) и полноту (*Recall*) работы алгоритма:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

где TP - число верных предсказаний, FP - число ложноположительных предсказаний, FN - число ложноотрицательных.

В качестве одной из метрик также можно использовать *F-score*, являющийся средним гармоническим между точностью и полнотой:

$$F\text{-score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

В задачах регрессии для оценки результатов прогнозирования часто используется корень среднеквадратической ошибки модели. Если \hat{y}_i прогнозируемое значение i -го образца, и y_i соответствующее истинное значение, тогда корень среднеквадратической ошибки (RMSE, Root Mean Square Error) на N образцах определяется как:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

Принимает свои значения в диапазоне $[0, \infty)$. RMSE равно 0, когда модель не ошибается в своих прогнозах.

4. Исследовательская часть

4.1 Обзор статистико-текстурного аппарата на RGB изображениях

Работая с RGB изображениями, мы фактически уже работаем с мультиспектральными изображениями. В этом случае интересно определить можно ли ограничиться одним, достаточно информативным каналом.

Пигмент листьев растений, хлорофилл, сильно поглощает видимый свет (от 0,4 до 0,7 мкм) для использования в фотосинтезе. При заболевании образование хлорофилла в листьях нарушается, что приводит к увеличению отражения данных длин волн. Это выражается более явно в красном канале (рис. 2). Поэтому возникает предположение, что признаков, извлеченных из красного канала изображения, окажется достаточно, чтобы с успехом диагностировать заболевание.

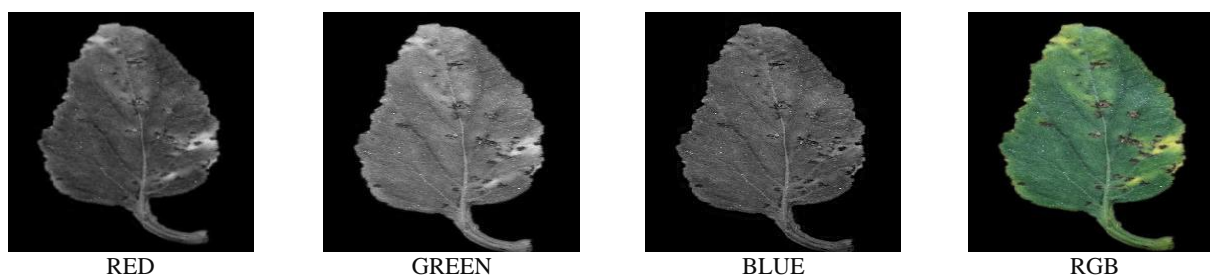


Рисунок 2. Разложение по каналам изображения листа томата, пораженного заболеванием Bacterial spot

Еще одним подкреплением такого предположения служит исследование [10], в котором установлено, что на основе показателя NDVI рассчитанного в красном и инфракрасном диапазоне, можно судить о состоянии здоровья растения. Поэтому, есть дополнительные основания полагать, что красный канал может быть довольно информативным. В данном исследовании также будет предпринята попытка вычислить NDVI образы для всех изображений.

Поскольку признаки заболевания растения проявляются в разных областях листа, то для того, чтобы обеспечить устойчивость показаний признаков к пространственному сдвигу, помимо глобальных признаков, будут рассматриваться и локальные признаки. Под *глобальными (global)* понимаются признаки, извлеченные неким оператором над всем изображением сразу. Под *локальными (local)* признаками понимаются признаки, извлеченные под малой, по сравнению с размерами изображения, маской инструмента.

4.1.1 Описание базы данных

Для тестирования предложенного аппарата диагностики заболеваний растений взята открытая база данных Plant Village [11]. Она включает в себя 54 тыс. сегментированных изображений здоровых и больных листьев для 14 типов растений. В рамках исследования используются только данные о томатах, поскольку информация о них предоставлена в наибольшем объеме. Заболевания томатов подразделяются на 6 классов (рис. 3).

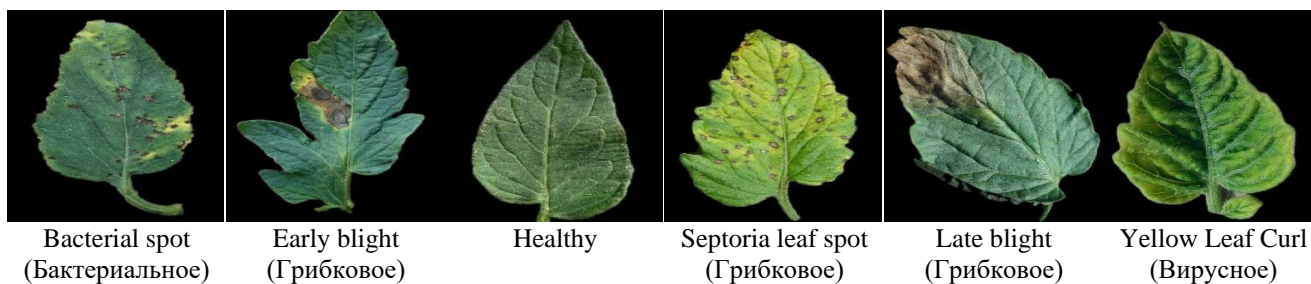


Рисунок 3. Примеры изображений листьев томата каждого класса заболевания

Чтобы сбалансировать классы, решено урезать размер выборок в каждом классе до величины минимальной выборки, то есть до 1000 элементов.

Описание изображений:

- Разрешение: 256x256
- Глубина цвета: 8 бит
- Формат хранения: .jpeg
- Цветовой профиль: RGB

Анализируя изображения на рис. 3, можно видеть, что для классов Septoria leaf spot и Yellow Leaf Curl Virus характерно изменение цвета листа. Для первого оно происходит повсеместно, а для второго только по краям. Классы Bacterial spot, Early blight, и Late blight отличаются возникновением темных пятен. Для Bacterial spot это маленькие многочисленные пятна, для Early blight локальные сильные потемнения, для Late blight более обширные поражения с сильным изменением текстурных признаков. Такой визуальный анализ, позволяет убедиться в необходимости применения статистико-текстурного аппарата диагностики заболеваний.

4.1.2 Обзор изображений в красном канале

Все изображения содержат большую область черного фона. Рассмотрим влияние черного фона на примере изображения листа томата, страдающего Bacterial spot (рис. 4a). На гистограмме (рис. 4b) виден пик для малых интенсивностей. Его наличие зашумляет реальное распределение пикселей листа. На соседнем рис. 4c показано, как пиксели в диапазоне [1, 10]

отражены на изображении. Такое выделение краев вероятно является побочным эффектом сегментации. Поэтому, следует исключить из распределения все «околонулевые» пиксели с интенсивностью <10 .

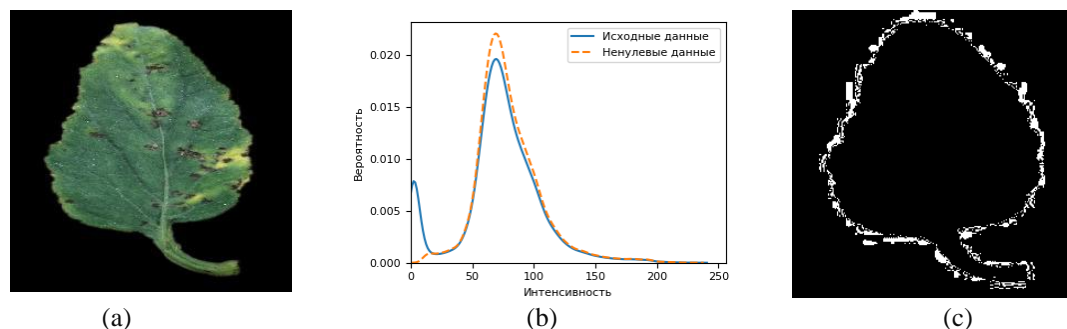


Рисунок 4. (a) Исходное изображение Bacterial spot листа, (b) его гистограмма красном канале и (c) образ пикселей в диапазоне интенсивности $[1, 10]$.

Изменение статистики после удаления фоновых пикселей рассматривается на примере изменений распределений мат. ожидания и стандартного отклонения интенсивностей каждого из классов (рис. 6). Интересным является то, что у группы здоровых растений образовалось 2 четко выраженных пика на гистограмме стандартных отклонений. Они сигнализируют о том, что выборка здоровых растений состоит из 2 типов изображений. Визуальный осмотр подтверждает этот факт. И оказывается, что здоровая выборка содержит 2 разных типа изображений – гладкие и равномерно освещенные листья (рис. 5a) и листья с фактурой и тенью (рис. 5b).

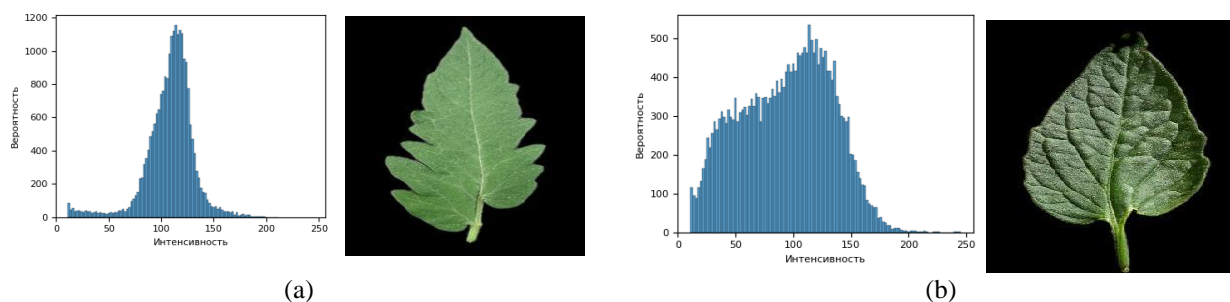


Рисунок 5. Два типа изображений в здоровой выборке и их гистограммы в красном канале

Для проведения исследования решено оставить те изображения, что ближе к реальности – с фактурой и тенью. Отсеивая все изображения, чьи стандартные отклонения меньше 30, мы обрежем распределение по данному признаку. Чтобы этого не допустить добавлена проверка на однородность (2): $\text{НОМ} > 0.4$ (рис. 7). Все пороги отсеивания подобраны из вида распределения соответствующего признака.

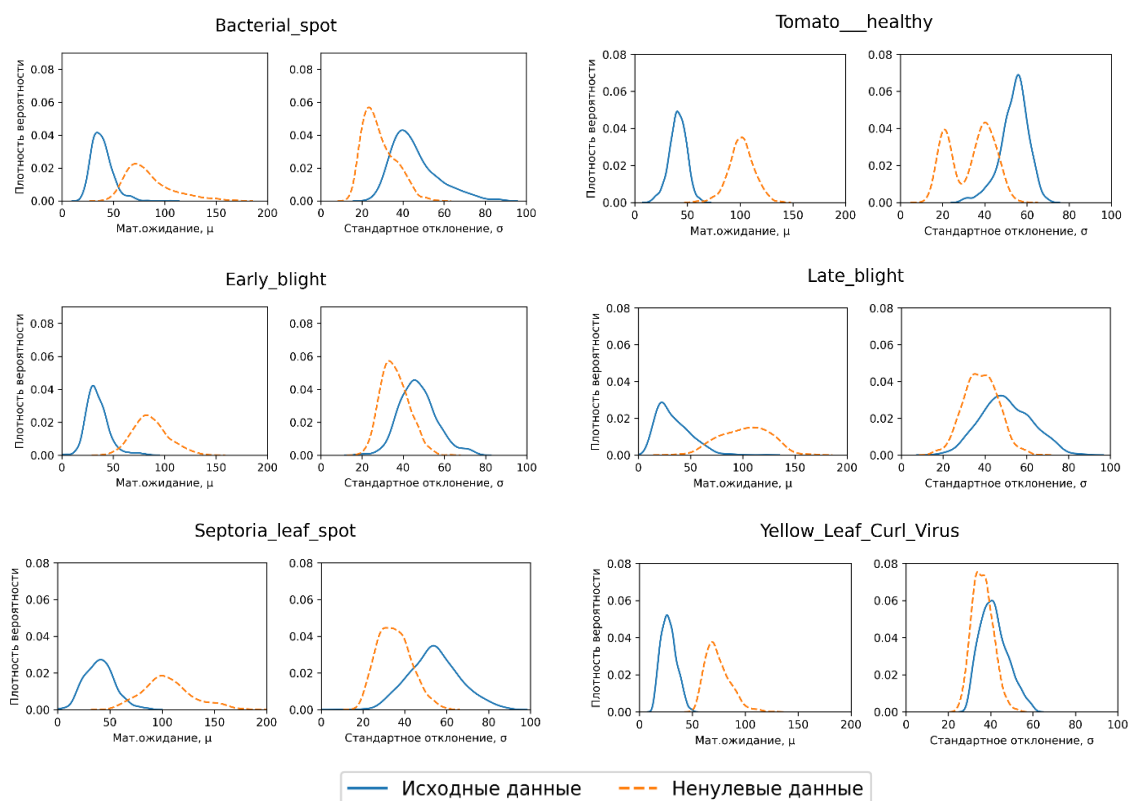


Рисунок 6. Изменение статистики после удаления фоновых пикселей рассматривается на примере изменений распределений мат. ожидания и стандартного отклонения интенсивностей каждого из классов

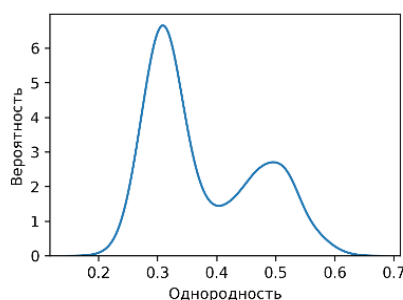


Рисунок 7. Распределение признака однородность НОМ (2) для здоровых растений, без учета фоновых пикселей

4.1.3 Исследование статистических признаков

В качестве статистических признаков изображений используются: математическое ожидание – *mean*, стандартное отклонение – *std*, минимальное – *min* и максимальное – *max* значения, выраженные в единицах стандартного отклонения. Пиксели с интенсивностью меньше 10 не участвовали в извлечении статистики.

Для каждого признака больных и здоровых листьев построены гистограммы (рис. 8). В качестве первого приближения в решении задачи детектирования заболеваний можно воспользоваться информацией из вида распределений извлекаемых признаков, для выделения самых простых для классификации случаев заболевания. Вид мультиклассового распределения статистических признаков доступен на рисунках Приложения А.

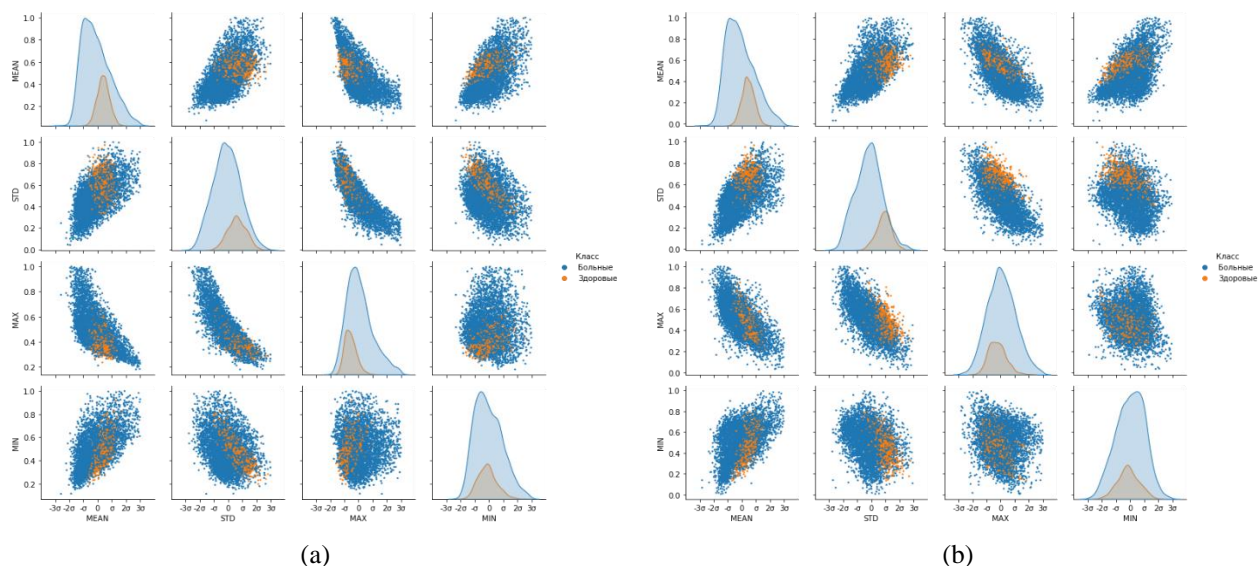


Рисунок 8. Вид распределения статистических признаков больных и здоровых растений, извлеченных глобально (а) и локально (б)

Из полученных гистограмм видно, что распределения этих признаков пересекаются. Но несмотря на это, можно определить «пороги» бинарной классификации для некоторой доли изображений. Т.е. можно сказать, что образец больной, если у него хотя бы один признак выбивается из диапазона распределения здоровых образцов. *Приоритетным* признаком будем называть тот признак, который отсекает как можно больше изображений больных листьев от здоровых. Пользуясь такой логикой, можно определить иерархию признаков.

На этой идее разработан алгоритм работы Priority Decision Tree (PDT) для детекции заболевания, описанный в таблице 1. Пример работы алгоритма показан на (рис. 9).

Таблица 1. Описание PDT

Параметры алгоритма
α – величина смещения в стандартных отклонения от среднего, для определения границ распределений.
Построение дерева решений
Для каждого признака из обучающих данных считается величина приоритета <i>priority</i> : $priority = \text{abs}(\text{healthy}_{\text{left}} - \text{disease}_{\text{left}}) + (\text{disease}_{\text{right}} - \text{disease}_{\text{right}})$ Левые и правые края здоровых и больных распределений определяются как отклонения от среднего на α единиц стандартного отклонения. Далее признаки сортируются по уменьшению <i>priority</i> .
Работа дерева решений
В порядке приоритета для каждого признака образца проверяется попал ли он в диапазон $[\text{healthy}_{\text{left}}, \text{healthy}_{\text{right}}]$. Если попал, то предполагается что растение здорово, переходим к проверке следующего по приоритету признака. Иначе – растение болеет. Выход из алгоритма.

Результаты по детектированию заболевания, полученные с использованием PDT($\alpha=2$) на тестовых статистических данных представлены в таблице 2.

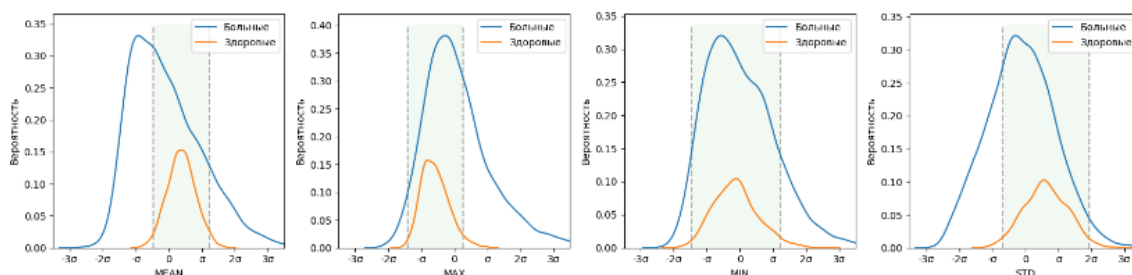


Рисунок 9. Пример работы алгоритма PDT($\alpha=2$) на глобальных статистических данных. Зеленым выделена область, при попадании в которую образец по данному признаку считается здоровым, и переходит к проверке следующего признака. В случае непопадания, выходит из алгоритма со статусом «болен».

Таблица 2. Результаты применения PDT($\alpha=2$) на статистических признаках

Метрика	Масштаб извлечения признаков	
	global	local
Precision	36.1	39.5
Recall	85.2	86.2
F-score	50.0	54.2

Высокий recall убеждает, что большая часть реально здоровых образцов была детектировано верно. Но низкий precision сообщает, что много больных растений помечено как здоровые. То есть на статистических признаках, достаточно часто получалось так, что все признаки больного листа попадали в диапазон значений здоровых листьев. То есть, ограничиваясь только данными признаками, подход PDT не применим. Следует расширить число признаков, чтобы снизить шансы попадания больных признаков в диапазон здоровых.

Исследование статистических признаков показало, что локальные признаки отделяют здоровые образцы лучше глобальных. Использование только статистических данных недостаточно для решения задачи детектирования заболевания алгоритмом PDT. Необходимо расширение вектора признаков новыми данными. В качестве таких данных предлагается взять информацию о гистограмме изображения.

4.1.4 Метод квантования изображений

Извлечение гистограммы, а далее и текстурных признаков из изображений, имеющих 256 градаций серого, требует большой вычислительной работы. Как правило точность признаков, полученных таким способом, является избыточной и сами признаки склонны к зашумленности. Поэтому, для уменьшения вычислительной нагрузки в данной работе решено сжать изображения путем их квантования.

Перед квантованием, все изображения проходили процедуру нормализации:

$$\text{image}_R = \frac{(\text{image}_R - \mu)}{\sigma}, \quad (6)$$

где нижняя подпись R означает красный канал, μ и σ - среднее и стандартное отклонение яркости ненулевых пикселей по выборке здоровых образцов в тренировочном наборе данных.

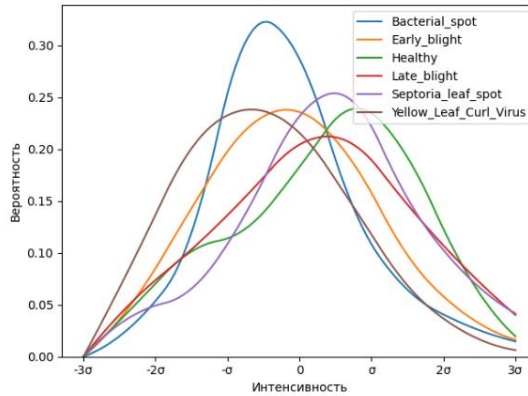


Рисунок 10. Распределение пикселей классов после отбеливания.

Из рис. 10 видно, что как бы мы ни дробили диапазоны квантования, мы не сможем четко отличить классы друг от друга пользуясь только гистограммой. Поэтому, для простоты эксперимента, распознавание будет основано на гистограмме с 4 ячейками шириной σ :

$$Q = \begin{cases} [\mu - 2\sigma, \mu - \sigma), & 0\text{-й бин} \\ [\mu - \sigma, \mu), & 1\text{-й бин} \\ [\mu, \mu + \sigma), & 2\text{-й бин} \\ [\mu + \sigma, \mu + 2\sigma], & 3\text{-й бин} \end{cases} \quad (7)$$

Для пикселей с яркостью, не попавшей в промежуток $[\mu - 2\sigma, \mu + 2\sigma]$, выделяются отдельные бины, которые не учувствуют в классификации, поскольку они не несут полезной информации о различии классов. Предполагается, что квантования на 4 уровня хватит для классификации.

Добавление гистограммы из 4-х бинов все же позволяет внести полезный вклад к статистическим признакам и улучшает качество детектирования заболевания с использованием PDT($\alpha=2$) на 7% (таблица 3). Значит предположение о том, что нужно расширять вектор признаков для улучшения качества детекции на PDT верно.

Таблица 3. Результаты применения PDT($\alpha=2$) на статистических признаках с добавлением гистограммы

Метрика	Масштаб извлечения признаков	
	global	local
Precision	44.1	45.2
Recall	81.3	83.6
F-score	57.5	59.1

4.1.4 Исследование текстурных признаков

В данном разделе исследуется идея о том, что разделить растения на здоровые и больные можно с помощью GLCM-признаков изображения. Для этого строятся симметричные, нормированные матрицы GLCM, со следующими параметрами d и φ :

$$d = \{1px, 4px, 8px\}, \quad \varphi = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\} \quad (8)$$

Такой выбор параметров d обусловлен, тем что на близких расстояниях текстурные признаки сильно коррелированы, поэтому предполагается, что разница на $4px$ обеспечит отсутствие корреляции между признаками при разных d .

GLCM строилась для изображения квантованного на 6 уровней и имела размер 6×6 . Для удаления информации о краевых бинах GLCM обрезалась по краям до размера 4×4 . Такая матрица строилась для каждого направления d и расстояния φ (8) и обладала 5 текстурными характеристиками (1)-(5) – HOM, COR, ENG, ENT, CON. Таким образом, одно изображение характеризуется вектором текстурных признаков длиной в $3 \times 4 \times 5$ элементов.

Для выбора оптимальных параметров d и φ , способных наилучшим образом отделять классы, использовалась информация о косинусном расстоянии между средними векторами глобальных текстурных признаков. Чем меньше угол, тем выше сходство. Для каждого класса вычислен средний вектор глобальных текстурных признаков и найдено среднее значение косинуса с векторами остальных классов при заданной паре параметров (d, φ) (рис. 11).

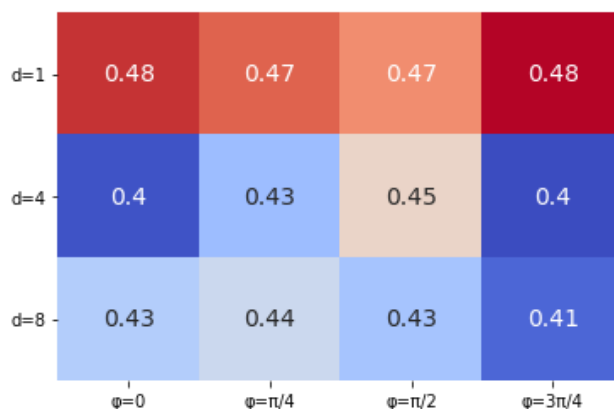


Рисунок 11. Косинусное сходство между средними векторами глобальных текстурных характеристик разных классов

В качестве лучшего расстояния можно выделить $d=4$ (рис. 11). Признаки, рассчитанные при таком значении d , имеют наименьшее косинусное сходство, значит более разнообразно описывают классы. Относительно параметра φ сложно выделить лучшее значение. Косинус слабо различается при разных φ . Поэтому в целях снижения размерности пространства признаков можно рассмотреть вариант усреднения признаков по параметру φ . В зависимости

от выбора набора параметров d , φ рассмотрены 3 варианта векторов текстурных признаков разной длины (таблица 4).

Таблица 4. Описание векторов текстурных признаков разной длины

Обозначение	Значения параметра d	Значения параметра φ	Длина вектора
long	$d = \{1px, 4px, 8px\}$	$\varphi = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$	60
middle	$d = \{4px\}$	$\varphi = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$	20
short	$d = \{4px\}$	$mean[\varphi = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}]$	5

Для текстурных признаков также была протестирована возможность PDT($\alpha=2$) детектирования заболевания. Метрики качества оказались следующими:

Таблица 5. Результаты применения PDT($\alpha=2$) на текстурных признаках

Метрика	Масштаб извлечения признаков					
	global			local		
	short	middle	long	short	middle	long
Precision	39.5	41.2	53.8	39.2	41.1	54.2
Recall	86.0	84.6	82.0	88.6	88.8	85.3
F-score	54.2	55.4	65.1	54.6	64.9	66.3

Для текстурных признаков использование глобальных или усредненных локальных данных не оказало никакого влияния. С точки зрения длины вектора можно заметить, что с увеличением числа признаков растет Precision, как и ожидалось, но падает Recall. Падение Recall происходит за счет того, что PDT отсекает все больше здоровых растений, все признаки которых лежат вне диапазонов соответствующих распределений. Чтобы избежать падения Recall следует расширить диапазоны, но это приведет к падению Precision. Таким образом, PDT не способно решать задачу детектирования, поэтому и с задачей классификации оно не справится. Исследование влияния длины векторов текстурных признаков проведем на более мощных алгоритмах классификации.

4.1.5 Описание полного вектора признаков

Для анализа изображений выбран следующий вектор признаков:

$$ALL = [STAT, HIST, GLCM] \quad (9)$$

составленный из 3-х групп признаков:

1. Статистических характеристик изображения:

$$\text{STAT} = \left[\text{mean}, \text{std}, \frac{\text{max} - \text{mean}}{\text{std}}, \frac{\text{mean} - \text{min}}{\text{std}} \right], \quad (10)$$

где mean, std, max, min – среднее, стандартное отклонение, максимальное и минимальное значения пикселей изображения. Максимальное и минимальное значения подвергаются стандартизации.

2. Нормированной гистограммы квантованного изображения Q:

$$\text{HIST} = [N(Q_i)], \quad i = \overline{0,3}, \quad (11)$$

где $N(Q_i)$ - нормированные значения уровня Q_i (7) изображения.

3. Текстурных GLCM признаков квантованного изображения Q:

$$\text{GLCM} = [\text{CON}, \text{HOM}, \text{ENG}, \text{COR}, \text{ENT}]_{d,\varphi} \quad (12)$$

GLCM-матрица вычисляется для каждого угла и расстояния квантованного изображения имеет размер 4x4. Всего в работе рассмотрено 3 расстояния и 4 направления (8). Поэтому каждый из 5 текстурных признаков (1)-(5) представлен 12 числами.

Таким образом, полный вектор ALL описывается 4-мя статистическими признаками, 4-мя бинами гистограммы и 60 текстурыми признаками. В данной работе также будет исследована возможность использования укороченного вектора GLCM признаков. Для описания длины вектора признаков будут использоваться обозначения short, middle и long согласно таблице 4. В итоге, длина вектора ALL может принимать значения 13, 28 и 68 элементов.

4.1.6 Алгоритм извлечения вектора локальных признаков

Для вычисления локальных признаков размер маски был взят равным 17x17. Алгоритм извлечения локальных признаков ALL (9) для конкретного изображения пошагово описан в таблице 6. Блок операций 2 выполняется по всему изображению параллельно.

На рис. В1 и В2 Приложения В изображены примеры исходного изображения здорового и больного Bacterial spot листьев томатов, а рядом с ними результаты извлечения 68 признаков под маской инструмента. Данные изображения возможно получить в п. 3.3 таблицы 6 выполнения модели извлечения локальных признаков из оригинальных изображений. Листинг кода алгоритма извлечения признаков доступен в Приложении Н.

Таблица 6. Алгоритм извлечения признаков

Шаг	Действие	Размерность
1. Подготовка данных и вычисление статистических характеристик STAT (10)		
1.1	Считывание RGB-изображения.	[3,256,256]
1.2	Извлечение информации из канала R.	[1,256,256]
1.3	Паддинг изображения для подготовки к операции скользящего окна.	[1,257,257]
1.4	Построение массива скользящих окон 17x17 с шагом stride=4 для изображения.	[1, 61, 61, 17, 17]
1.5	Вычисление характеристик STAT (10) в каждом окне.	[4, 61, 61]
1.6	Квантование изображения (7). Получение квантованного изобр. Q.	[1, 61, 61, 17, 17]
1.7	Изменение размера Q.	[3721, 17, 17]
2. Вычисление характеристик HIST (11) и GLCM (12) в каждом из 3721 окон		
2.1	Вычисление нормированной гистограммы HIST (11) для Q.	[4]
2.2	Вычисление нормированной, симметричной GLCM-матрицы.	[N, N, 3, 4]
2.3	Вычисление признаков GLCM (12)	[5,3,4]
2.4	Уплощение матрицы признаков GLCM	[60]
2.5	Конкатенация гистограммы HIST с шага 2.1 и признаков GLCM с шага 2.4.	[64]
3. Формирование результирующего вектора признаков по данному изображению		
3.1	Окончание операций под масками инструмента.	[3721, 64]
3.2	Приведение формы матрицы признаков, полученной на шаге 3.1, к размеру с шага 5 и её транспонирование.	[64,61,61]
3.3	Конкатенация STAT, HIST, GLCM	[68,61,61]
3.4	Усреднение локальных признаков по всем скользящим окнам.	[68]

4.1.7 Выбор оптимальных параметров классификаторов

Одной из задач работы являлась проверка возможностей различных алгоритмов классификации на одном и том же наборе данных с целью выбрать наилучший из них. В данном разделе описываются оптимальные параметры классификаторов и метод их подбора.

Для изучения были выбраны следующие алгоритмы: линейный дискриминант Фишера (LDF), дерево решений (DTC), метод k-ближайших соседей (KNN), случайный лес (RF), метод опорных векторов (SVM) и одноуровневый персептрон (SLP).

Оценка параметров моделей проводилась с помощью поиска по сетке с применением k-fold валидации (k=5). То есть для каждой конкретной модели брались возможные значения параметров, и в сетке параметров каждая их комбинация тестировалась k-fold валидацией. В конце выбирался набор параметров, давший наилучший результат при валидации. Валидация проводилась на полном векторе глобальных признаков ALL (9).

Результаты такого тестирования по сетке параметров для каждого классификатора представлены в Приложении С. Выбранные параметры классификаторов доступны в таблице 7.

Таблица 7. Параметры алгоритмов классификации

Название метода	Параметры для scikit-learn
LDF	-
DTC	criterion='entropy', max_depth=9
KNN	n_neighbors=3, metric='euclidean', weights='distance'
RF	n_estimators=80, criterion='entropy', max_depth=15
SVM	C=10, kernel='rbf', gamma='auto'
SLP	hidden_layer_sizes=50, activation=Relu, activation_function_of_output_layer=None, loss_function= CrossEntropyLoss, optimizer = Adam, learning_rate=1.e-3, epoch=100, batch_size=100

4.1.8 Классификация на глобальных и локальных данных

Исследуемые алгоритмы классификации применялись как ко всем извлечённым признакам ALL (9), так и к их частям: STAT (10), HIST (11) и GLCM (12). Эти признаки брались нормализованными по данным тренировочной выборки. Дополнительно рассматривается возможность использования укороченного вектора GLCM признаков. Для описания длины вектора используются обозначения short, middle и long (таблица 4). Исследуется два способа извлечения признаков: глобальный (global) и локальный (local). В Приложениях D-E представлены таблицы с результатами бинарной и мультиклассовой классификации на основе признаков из красного канала. Анализ результатов позволяет сделать следующие выводы:

- Дерево решений DT показало один из худших результатов. Значит, на рассматриваемых признаках нельзя создать простое правило классификации, основанное на используемых статистико-текстурных измерениях.
- Сравнимые результаты дали KNN, RF и LDF. Невысокая точность RF, объясняется тем, что RF – объединение результатов множества деревьев решений, а само дерево решений показало низкий результат. В случае KNN, можно предположить, что границы классов сильно соприкасаются друг с другом, поэтому «голоса» соседей не дают надежной информации. Высокие результаты линейного дискриминанта Фишера LDF подтверждают существование «хорошей» проекции признаков, которая позволяет классифицировать объекты исходя из вида распределений при помощи наивного Байеса.
- Стабильно лучшие результаты показывали одноуровневый перцептрон SLP и метод опорных векторов SVM над полным long вектором признаков ALL. В задаче детектирования достигнут 95% F-score, классификации – 83%. Это говорит о том, что

переход в новое пространство, большей размерности позволяет построить разделяющую гиперплоскость. А в случае метода опорных векторов удастся построить нелинейную гиперплоскость, удачно разделяющую классы.

- Алгоритмы классификации на основе характеристик STAT и STAT+HIST, показывают низкое качество. Текстурные характеристики GLCM вносят заметный вклад в улучшение работы алгоритмов.
- Использование векторов разной длины (short, middle и long) не влияет на точность детектирования, но на классификации точность может варьироваться в пределах 10% на полном векторе признаков ALL. Значит для детектирования достаточно short вектора, но для классификации предпочтительнее использовать long вектор.
- Использование локальных признаков до 7% улучшает качество классификации. Отчетливо видно улучшение на укороченных short и middle векторах.
- Классификация на локальных признаках STAT+HIST приобретает до 10% качества относительно классификации на глобальных признаках STAT+HIST.
- При классификации только на признаках GLCM способ извлечения признаков (глобальный или локальный) не оказывает влияния на итоговую точность.

На основе последних двух пунктов, можно сказать, что глобальные признаки проигрывают в качестве за счёт потерь на признаках STAT+HIST. Поэтому ставится предположение что можно повысить точность классификации путем использования локальных статистических и глобальных текстурных признаков. Такой вектор комбинированных признаков обозначим *comb*.

Результаты эксперимента по применению комбинированных признаков *comb* представлены в Приложениях D-E. Видно, что наше предположение верно и показатели F-Score для комбинированного вектора *comb* и для вектора локальных данных *local* оказались идентичными.

За счет отсутствия упадка качества работы классификаторов на комбинированном векторе признаков, становится возможным использования вместо локальных GLCM признаков их глобальные аналоги. В связи с этим уменьшится вычислительная нагрузка извлечения признаков из исходных изображений. Время обучения лучших классификаторов не превышает 6 секунд, а инференса 2 секунд (таблица 8).

Таблица 8. Время обучения и инференса лучших классификаторов на векторе комбинированных признаков, полученных в красном канале

Классификатор	Тренировка, с			Инференс, с		
	short	middle	long	short	middle	long
SLP	5.17	5.46	5.79	1.18	1.27	1.39
SVC	0.36	0.42	0.52	1.12	1.21	1.33

Лучше всех подвергаются классификации здоровые листья, листья, пораженные заболеваниями *Yellow Leaf Curl Virus* и *Bacterial spot* (рис 13). *Early blight* и *Septoria* являются самыми трудным для классификации, их Precision составляет 69%. Часто происходит путаница *Early blight* и *Late blight*. Многие изображения *Septoria* классифицируются как *Late blight*.

Действительные метки	Bacterial_spot	93%	1%	0%	1%	5%	0%
	Early_blight	3%	69%	1%	17%	4%	4%
	Healthy	0%	3%	89%	2%	6%	0%
	Late_blight	1%	15%	0%	75%	8%	1%
	Septoria	6%	8%	3%	14%	69%	0%
	Yellow_Virus	0%	3%	0%	0%	0%	97%
		Bacterial_spot	Early_blight	Healthy	Late_blight	Septoria	Yellow_Virus
		Предсказанные метки					

Рисунок 13. Матрица ошибок (%) одноуровневого персептрона SLP на полном векторе комбинированных признаков полученных в красном канале

Таким образом, в данном разделе удалось без потери качества классификации оптимизировать процесс извлечения признаков заменой локальных текстурных признаков, глобальными.

4.2 Исследование применения NDVI_G аналога

В данной работе исследуется возможность классификации болезней листьев томата. Для решения такой задачи целесообразно использовать характерные для растительности индикаторы заболевания. Одним из таких показателей является NDVI – нормализованный вегетационный индекс, по которому можно судить о количестве и качестве растительности на участке поля во время вегетации. В данном разделе будет предпринята попытка вычислить NDVI образы для всех изображений.

NDVI рассчитывается в красном и инфракрасном диапазоне следующим образом:

$$NDVI = \frac{NIR - RED}{NIR + RED},$$

Расчет NDVI базируется на двух участках спектральной кривой отражения сосудистых растений (рис. 14). В красной области спектра (0,6-0,7 мкм) лежит максимум поглощения солнечной радиации хлорофиллом, а в инфракрасной (0,7-1,0 мкм) находится область максимального отражения клеточных структур листа. То есть высокая фотосинтетическая активность ведет к меньшему отражению в красной области спектра и большему в

инфракрасной. Использование нормализованной разности между минимумом и максимумом отражений увеличивает точность измерения, позволяет уменьшить влияние таких явлений как различия в освещенности снимка, облачности, дымки, и пр.

4.2.1 Расчет $NDVI_G$ образов изображений

Поскольку мы работаем с RGB изображениями, то в чистом виде информации об этом инфракрасном диапазоне нет. Вместо этого мы обладаем информацией о зеленой области спектра (0.5-0.6 мкм). Поскольку растения отражают зеленый свет сильнее красного (рис. 14) предлагается ввести аналог вегетационного индекса на основе зеленого и красного каналов.

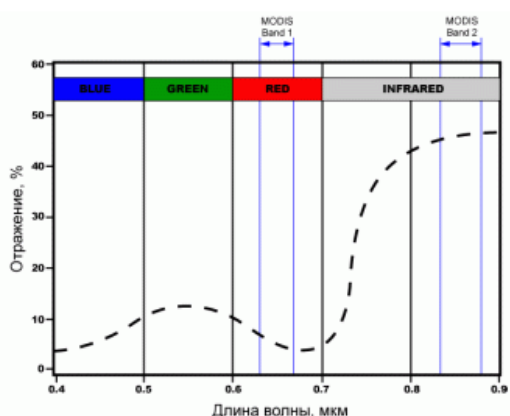


Рисунок 14. Отражение растениями разных длин волн [12]

Аналог $NDVI$, основанный на красном и зеленом каналах обозначим как $NDVI_G$:

$$NDVI_G = \frac{GREEN - RED}{GREEN + RED} \quad (13)$$

После вычисления $NDVI_G$ образа изображение проходило процедуру отбеливания (центрирования и стандартизация). Проанализируем примеры $NDVI_G$ изображений для здорового листа томата, и листа, страдающего заболеванием Bacterial Spot (рис. 15-16) с точки зрения физики.

Поскольку пигмент листьев растений, хлорофилл, сильно поглощает видимый свет, RED изображение должно иметь более темные участки в здоровых, «живых», областях растения, и более светлые участки в больных, «омертвевших», областях. С другой стороны, клеточная структура листьев сильнее отражает свет в зеленой области спектра. Поэтому GREEN изображение должно быть светлее RED, там, где растение здорово. В итоге $NDVI_G$ должно описывать здоровые области светлыми участками, «омертвевшие» – темными.

Полученные изображения (рис. 15-16) корректно описывают физические явления. Как и ожидалось на $NDVI_G$ изображениях область заболевания описывается темными участками, то есть малыми значениями индекса $NDVI_G$.

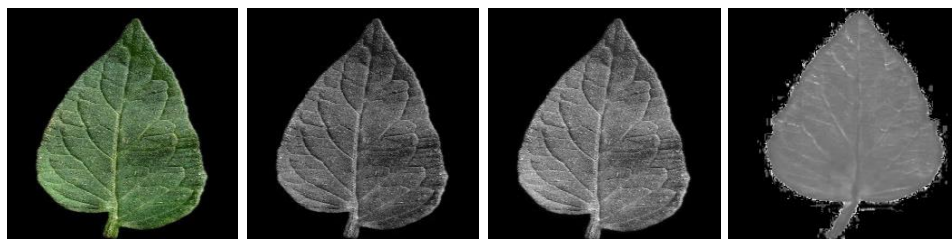


Рисунок 15. RGB, RED, GREEN и NDVI_G изображения здорового листа томата

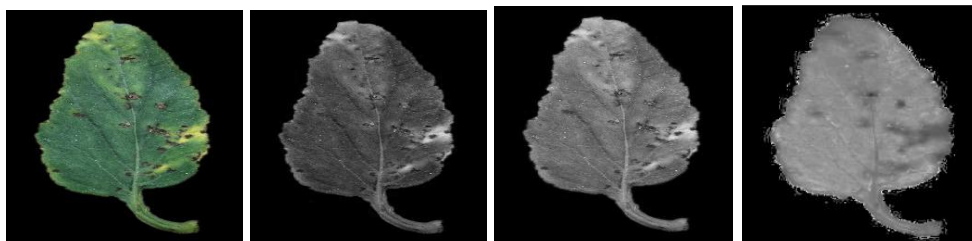


Рисунок 16. RGB, RED, GREEN и NDVI_G изображения больного Bacterial spot листа томата

4.2.2 Результаты использования вектора NDVI_G признаков

Аналогично тому, как извлекались признаки из красного канала, были извлечены статистические и текстурные признаки ALL (9) из NDVI_G изображений. Результаты бинарной и мультиклассовой классификации доступны в Приложениях F-G.

Задача детектирования на NDVI_G признаках решается на 100% как для глобальных, так и для локальных данных. Все классификаторы отлично справляются с этой задачей. Уже на детектировании можно заметить, что глобальные текстурные признаки, работают лучше усредненных локальных. А локальные статистические лучше глобальных. Поэтому возникает необходимость комбинировать признаки.

В задаче классификации снова лучшими оказались одноуровневый персептрон SLP и метод опорных векторов SVM. Точность классификации достигает 86% на локальных данных при полном long векторе данных ALL. Если использовать самый короткий вектор short, то точность классификации будет не ниже, чем при классификации на long векторе признаков из красного канала. И снова можно заметить необходимость в комбинировании локальных статистических и глобальных текстурных данных.

При классификации на комбинированных NDVI_G признаках удастся повысить точность на 5-7% для всех классификаторов, относительно классификации на RED признаках. И теперь максимальный F-score составляет 87% на полном векторе признаков ALL для SLP модели. Время обучения лучших классификаторов не превышает 6 секунд, а инференса 2 секунд (таблица 9).

Таблица 9. Время обучения и инференса лучших классификаторов на векторе комбинированных признаков NDVI_G

Классификатор	Тренировка, с			Инференс, с		
	short	middle	long	short	middle	long
SLP	4.17	5.46	5.79	1.34	1.34	1.44
SVC	0.36	0.42	0.52	1.33	1.32	1.41

Как и в случае красного канала, лучше всех подвергаются классификации здоровые листья, листья, пораженные заболеваниями *Yellow Leaf Curl Virus* и *Bacterial spot* (рис. 17). Точность их выявления не менее 92%. *Early blight* и *Septoria* являются самыми трудными для классификации.

Действительные метки	Bacterial_spot	92%	2%	0%	0%	6%	0%
	Early_blight	7%	72%	0%	13%	7%	1%
	Healthy	0%	0%	100%	0%	0%	0%
	Late_blight	1%	13%	0%	78%	7%	0%
	Septoria	4%	14%	0%	11%	71%	0%
	Yellow_Virus	0%	0%	0%	0%	0%	100%
		Bacterial_spot	Early_blight	Healthy	Late_blight	Septoria	Yellow_Virus
		Предсказанные метки					

Рисунок 17. Матрица ошибок (%) одноуровневого персептрона SLP на полном векторе комбинированных признаков NDVI_G

Таким образом, NDVI_G признаки лучше решают поставленные задачи, чем признаки, извлеченные в красном канале. Задачу детекции удастся решить на все 100%, а классификации на 87% F-score. Такие высокие результаты в целом подтверждают эффективность предложенного статистико-текстурного аппарата. Поэтому предлагает протестировать возможность их использования в качестве интерактивных XAI-блоков.

4.3 Построение XAI-блока в эксперименте на реальных условиях

До этого момента рассматривались игрушечные изображения растений – крупно отснятые листья томата без фона, и выровненные вертикально. На таких хороших данных предложенный статистико-текстурный аппарат показал высокие результаты. В действительности такие изображения сложно получить беспилотными летательными аппаратами или статически установленными камерами. Поэтому в данном разделе пойдет речь о применении нашего аппарата в условиях, приближенных к реальности. Как мы убедились ранее, лучшей моделью классификации на таких признаках оказался одноуровневый персептрон SLP. Далее в работе будет использоваться только он. Также использование признаков NDVI_G на базе зеленого и красного каналов показало лучшие

результаты, по сравнению с признаками извлеченными из одного red канала. Поэтому далее будет использоваться такая нормированная разность. Также отметим, что разница в точности классификации на локальных и глобальных данных оказалась незначительной. Поэтому далее будет рассматривать только глобальные данные.

4.3.1 Подготовка набора данных

Исследования проведены на растениях пшеницы (*Triticum aestivum* L.) сорта Злата. После замачивания семена высаживали в горшки с грунтом. Растения выращивали до 14-дневного возраста, после чего их разделили на контрольные и опытные. Данные по засухе пшеницы были получены за 25 дней с регулярным интервалом фиксации состояния, равным 2-3 дням. В опытной группе полив прекратили, в контрольной продолжили по стандартному графику. Изображения контрольных и опытных растений были получены с помощью трех устройств:

- 1) RGB-камера Nikon D5100 DSLR (Nikon Corporation, Япония) (далее датчик RGB)
- 2) Тепловизор Testo 885-2 (Testo SE & Co, KGaA, Германия) (далее датчик TIR)
- 3) Гиперспектральная камера Specim IQ: диапазон: 400-1000 нм, спектральное разрешение: 7 нм, каналы: 204; 512x512 pix (далее датчик HSI).

Изображения были получены при разной ориентации датчиков по отношению к объектам: 1) 90° (вид сверху), соответствует получению изображений датчиками, установленными на БПЛА; 2) 45° (вид сверху), соответствует получению изображений датчиками, установленными на сельскохозяйственных наземных транспортных средствах. Снимки получали на разном расстоянии между датчиком и объектом - 30см (масштаб отдельных растений) и 100см (масштаб площади поля). Количество изображений, полученных для набора базы знаний, показано в таблице 10.

Таблица 10. Число изображений контрольных и опытных растений

«Фрагмент поля» (боксы по 30 горшков)						«Растение» (2 горшка)		
Число изображений = 33 (3 бокса * 11 дней)						Число изображений=135		
Позиция датчика: 45°			Позиция датчика: 90°			Позиция датчика: 45°		
RGB	HSI	IR	RGB	HSI	IR	RGB	HSI	IR
33	33×204	33	33	33×204	33	165	165×204	165

На рис. 18 и 19 представлены изображения контрольных (слева) и опытных (засуха, справа) растений пшеницы с датчиками RGB и TIR под углом 90° над «фрагментом поля» на 16-е сутки после прекращения полива опытных растений.



Рисунок 18. Пример RGB-изображения 1-го бокса под углом 90°, 16-е сутки

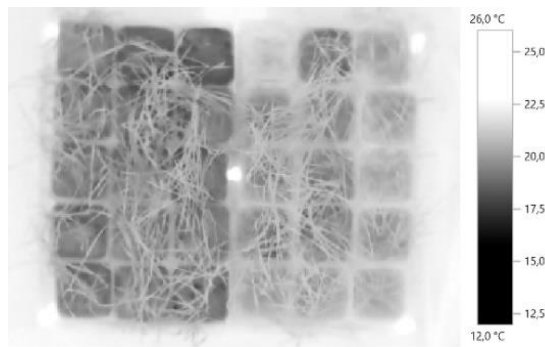


Рисунок 19. Пример TIR-изображения 1-го бокса под углом 90°, 16-е сутки

В дни фиксации изображений измеряли температуру листьев и разницу в содержании воды у контрольных и опытных растений. Графики представлены на рис. 20. Из графиков видно, что заметное изменение содержания воды в растении появляется только на 11-й день засухи. Но по температуре листа (и, следовательно, по TIR-изображению) можно заметить начало стрессового состояния растения на 5-й день и полить его так, чтобы стресс не привел к усыханию растения.

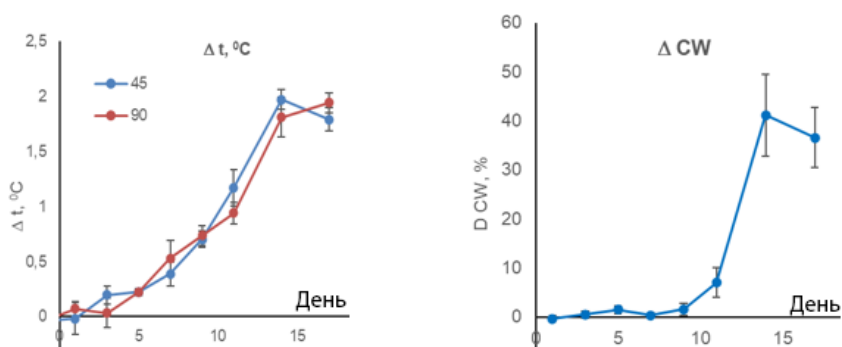
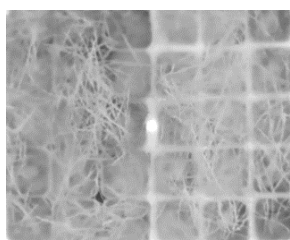


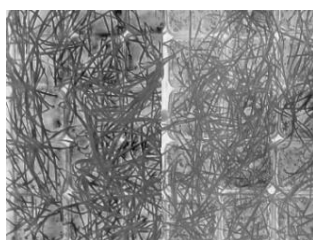
Рисунок 20. Зависимость от дня с момента прекращения полива разницы температур опытных и контрольных растений пшеницы, регистрируемой TIR-датчиком под углом 45 и 90 (слева). Зависимость разницы содержания воды контрольных и опытных растений пшеницы (справа)

4.3.2 Предварительная обработка изображений

Рассмотрим изображения TIR, поскольку они способны к раннему обнаружению стресса на основе оценки аспирации растений, и соответствующие изображения RGB как самые дешевые и имеющие самое высокое разрешение (рис. 21).



TIR (12-26° C)



NDVI_T



RGB

Рисунок 21. Мультисенсорное представление образца на 10-е сутки без полива.

Как обсуждалось в главе 4.2 «Исследование применения NDVI аналогов» по нормированному разностному индексу растительности (NDVI) можно судить о состоянии, здоровья и засухи растений. В данном разделе был построен аналог NDVI на более широких каналах RED и Thermal IR (TIR), но с аналогичным относительным поведением. Итак, аналог NDVI на базе TIR будет следующим:

$$NDVI_T = \frac{TIR - RED}{TIR + RED} \quad (14)$$

Исследуемый диапазон изменения значений TIR соответствует диапазону температур 12-26 градусов. Используя (14), было преобразовано TIR-изображение (рис. 21, слева) вместе с R-каналом RGB-изображения (рис. 21, справа) в изображение NDVI_T (рис. 21, посередине). Все значения NDVI_T были центрированы и нормализованы. Изображение (рис. 21, посередине) представляется сразу в отбеленном виде, и мы видим, что особенности высыхания заметны визуально.

Пример гистограммы NDVI_T-изображения для 1-го из 10-дневных боксов после такой предварительной обработки показан на рис. 22. Интересно также наблюдать динамику изменения гистограммы от дня к дню (рис. 23). Из рис. 22 и 23 также можно видеть, что распознавание стрессового состояния растения может быть основано на гистограмме с 4 ячейками шириной std (7). Значения в ячейках мы можем использовать как часть вектора характеристик стресса растений.

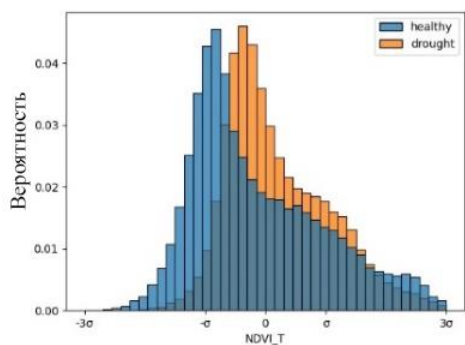


Рисунок 22. Гистограмма для здоровой половины(healthy) и половины стресса от засухи (drought) изображения NDVI_T для 10-дневного образца, показанного на рис. 21

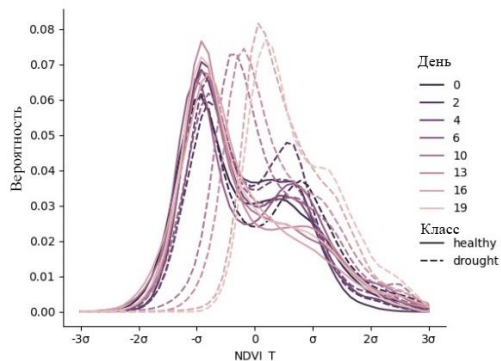


Рисунок 23. Изменение гистограмм NDVI_T изображений с 1-го по 20-й день без полива. Виден дрейф вправо на гистограмме засушливых(drought) растений, который, вероятно, начался между 6 и 10 днями.

TIR и R-канал из RGB-изображений были преобразованы в ежедневную статистику эксперимента для NDVI_T-изображений (рис. 24). Такое преобразование дает нам возможность перейти к вектору признаков NDVI_T-изображения вместо самих изображений.

Увеличение площади засухи со временем (рис. 24а) указывает на то, что все больше пикселей изображения NDVI_T попадают в соответствующий диапазон гистограммы. Это означает, что NDVI_T чувствителен к изменениям во время стресса. Мы также можем видеть

линейный рост разницы между значениями $NDVI_T$ для стрессовых и контрольных образцов, полученных в один и тот же день (рис. 24с). Это гарантирует возможность разделения стрессовых и контрольных образцов на ранних стадиях засухи (на 4-6 сутки без полива). Последующий анализ изображений $NDVI_T$ позволит проверить правильность использования такого индекса для решения задачи раннего выявления стресса пшеницы от засухи.

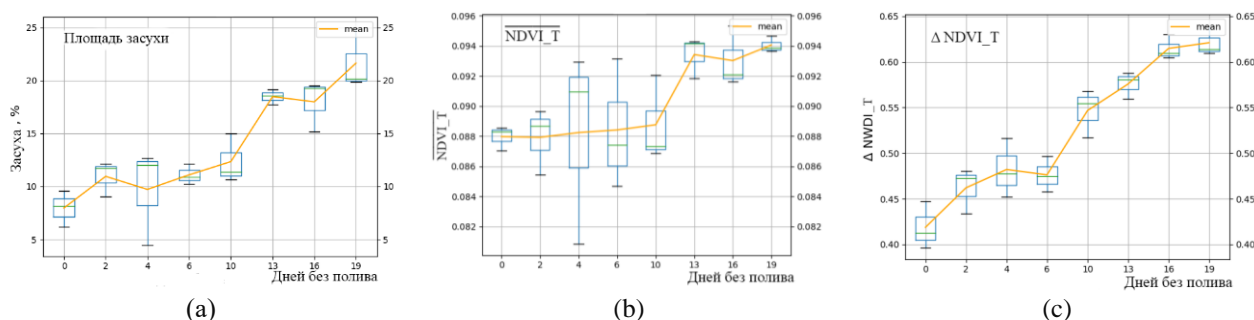


Рисунок 24. (а) Динамика увеличения площади засушливых почв и растений на основе сегментации данных $NDVI_T$. (б) Динамика изменения среднего значения $NDVI_T$ опытной зоны. (с) Динамика разницы $NDVI_T$ между контрольными и засушливыми растениями.

4.3.3 Анализ результатов эксперимента

Проблема раннего обнаружения засухи исследуется здесь с использованием однослойного перцептрона SLP (рис. 25). SLP-классификатор далее будем обозначать его как $SLP_C(N)$, где N - количество нейронов в скрытом слое. F-score используется в качестве показателя качества. Обучение $SLP_C(N)$ выполняется на основе трех групп признаков: STAT (10), HIST (11), GLCM (12).

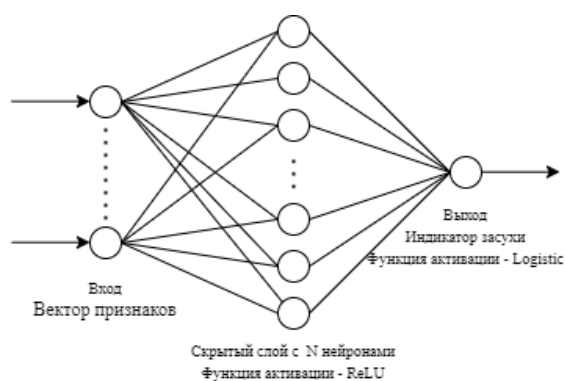


Рисунок 25. Архитектура $SLP_C(N)$ детектора засухи

Весь набор данных состоял из 33 изображений для 90 горшков с растениями пшеницы в 3 ящиках. Размер обучающей и тестовой частей составляет 80% и 20% соответственно.

Для формальной оценки состояния пшеницы и построения вектора признаков использовались: статистические характеристики изображения (10), L параметров квантованной гистограммы изображения (L - число бинов) (11), текстурные признаки 5×4 на

основе GLCM ($\phi=\{0^\circ, 45^\circ, 90^\circ, 135^\circ\}, d=1$) (12). Четырехуровневое квантование изображения (7) позволяет вычислять малолучающие гистограммы и GLCM-матрицы для изображений NDVI_T. Все уровни находятся внутри промежутка $[-2\sigma, 2\sigma]$ относительно μ , и нормализованы на этот интервал для преобразования изображения в оттенки серого $[0.0, 1.0]$ или $[0, 255]$ с целями диагностики и визуализации.

Нельзя полностью быть уверенным, что квантование гистограммы с 4 ячейками, как и квантование 4 уровней серого для GLCM, является оптимальным решением для точности классификации. Например, шаг размером в половину стандартного значения может оказаться более успешным. И в процессе эксперимента будет проанализировано влияние количества уровней квантования на точность результата.

Сначала было выполнено исследование эффективности детекции засухи с использованием отдельных признаков и групп признаков изображений NDVI_T (рис. 26). Это было выполнено с использованием SLP_C(3) в качестве инструмента анализа.

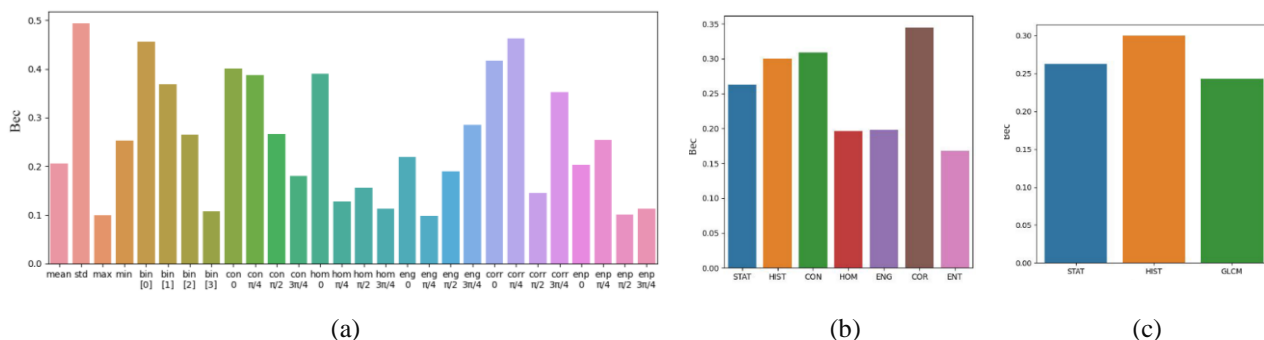


Рисунок 26. NDVI_T графики. Веса, полученные после обучения детектора засухи SLP_C(3).
 (a) Значения веса для каждого элемента вектора признаков. Вектор признаков включает 5 скаляров (1)-(5) на основе GLCM для каждого из 4 углов поворота. (b) Значения весов для групп признаков STAT, HIST и, отдельно, 5 скаляров на основе GLCM. (c) Значения весов для различных групп STAT, HIST и GLCM.

Следующим шагом в определении эффективности SLP_C(N) было изучение ее зависимости как от числа нейронов (N), так и от различных групп признаков (рис. 27).

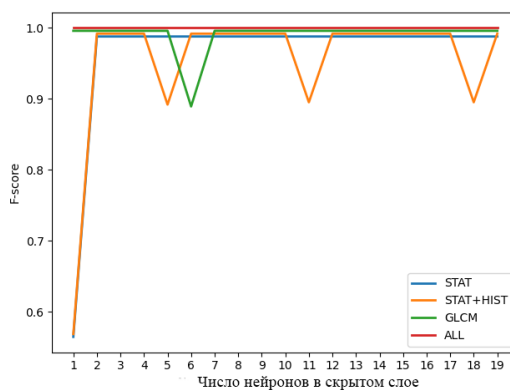


Рисунок 27. NDVI_T график. Зависимость эффективности SLP_C(N) как от N, так и от использования разных групп признаков

Мы видим, что для изображений $NDVI_T$ F-score, равный 1, достигим для любого количества нейронов. При $N=1$ для этого достаточно использовать только группу текстурных признаков GLCM. В целом все группы признаков работают хорошо и примерно эквивалентны в рассматриваемых экспериментальных условиях. $N = 2,3,4$ - минимальное количество нейронов, при котором любая из трех групп признаков обеспечивает максимальную эффективность.

Такой успех $NDVI_T$ заставляет задуматься об использовании аналогичного индекса, только на этот раз в пределах одного спектрального диапазона: видимого спектра (RGB-изображения). Для этого предлагается ввести аналог индекса растительности на основе зеленого и красного каналов RGB-изображений (13). Пример изображения $NDVI_G$ для одного из ящиков (слева) и изменения $NDVI_G$ -гистограммы с 1-го по 20-й день без полива показаны на рис. 28.

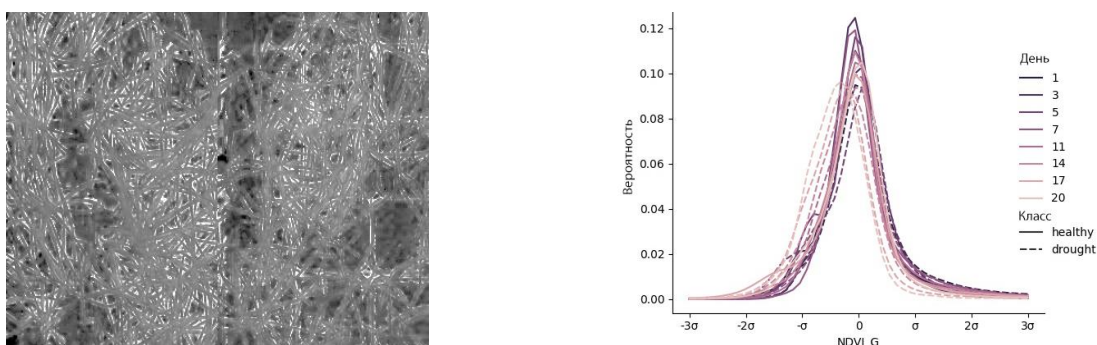


Рисунок 28. $NDVI_G$ графики. Слева - изображение $NDVI_G$ одного из ящиков (см. рис. 21, справа); Справа - изменение гистограмм $NDVI_G$ изображений с 1-го по 20-й день без полива.

Гистограмма $NDVI_G$ более компактна, чем $NDVI_T$, но у нее также есть дрейф в сухой половине поля, только влево, а не вправо, и заметно меньше по величине. Мы также видим, что разница между засухой и нормой визуально не определяется. Несмотря на это, обучение $SLP_C(N)$ на признаках $NDVI_G$ показывает хороший результат. Для изображений $NDVI_G$ рассматривается зависимость эффективности $SLP_C(N)$ как от N , так и от использования различных групп признаков для того же квантования, что и для $NDVI_T$ (рис. 29).

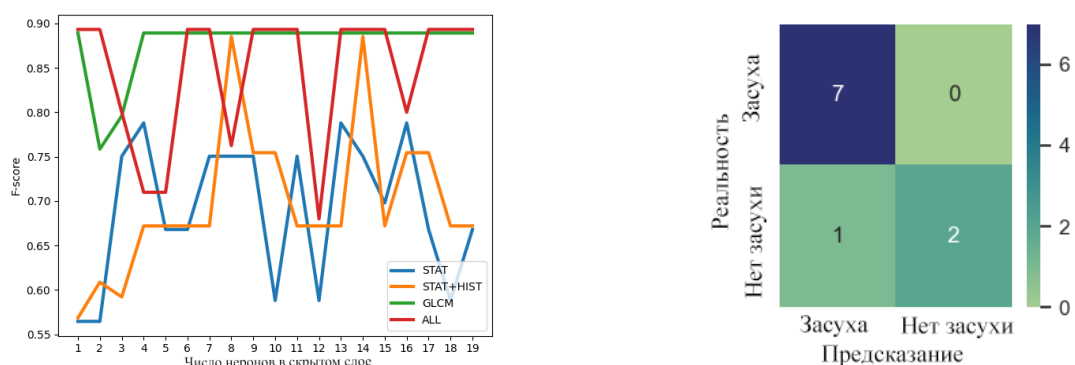


Рисунок 29. $NDVI_G$ графики. Зависимость эффективности $SLP_C(N)$ как от N , так и от использования разных групп признаков (слева); матрица ошибок для $SLP_C(14)$.

Достигнутый здесь F-score составляет 0.9. Это значение достигается для любого числа нейронов, кроме $N=3$ (рис. 29, слева). А признаки текстуры GLCM обеспечивают $F\text{-score}=0.9$ также для всех количеств нейронов, за исключением только 2 и 3. Таким образом, переход с $NDVI_T$ на $NDVI_G$ позволяет фермерам иметь хорошие результаты в детектировании засухи пшеницы, экономя свои ресурсы используя только RGB камеры.

На рис. 30 мы видим, что для 4 уровней квантования с 14 нейронами наиболее информативной является группа STAT, вторая - группа GLCM (рис. 15с). Число нейронов взято 14, поскольку при таком их числе группы признаков в совокупности дают наивысший F-score (рис 29, слева).

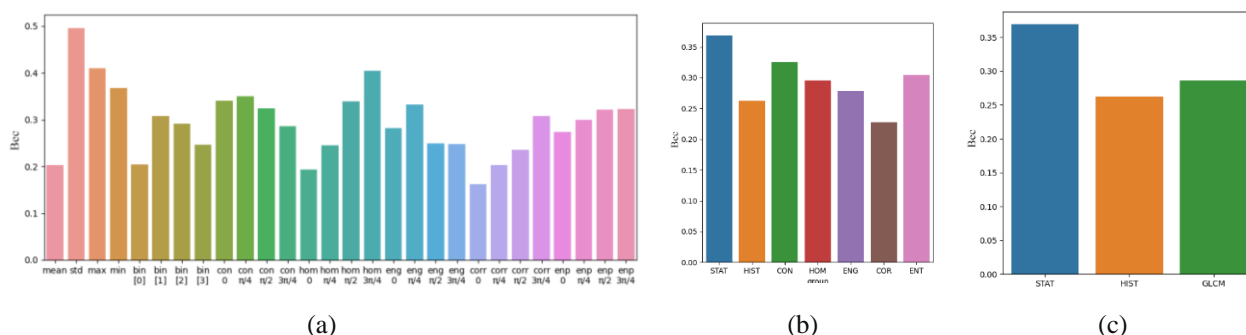


Рисунок 30. $NDVI_G$ графики. Веса, полученные после обучения детектора засухи $SLP_C(14)$. (a) Значения веса для каждого элемента вектора признаков. Вектор признаков включает 5 скаляров (1)-(5) на основе GLCM для каждого из 4 углов поворота. (b) Значения весов для групп признаков STAT, HIST и, отдельно, 5 скаляров на основе GLCM. (c) Значения весов для различных групп STAT, HIST и GLCM.

4.3.4 Модель регрессии SLP и ее анализ

Задача определения дня засухи имеет большее практическое значение. И наиболее сложной и актуальной задачей является определение 1-го дня стресса от засухи, который в нашем случае соответствует 5-му дню эксперимента без полива (см. рис. 20, слева). Для решения задачи с использованием признаков ALL (10), создан SLP-регрессор или $SLP_R(N)$ (рис. 31). Среднеквадратичная ошибка RMSE используется как индикатор качества.

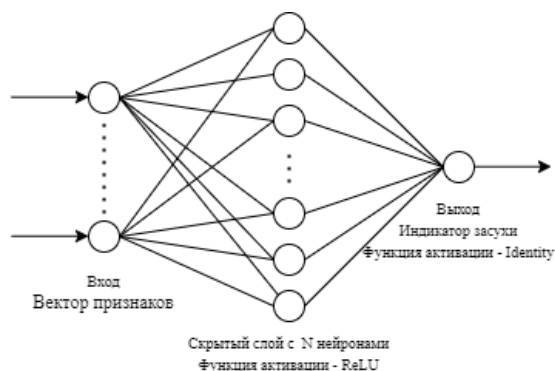


Рисунок 31. Архитектура $SLP_R(N)$ для прогнозирования дня засухи или стресса от засухи растений.

Случай с NDVI_T-изображениями. Исследование зависимости RMSE как от числа нейронов в скрытом слое, так и от выбора входного признака-вектора показало, что набор атрибутов STAT+HIST является лучшим решением в задаче регрессии, а SLP_R(27) оказался наиболее оптимальным (рис. 32a). Исследование было выполнено для 4-х однородных уровней квантования серого и бинов гистограммы ($L=4$) на промежутке $[-2\sigma, 2\sigma]$, как определено в (7). Лучшее RMSE составило 0,7 дня. Для атрибутов STAT + HIST также есть 5 значений RMSE менее 1 дня для $N=3,8,12,19,21$. В среднем, самой слабой группой функций оказалась GLCM.

Для случая NDVI_G-изображений было повторено то же исследование (рис. 32b). Здесь, в среднем, самой слабой группой функций была STAT. Наилучшую точность RMSE=3,1 дает SLP_R(2) для полного вектора признаков ALL. ALL остается лучшим для N от 1 до 3. Начиная с $N=4$ и далее, GLCM была заметно более сильной группой функций, но со значениями RMSE, колеблющимися около 4. Ближайшим конкурентом GLCM является группа STAT+HIST.

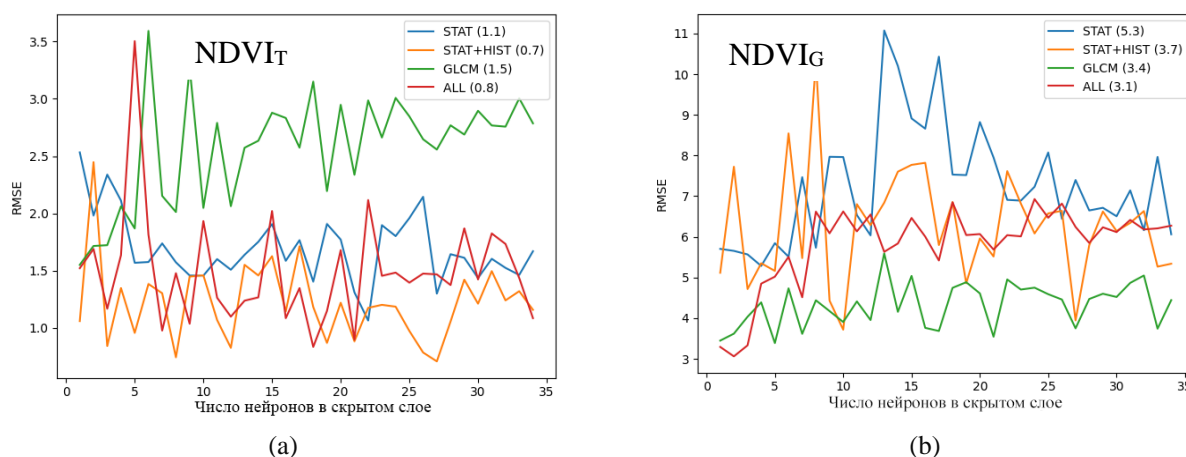


Рисунок 32. Индикатор RMSE [дни] для каждой группы функций: STAT, STAT+HIST, GLCM и ALL признаки. В легенде в скобках указано минимальное значение RMSE для данной группы признаков.
(a) NDVI_T график (b) NDVI_G график

Один из способов улучшить RMSE - изменить количество уровней квантования (L) как для изображений NDVI_T, так и для изображений NDVI_G. В работе исследуется использование двух правил квантования (таблица 11).

Таблица 11. Правила квантования

Число уровней	Пороги квантования (σ)	
	Правило I [-2,2]	Правило II [-1.5,1.5]
4	(-2, -1, 0, 1, 2)	(-1.5, -0.5, 0, 0.5, 1.5)
5	-	(-1.5, -0.75, -0.25, 0.25, 0.75, 1.5)
6	(-2, -1, -0.5, 0, 0.5, 1, 2)	(-1.5, -0.75, -0.25, 0, 0.25, 0.75, 1.5)
7	-	(-1.5, -1.0, -0.6, -0.2, 0.2, 0.6, 1.0, 1.5)
8	(-2, -1.5, -1, 0.5, 0, 0.5, 1, 1.5, 2)	(-1.5, -1.0, -0.6, -0.2, 0, 0.2, 0.6, 1.0, 1.5)
10	(-2, -1.5, -1, -0.5, -0.25, 0, 0.25, 0.5, 1, 1.5, 2)	-

Аргументы в пользу правил, предложенных в таблице 11, следующие:

Правило I. Квантование в пределах $[-2\sigma, 2\sigma]$. Для $NDVI_T$, в условиях смещения пика гистограммы вправо более чем на 1σ , уже была достигнута небольшая ошибка в 0.7 суток на текущих 4 бинах (-2,-1,0,1,2). Разделение интервалов сделает регрессор более чувствительным к первым дням засухи и уменьшит общую ошибку. Ожидается, что 10 уровней будут лучшими по точности.

Правило II. Квантование в пределах $[-1.5\sigma, 1.5\sigma]$. Более ориентирован на $NDVI_G$, распределение гистограмм у которого уже, чем у $NDVI_T$, а гистограммы во время засухи имеют смещение пика влево в пределах $0,5\sigma$. Наиболее заметные изменения происходят в промежутке $[-1\sigma, 0]$. При разделении этого интервала и его окрестности можно ожидать уменьшения ошибки регрессора.

При использовании правила I для $NDVI_T$ результат оптимизации $SLP_R(N)$ как по L, так и по комбинациям групп функций при каждом значении N показан на рис. 33а. Минимальное RMSE, равное 0.61, было достигнуто при $L=10$ с использованием группы функций STAT+HIST для двух значений $N=3, 8$ одновременно. Использование правила I для $NDVI_G$ (рис. 33b) дало снижение RMSE с 3.1 до 2.83, также с $L=10$, но уже с группой функций GLCM и только одним минимумом для $SLP_R(1)$. Это несколько менее важно, чем улучшения для $NDVI_T$.

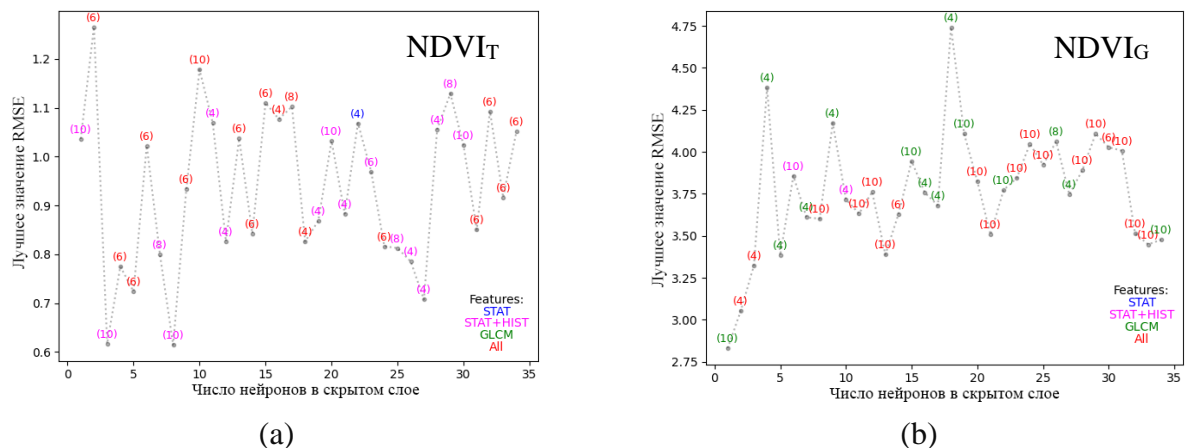
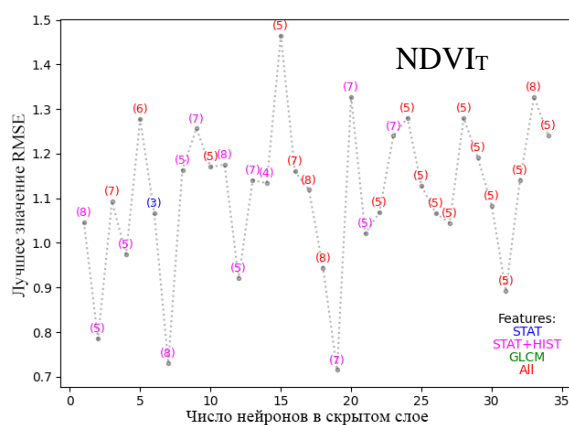


Рисунок 33. Зависимость лучших значений RMSE, полученных комбинируя различные группы признаков, от количества нейронов во внутреннем слое $SLP_R(N)$ и от способа квантования по правилу I (табл. 11).

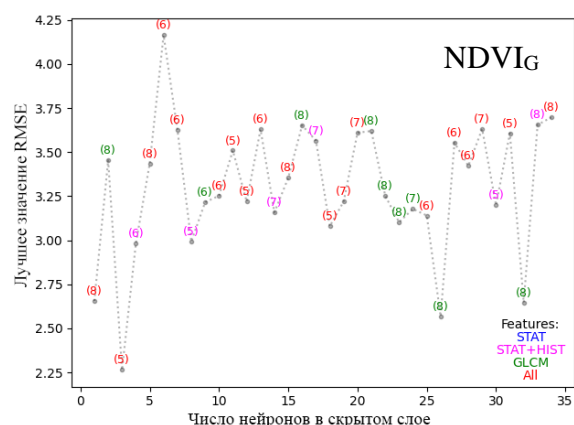
В скобках указано оптимальное количество уровней квантования.

$NDVI_T$ график слева, $NDVI_G$ график справа.

При использовании правила II для $NDVI_T$ (рис. 34а) лучший результат оптимизации $SLP_R(N)$ как по L, так и по комбинациям групп функций был: $RMSE = 0.71$ с $L=7$, группой функций STAT+HIST и $N=19$. То есть без улучшения относительно исходного результата без оптимизации по L: $RMSE=0,7$ при $L=4$, STAT+HIST для $SLP_R(27)$. Но использование правила II для $NDVI_G$ дало наилучшее $RMSE=2.26$ для $SLP_R(3)$ с группой функций ALL и три значения $RMSE$ меньше 2.75 с $L=8$ (рис. 34b).



(a)



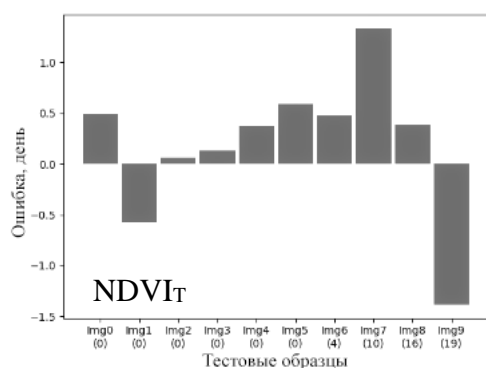
(b)

Рисунок 34. Зависимость лучших значений RMSE, полученных комбинируя различные группы признаков, от количества нейронов во внутреннем слое $SLP_R(N)$ и от способа квантования по правилу II (табл. 11).

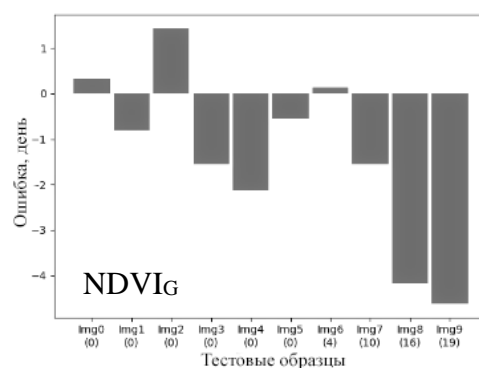
В скобках указано оптимальное количество уровней квантования.

NDVI_T график слева, NDVI_G график справа.

На рис. 35 анализируются ошибки для лучших моделей $SLP_R(N)$, для NDVI_T(N=3, L=10) и для NDVI_G(N=3, L=5).



(a)



(b)

Рисунок 35. Ошибки [в днях] для прогнозов лучших моделей $SLP_R(N)$ для NDVI_T (слева) и для NDVI_G (справа) на тестовой выборке. По оси абсцисс в скобках указано количество дней без полива.

Времена обучения лучших моделей представлены в таблице 12. Зеленым цветом выделены ячейки оптимальных групп признаков для советующей модели. Обучение на NDVI_G данных происходит медленнее, чем на NDVI_T. Время обучения SLP-классификатора не превышает 0.5 секунды, SLP-регрессора 2.5 секунд.

Таблица 12. Время обучения лучших моделей

Обучение на группе признаков	Классификация		Регрессия	
	Модель: SLP _C (14)	Модель: SLP _C (14)	Модель: SLP _R (N=3, L=10)	Модель: SLP _R (N=3, L=10)
	Данные: NDVI _T	Данные: NDVI _G	Данные: NDVI _T	Данные: NDVI _G
STAT	0.292	0.472	0.721	0.701
STAT + HIST	0.301	0.317	0.747	2.124
GLCM	0.141	0.215	1.422	2.339
ALL	0.111	0.179	0.421	1.08

4.3.5 Результаты исследования

Результаты использования построенных SLP(N)-классификатора и SLP(N)-регрессора на примере задачи ранней диагностики засухи растений пшеницы в целом подтверждают возможность их использования в качестве интерактивных ХАИ-блоков.

Замена обучения модели на наборе изображений обучением на векторе признаков дает блоку сразу несколько ХАИ-преимуществ: 1) простота; 2) уменьшение количества параметров модели; 3) объяснимые (связанные с методом предварительной обработки) и сгруппированные признаки на входе в блок; 4) возможность сравнительного исследования и оптимизации эффективности модели на уровне групп и индивидуальных особенностей; 5) возможность быстрого неитеративного обучения; 6) возможность использования блока в качестве таргет-корректора, который может эффективно исправлять ошибочные решения готовой ранее обученной сети.

Первым шагом, необходимым для замены набора изображений на набор векторов признаков, является преобразование данных в универсальную нормализованную форму. Для формирования признаков гистограммы, мы должны сделать следующее: 1) гистограммы здоровой и засушливой половин для всех ящиков и для всех дней эксперимента должны быть помещены на общую координатную ось; 2) необходимо вычислить общее среднее значение и стандартное значение; 3) гистограммы должны быть центрированы по среднему значению и нормализованы с использованием стандартного значения; 4) нормализованные гистограммы должны быть построены в виде дневных «здоровых» и «засушливых» графиков. Важно, чтобы статистика рассчитывалась для половинок засушливого ящика отдельно, но в координатах описанной нормализованной оси. В нашем исследовании диапазон $[-2\sigma, 2\sigma]$ был выбран в качестве характерной зоны.

Встроенные в блок инструменты анализа и визуализации позволяют принимать решения о характере взаимодействия различных групп признаков. Например, анализ зависимости эффективности групп признаков от количества нейронов в задаче регрессии для изображений NDVI_T (рис. 32a) показывает, что группа GLCM является не только наиболее неэффективной, но и антагонистом для STAT+HIST, поскольку ALL в среднем хуже, чем STAT+HIST. Для NDVI_G-изображений на (рис.32b) можно наблюдать противоположную картину: GLCM уже является наиболее эффективной группой, и STAT+HIST - его антагонист. В таких случаях может быть полезно не использовать антагониста. Уточнение характера взаимодействия позволяет объяснить результаты минимизации ошибки регрессора (рис. 33-34). Причина возросшей роли GLCM для изображений NDVI_G заключается в том, что

изображения RGB не имеют такого четкого дрейфа гистограммы, как в изображениях NDVI_T, а проявления засухи более заметны в текстуре листа растения.

4.4 Инструменты исследования

Разработка программного комплекса ведется на языке *Python*. Такой выбор языка программирования обусловлен наличием готовых библиотек для работы с тензорами. В частности, в данной работе используются библиотеки *numpy* и *pytorch*. Модель извлечения признаков реализована средствами библиотеки *pytorch*. Для вычисления GLCM и ее признаков используются функции библиотеки *scikit-image*. Обучение и тестирование классификаторов происходит с помощью инструментов библиотеки *scikit-learn*. Обучение классификаторов проводилось на процессоре *Intel Core i7-10700K @ 3,80 ГГц, 8 ядер, 16 ГБ*.

4.5 Описание структуры программы

Программа состоит из следующих модулей:

- *Serialize.py*. Содержит скрипт для сериализации ключевой информации об изображениях в json файл, содержащий: полный путь к изображениям и метку класса.
- *Dataset.py* содержит операции по доступу к изображениям базы данных.
- *Features.py* содержит скрипт для извлечения и сохранения признаков изображений.
- *Crossval.py* содержит скрипт для выполнения кросс-валидации и сохранения классификаторов с лучшим набором параметров.
- *Evaluate.py* содержит скрипт для тестирования классификаторов.
- *Utils.py* включает разные функции используемые в других скриптах
- *Inference.py* содержит скрипт для предсказания класса по изображению.
- Пакет *models* содержит модули, реализующие следующие классы:
 - *features.py* – содержит класс *Features* для извлечения признаков из изображений.
 - *slp.py* – содержит класс *SLP*, реализующий модель одноуровневого персептрона.
 - *healthyPlant.py* – содержит класс *HealthyPlant*, который реализует сквозной конвейер для классификации болезней растений, от извлечения признаков, до предсказания от классификатора.
 - *pdt.py* - содержит класс *PDT*, реализующий модель приоритетного дерева решений (таблица 1).

5. Заключение

В процессе выполнения работы был освоен теоретический материал, необходимый как база для дальнейших исследований. В частности, изучен статистический метод исследования текстуры с помощью матрицы GLCM. Получены базовые знания в области машинного обучения. Помимо теории, освоен с нуля язык программирования Python. Улучшены навыки работы с многомерными данными и их анализа.

Первая часть исследования содержит обзор методов извлечения признаков из изображений и моделей машинного обучения на примере задачи классификации болезней листьев растений.

Для обучения и тестирования классификаторов была найдена и подготовлена база данных, содержащая 6000 изображений листьев томатов, включающая в себя 6 классов больных и здоровых растений. В процессе исследования сформированы три группы признаков STAT (10), HIST (11) и GLCM (12), которые все вместе образовали группу ALL (9). Из полного набора статистических и текстурных признаков long выделены вектора меньшей длины middle и short (табл. 4), максимально описывающие полезную информацию. Также рассматривалось два варианта извлечения признаков: локальный и глобальный, и два «источника» признаков: изображения в красном канале и NDVI_G (13) образы.

Как показали результаты экспериментов (прил. D-E), глобальные признаки проигрывают в качестве в среднем на 2–3% при классификации на полном наборе данных. Это происходит в основном за счёт потерь на признаках STAT и HIST. Однако нет разницы в использовании глобальных или усредненных локальных GLCM признаков. Этот факт привел к созданию вектора комбинированных признаков, в котором использовались локальные статистические и глобальные текстурные признаки.

В ходе выполнения работы было предположено, что для диагностирования болезни достаточного использовать только красный канал изображений. Высокие результаты в Приложениях D-E, доказывают справедливость этого предположения. Также было предложено использовать NDVI аналог на базе красного и зеленого каналов (13), который также доказал свою работоспособность результатами в Приложении F-G.

В итоге лучшими классификаторами оказались одноуровневый персептрон SLP и метод опорных векторов SVM, показатели F-score которых для признаков из красного канала составили 83%, для NDVI_G признаков 87%. При этом для достижения такой точности им понадобился весь вектор признаков ALL (9).

Во второй части исследования предлагается конструкция простого интерактивного модуля объяснимого искусственного интеллекта (ХАИ) на основе однослойного персептрона SLP с N нейронами на скрытом слое. Этот блок был реализован в двух вариантах: как классификатор $SLP_C(N)$ и как регрессор $SLP_R(N)$, оснащенный средствами анализа и визуализации. Эти блоки ориентированы на обработку изображений, но не использование изображений напрямую. Каждое изображение должно быть представлено вектором признаков, состоящим из результатов обработки изображения с помощью ряда общих и/или проблемно-ориентированных методов. В данной работе общие методы представлены параметрами групп STAT и HIST, проблемно-ориентированные представлены группой GLCM.

Использование вектора признаков вместо самого изображения дает сразу 6 ХАИ-преимуществ: 1) простота; 2) резкое уменьшение количества параметров модели; 3) объяснимые (связанные с методом предварительной обработки) и сгруппированные признаки на входе в блок; 4) возможность сравнительного исследования и оптимизации эффективности модели на уровне групп и индивидуальных особенностей; 5) возможность быстрого неитеративного обучения; 6) возможность использования блока в качестве таргет-корректора.

Таким образом, все поставленные задачи выполнены. Предложенная конструкция классификатора и регрессора на основе SLP (SLP_C и SLP_R) доказала свою эффективность и надежность в раннем обнаружении засухи пшеницы и может быть рекомендована в качестве универсальных ХАИ-блоков.

6. Список литературы

1. S. Chakrabortya, A.C. Newton, “Climate change, plant diseases and food security: an overview”, *Plant Pathology* (2011), Vol. 60, Issue 1, pp 2–14, doi: 10.1111/j.1365-3059.2010.02411.x.
2. United Nations, Department of Economic and Social Affairs, Population Division (2019). *World Population Prospects 2019: Highlights* (ST/ESA/SER.A/423).
3. A.-K. Mahlein et al. “Development of spectral indices for detecting and identifying plant diseases” / *Remote Sensing of Environment* 128, 2013, pp. 21–30
4. K.P. Ferentinos, “Deep learning models for plant disease detection and diagnosis”. *Computers and Electronics in Agriculture* 145, 2018, pp. 311– 318
5. K. Jha, A. Doshi, P. Patel, M. Shah. “A comprehensive review on automation in agriculture using artificial intelligence” / *Artificial Intelligence in Agriculture*, vol.2, June 2019, pp. 1-12. DOI 10.1016/j.aiia.2019.05.004
6. Tanha Talaviya et al. “Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides” / *Artificial Intelligence in Agriculture*, vol.4, 2020, pp. 58-73. DOI 10.1016/j.aiia.2020.04.002
7. Misbah Pathan et al. “Artificial cognition for applications in smart agriculture: A comprehensive review” / *Artificial Intelligence in Agriculture*, vol. 4, 2020, pp.81-95. DOI 10.1016/j.aiia.2020.06.001
8. R. M. Haralick, K. Shanmugam, and I. Dinstein, “Textural features for image classification,” *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.
9. R. W. Connors and C. A. Harlow, “A theoretical comparison of texture algorithms” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-2, no. 3, pp. 204–222, May 1980.
10. J. Weier, D. Herring, “Measuring Vegetation”, *NASA Earth Observatory*, 2000-08-30, pp. 4.
11. Dataset of images. Available: <https://arxiv.org/abs/1511.08060>
12. Дубинин М. NDVI – теория и практика // *Географические информационные системы и дистанционное зондирование GIS-LAB – Электрон. журн.* – Режим доступа: <https://gis-lab.info/qa/ndvi.html>

7. Приложение

Приложение А. Визуализация локальных признаков.

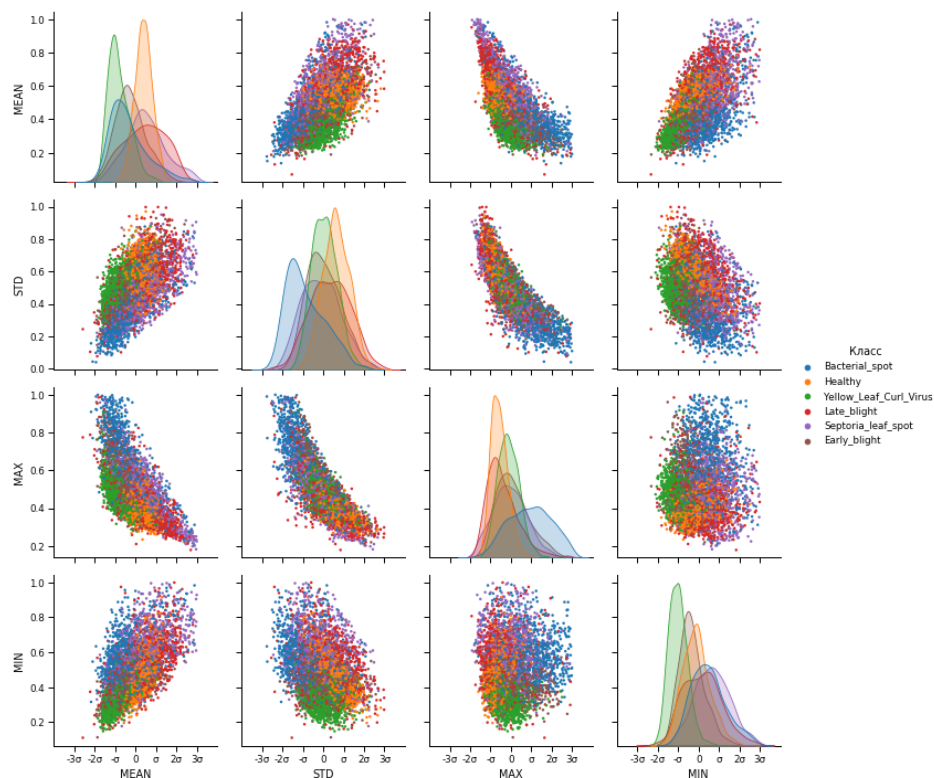


Рисунок А1. Вид распределения статистических признаков, извлеченных глобально

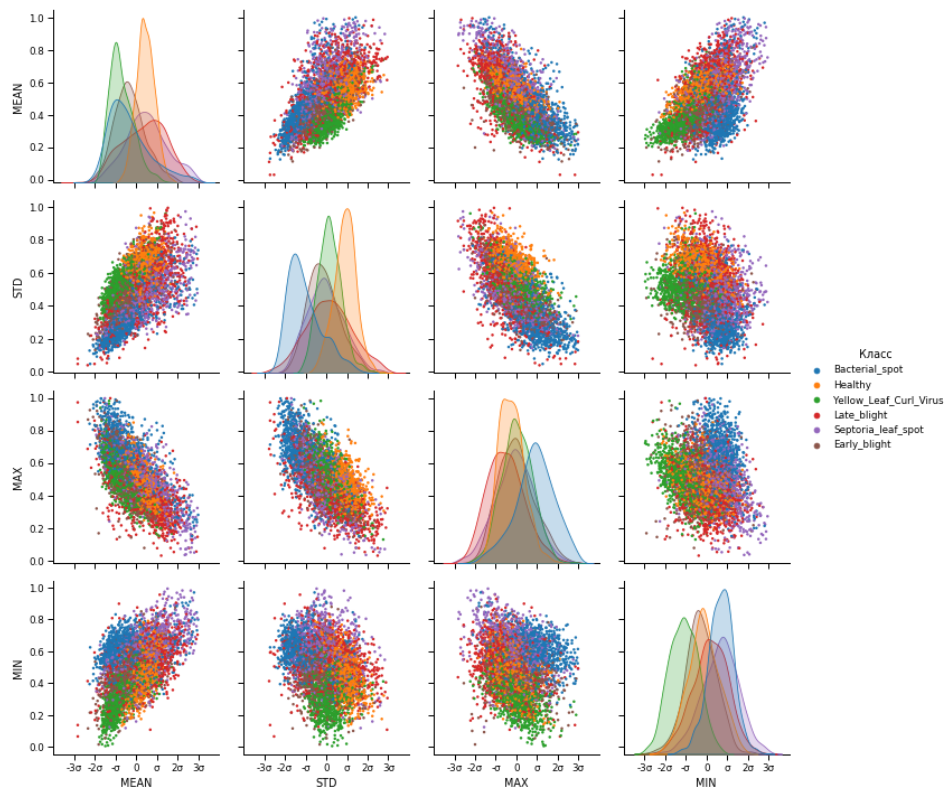


Рисунок А2. Вид распределения статистических признаков, извлеченных локально

Приложение В. Визуализация локальных признаков.

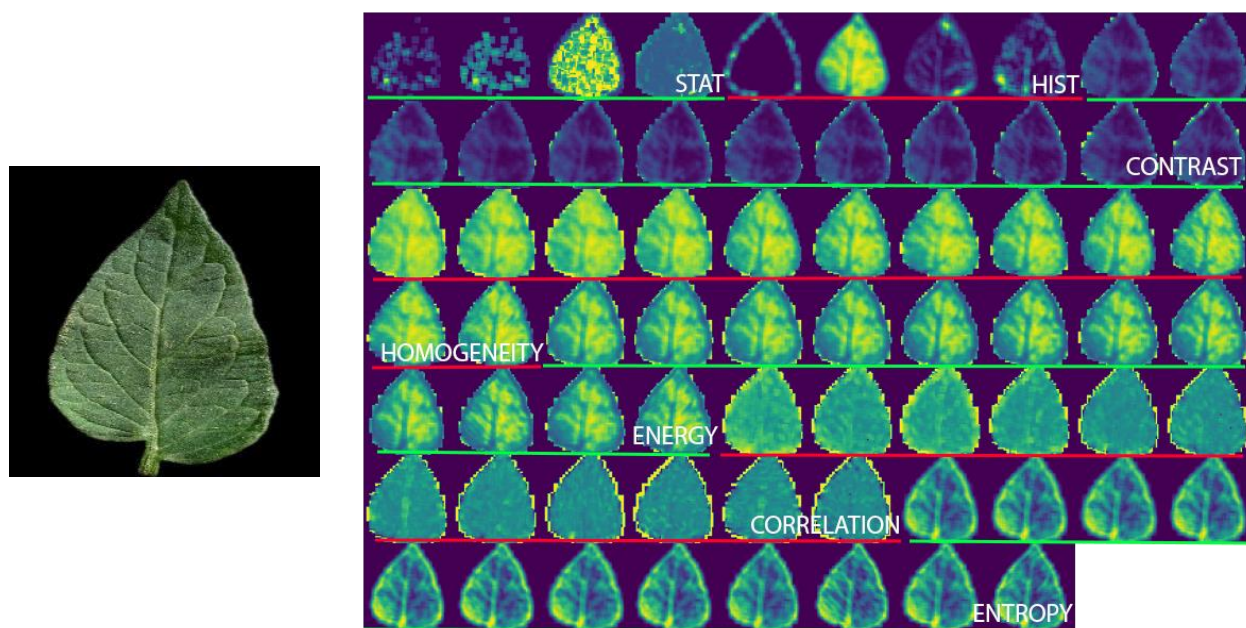


Рисунок В1. Исходное изображение здорового листа (слева) и вектор его локальных признаков ALL, описанный в пункте 3.3, табл. 6 (справа)

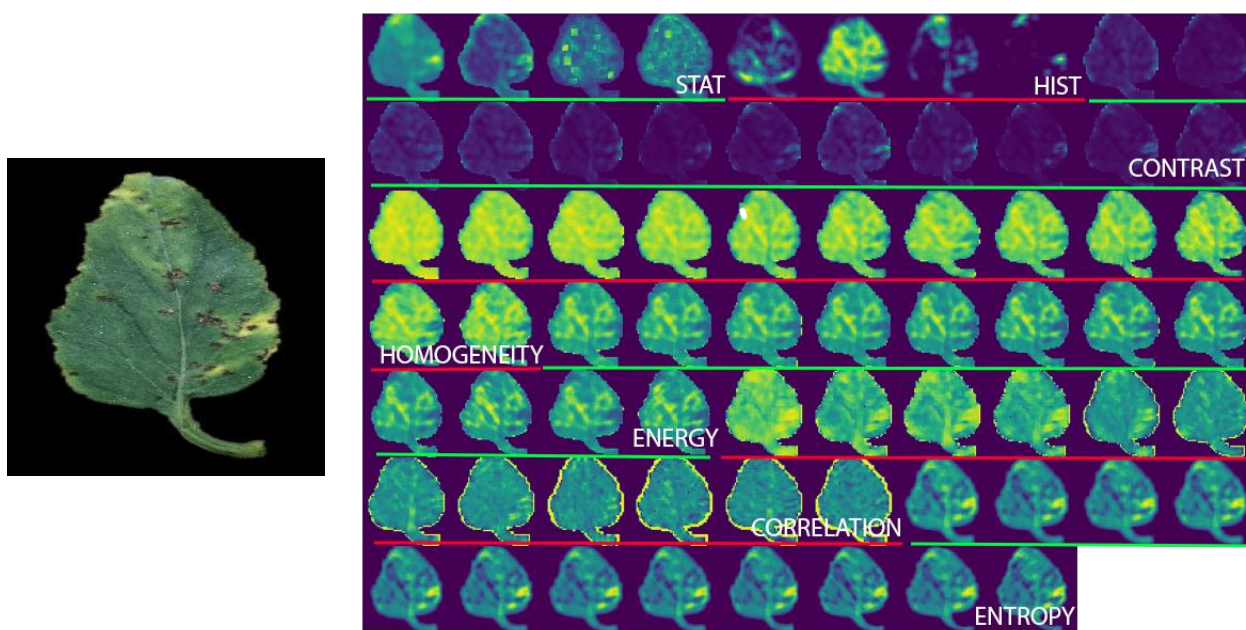


Рисунок В2. Исходное изображение больного Bacterial spot листа (слева) и вектор его локальных признаков ALL, описанный в пункте 3.3, табл. 6 (справа)

Приложение С. Подбор параметров классификаторов.

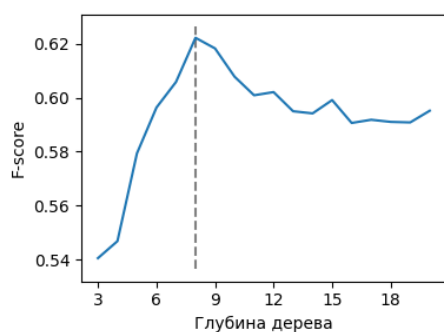


Рисунок С1. График F-score в зависимости от глубины (max_depth) дерева решений DT.

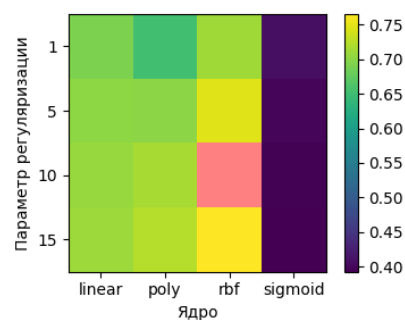


Рисунок С2. Значения F-score, в зависимости от выбора ядра (kernel), и значения параметра регуляризации(C) для метода опорных векторов SVM. Красным выделен набор параметров, показывающий наивысший F-score.

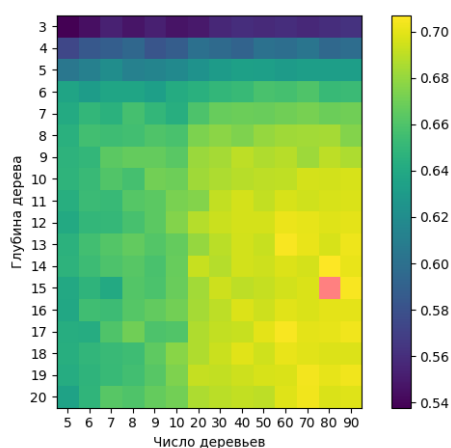


Рисунок С3. Значения F-score, в зависимости от выбора глубины деревьев (max_depth) и числа деревьев (n_estimators) для случайного леса RF. Красным выделен набор параметров, показывающий наивысший F-score

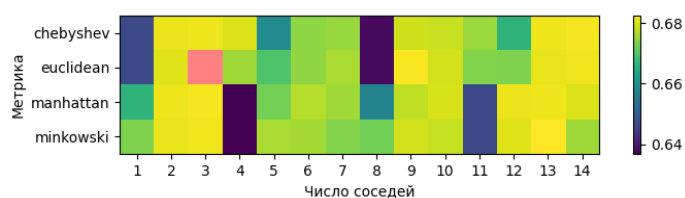


Рисунок С4. Значения F-score, в зависимости от выбора метрики (metric) и числа соседей (n_neighbors) для метода ближайших соседей KNN. Красным выделен набор параметров, показывающий наивысший F-score

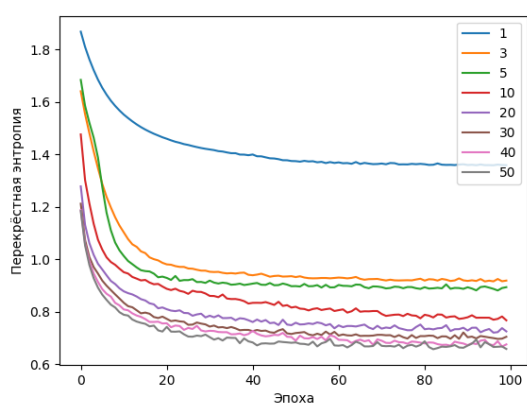


Рисунок С5. Выбор оптимального числа нейронов в скрытом слое для одноуровневого персептрона SLP на примере падения ошибки loss функции (CrossEntropyLoss).

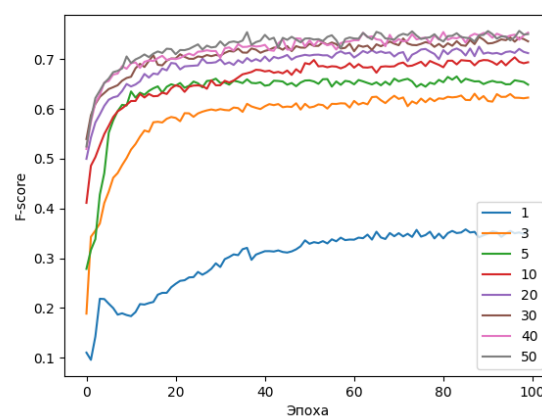
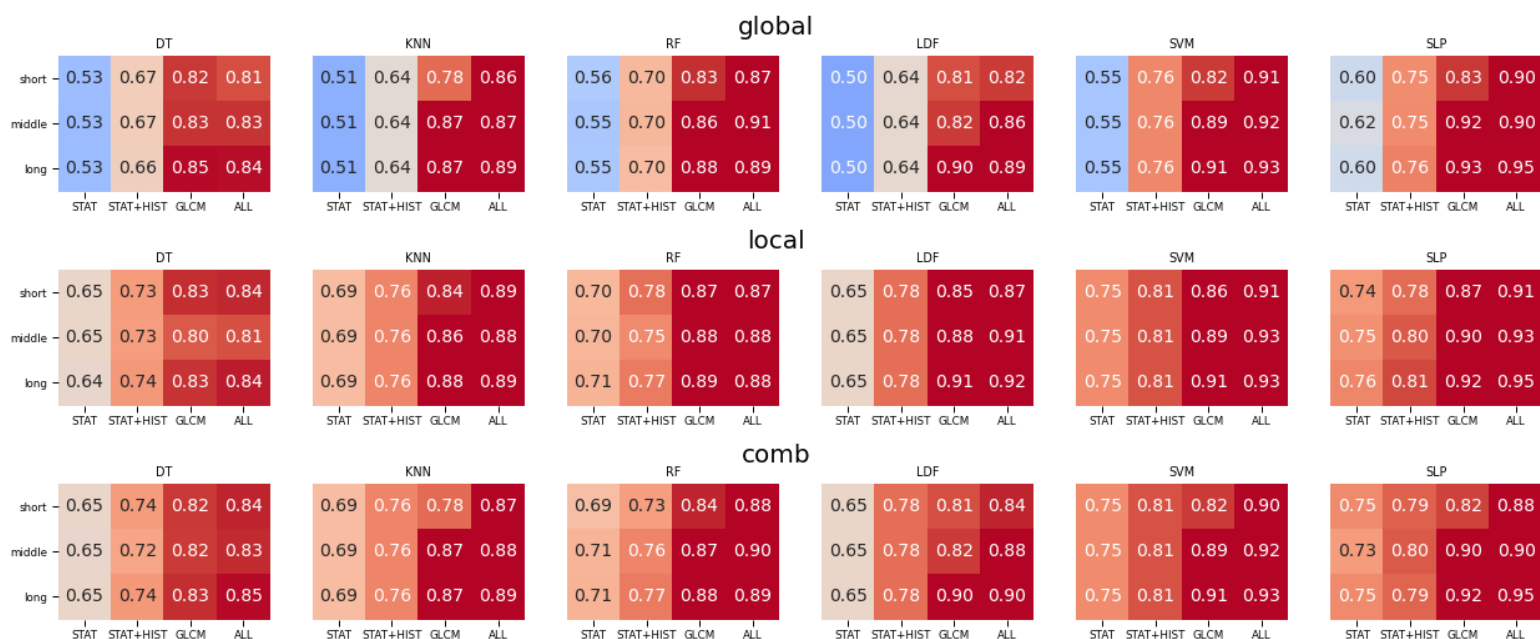
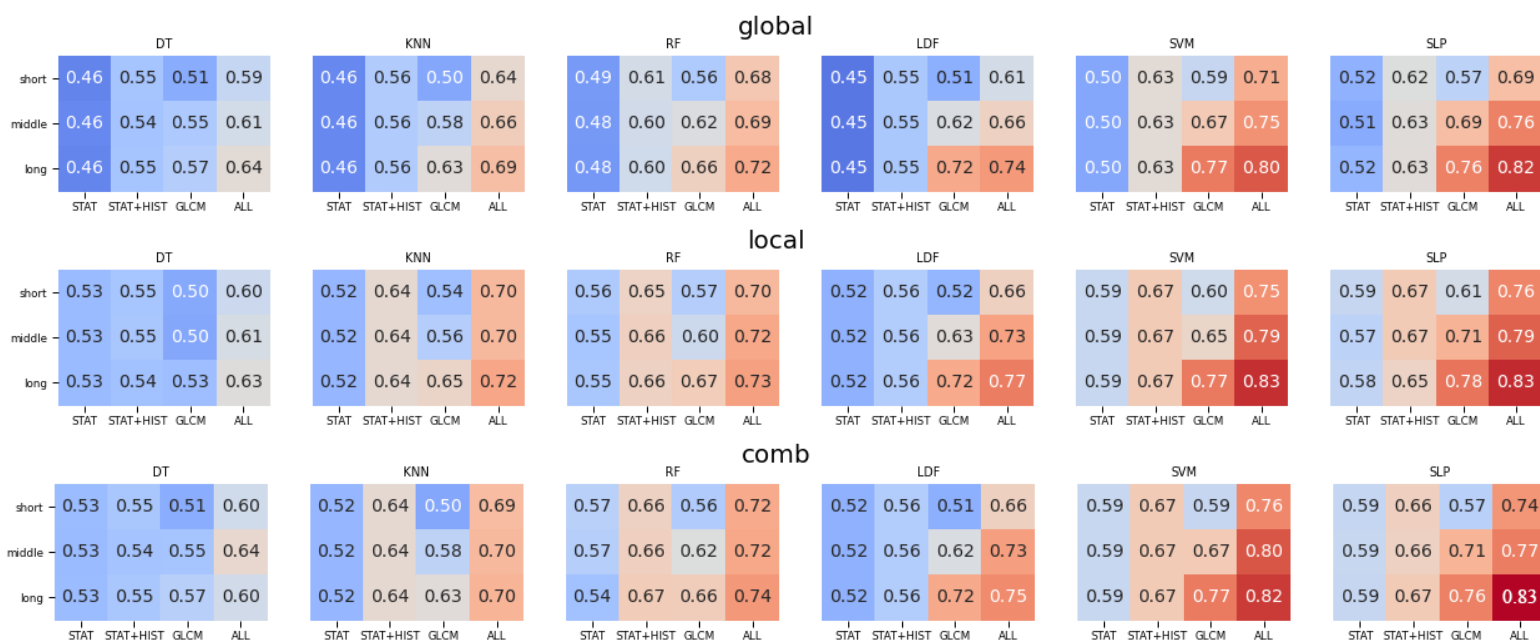


Рисунок С6. Выбор оптимального числа нейронов в скрытом слое для одноуровневого персептрона SLP на примере роста графика F-score.

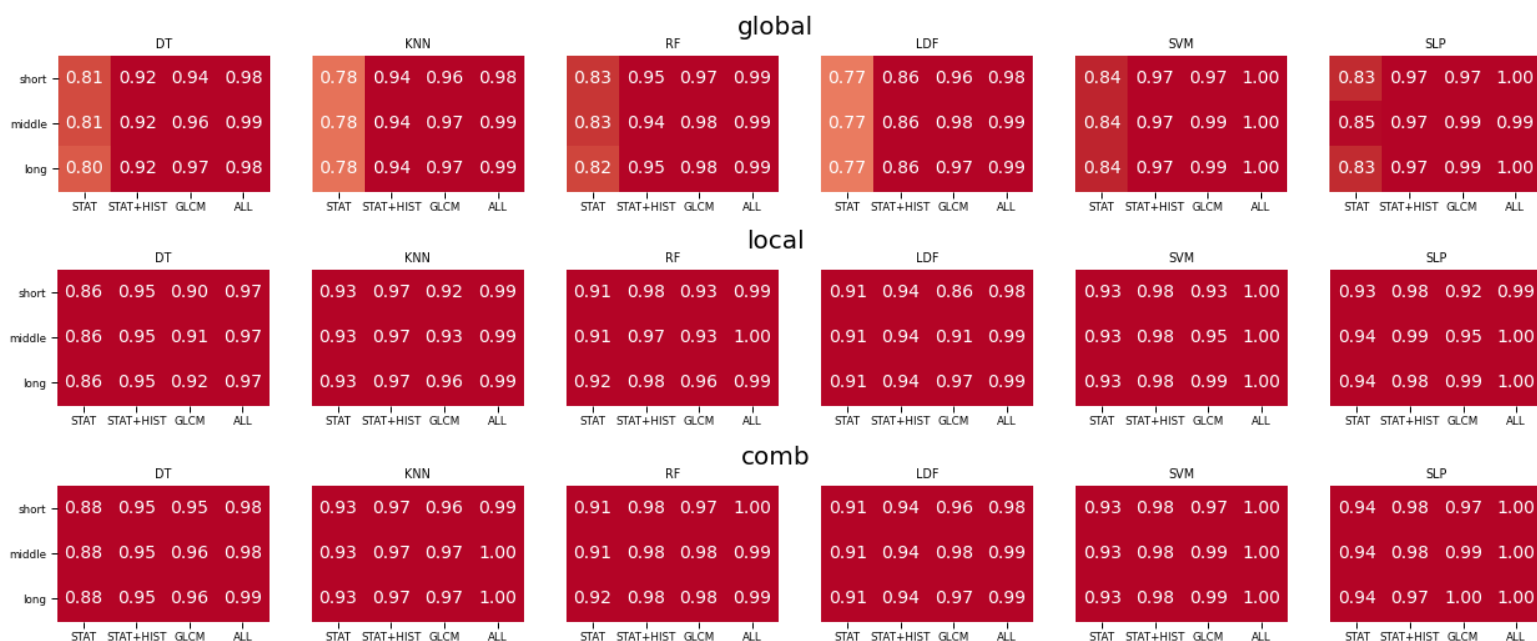
Приложение D. Показатели F-Score в задаче обнаружения болезни на основе локальных, глобальных и комбинированных признаков в красном канале



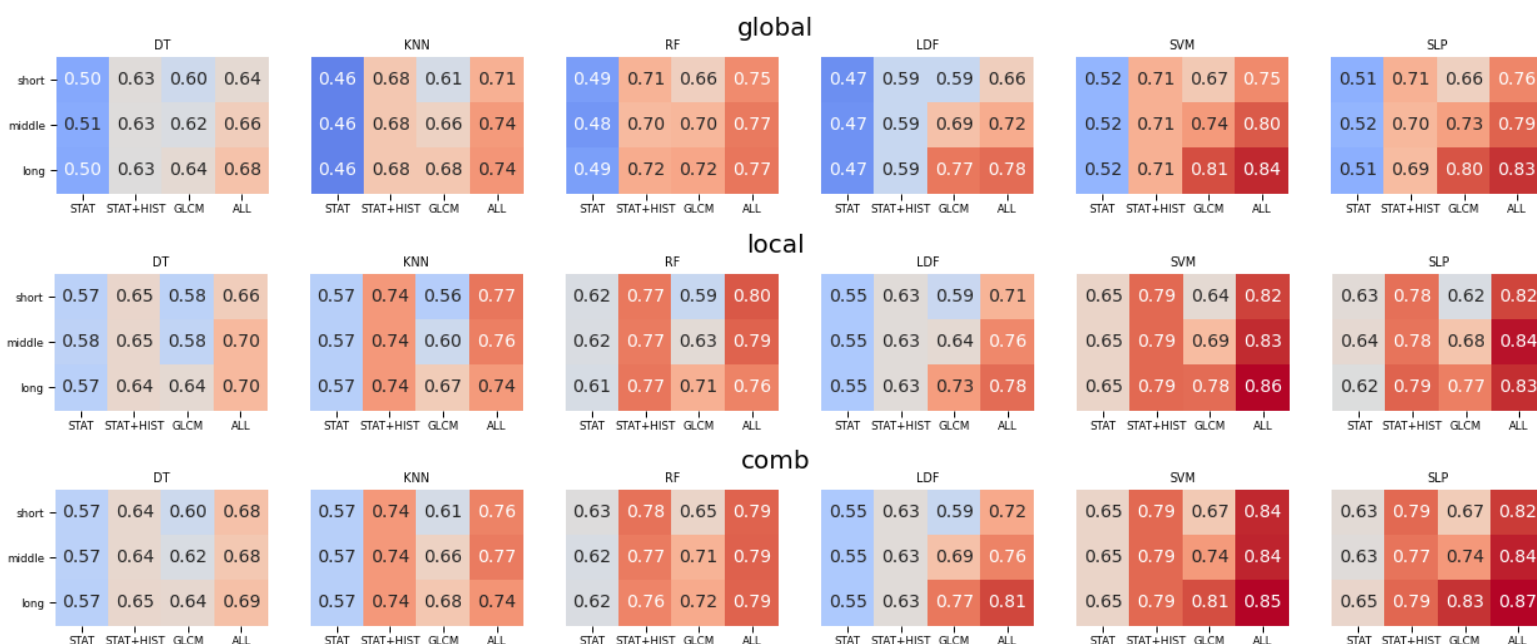
Приложение E. Показатели F-Score в задаче классификации болезни на основе локальных, глобальных и комбинированных признаков в красном канале



Приложение F. Показатели в задаче обнаружения болезни на основе локальных, глобальных и комбинированных NDVI признаков



Приложение G. Показатели F-Score в задаче классификации болезни на основе локальных, глобальных и комбинированных NDVI признаков



Приложение Н. Листинг кода извлечения полного вектора признаков

```
1. class Features(nn.Module):
2.     GLCM_PROPERTIES = ('contrast', 'homogeneity', 'energy', 'correlation')
3.
4.     def __init__(self, mode, channel, ds_stat, bins, dist, theta, n_jobs=8):
5.         super().__init__()
6.         if channel not in ['B', 'G', 'R', 'ndvi']:
7.             raise ValueError(f'Arg \'channel\'={channel} is not valid')
8.         if len(dist) is None or len(theta) is None:
9.             raise ValueError(f'Arg \'dist\' or \'theta\' is empty')
10.        if ds_stat is None:
11.            raise ValueError(f'Arg \'stats\' is empty')
12.
13.        self.stride = [4, 4] if mode == 'local' else [1, 1]
14.        self.mask_size = [17, 17] if mode == 'local' else [257, 257]
15.        self.bins = bins
16.        self.n_bins = len(self.bins) + 1
17.        self.channel = channel
18.        self.mean = ds_stat[f'{channel}_mean']
19.        self.std = ds_stat[f'{channel}_std']
20.        self.dist = dist
21.        self.theta = theta
22.        self.n_jobs = n_jobs
23.
24.
25.        def all_sliding_windows(self, imgs):
26.            """RETURNS ARRAY OF SHAPE (B, C, Nx, Ny, MaskX, MaskY)"""
27.            shape = imgs.shape[:-2] + \
28.                ((imgs.shape[-2] - self.mask_size[0]) // self.stride[0] + 1,) + \
29.                ((imgs.shape[-1] - self.mask_size[1]) // self.stride[1] + 1,) + \
30.                tuple(self.mask_size)
31.
32.            strides = imgs.strides[:-2] + \
33.                (imgs.strides[-2] * self.stride[-2],) + \
34.                (imgs.strides[-1] * self.stride[-1],) + \
35.                imgs.strides[-2:]
36.            sliding_windows = np.lib.stride_tricks.as_strided(imgs,
37.                shape=shape, strides=strides)
38.            return sliding_windows
39.
40.
41.        @staticmethod
42.        def stat(img):
43.            img = img[img != 0]
44.            mean = img.mean()
45.            std = img.std()
46.            maximum = (img.max() - mean) / (std + 1e-9)
47.            minimum = (mean - img.min()) / (std + 1e-9)
48.            return [mean, std, maximum, minimum]
49.
```

```

50.     def quantize(self, img):
51.         if self.channel != 'ndvi':
52.             img = (img - self.mean) / self.std
53.             img = (img + 3) / 6
54.             bins = np.full((len(self.bins),), 0.5) + np.asarray(self.bins)*1/6
55.             q_img = np.digitize(img, bins)
56.             return q_img
57.
58.
59.     def hist(self, q_img):
60.         hist, _ = np.histogram(q_img.flatten(), density=True
61.                                bins=np.arange(self.n_bins + 1))
62.         return hist[1: -1]
63.
64.
65.     def glcm(self, q_img):
66.         g = greycomatrix(q_img, self.dist, self.theta, self.n_bins,
67.                           normed=True, symmetric=True)
68.         g = g[1:1, 1:1, :, :]
69.         props = np.array([greycoprops(g, p) for p in
70.                           self.GLCM_PROPERTIES]).reshape(-1)
71.         entropy = -np.sum(np.multiply(g, np.log2(g + 1e-8)),
72.                            axis=(0, 1)).reshape(-1)
73.         props = np.concatenate([props, entropy])
74.         return props
75.
76.
77.     def stat_hist_glcm(self, img):
78.         if np.max(img) == 0:
79.             return [0] * (4 + len(self.bins) - 1 + 3 * 4 * 5)
80.         stat = self.stat(img)
81.         q_img = self.quantize(img)
82.         hist = self.hist(q_img)
83.         glcm = self.glcm(q_img)
84.         return np.concatenate([stat, hist, glcm])
85.
86.
87.     def parallel_extractor(self, imgs):
88.         if self.channel != 'ndvi':
89.             imgs[imgs < 10] = 0
90.             shp = np.shape(imgs)
91.             loc_imgs = imgs.reshape((-1,) + shp[-2:])
92.             gcs = Parallel(n_jobs=self.n_jobs)
93.                 (delayed(self.stat_hist_glcm)(img) for img in loc_imgs)
94.             gcs = np.stack(gcs, axis=0)
95.             gcs = gcs.reshape(shp[:-2] + (-1,))
96.             gcs = gcs.transpose(0, -1, 2, 3, 1)
97.             return gcs.squeeze(-1)

```

```

98.     def forward(self, x):
99.         if self.channel == 'ndvi':
100.             red = x[:, 2, :, :]
101.             green = x[:, 1, :, :]
102.             ndvi = (green - red) / (green + red + 1e-9)
103.             ndvi = (ndvi - self.mean) / self.std
104.             ndvi = (ndvi + 3) / 6
105.             ndvi[(red == 0) & (green == 0)] = 0
106.             x = ndvi
107.         else:
108.             ch = ['B', 'G', 'R'].index(self.channel)
109.             x = x[:, ch, :, :]
110.
111.             x = x.numpy()
112.             x = np.expand_dims(x, 1)
113.
114.             x = self.all_sliding_windows(x)
115.             features = self.parallel_extractor(x)
116.             features = np.sum(features, axis=(-2, -1)) /
117.                 (np.count_nonzero(features, axis=(-2, -1)) + 1e-9)
118.             return features

```