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:
«Методы машинного обучения в задаче классификации болезней листьев томатов»

Выполнила:
студентка группы 381706-1
Максимова Ирина Игоревна

_____
(подпись)

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

_____
(подпись)

Нижний Новгород
2020

Оглавление

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 [3]. 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 [4]. Поэтому, 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 классификации заболеваний растений.

Актуальность данной работы заключается в автоматизации процесса мониторинга сельскохозяйственных угодий. **Проблематика** данной работы заключается в возможности успешно классифицировать болезни листьев томатов.

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, мумификация и деформация органов растения, отмирание, изменение цвета тканей, появление гнили, пятен, налета, пустулы, нароста или опухоли, выделение камеди.

Поскольку появление побочных внешних эффектов сигнализирует о реакции растения на заболевание, то классификация больных 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, поскольку оно обладает высоким показателем точности и большим потенциалом.

В зависимости от задачи, используются разные методы машинного обучения. Касательно рассматриваемой темы в работе будут применяться алгоритмы классификации. Каждый из них имеет свои преимущества и недостатки в задаче классификации. Поэтому необходимо провести обзор классификаторов и определить наиболее эффективный метод классификации, применяемый в исследуемой области.

Целью данной работы является поиск наилучшего алгоритма классификации болезней листьев растений. **Объектом** исследования являются методы извлечения признаков изображения и базовые алгоритмы машинного обучения, а **предметом** исследования – классификация болезней листьев томатов по внешним симптомам.

В рамках преследуемой цели сформулирован общий подход по созданию автоматизированной системы классификации болезней листьев растений. Предлагаемый подход состоит из следующих этапов:

Подготовка данных. Для начала необходимо найти базу данных, на которой будет выполняться тренировка и тестирование алгоритмов машинного обучения. Данные должны содержать размеченные изображения как больных, так и здоровых листьев исследуемого сорта растения. Качество листовых изображений играет важную роль, и поэтому необходимо использовать надежный источник. Выборка должна быть достаточно большой для обеспечения высокой обобщающей способность будущей модели.

Извлечение и анализ статистических и текстурных признаков. На данном шаге мы уменьшаем размер данных путем получения необходимой информации из изображения. Предполагается, что детектировать заболевание, то есть обнаружить пораженную область, 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. Анализ текстур направлен на поиск уникального способа представления основных характеристик текстур и представления их в некоторой более простой, но уникальной форме. Поэтому, в рамках задачи классификации, будут использоваться текстурные характеристики листьев. Наиболее распространенным вариантом получения статистических характеристик текстуры является матрица совместной встречаемости уровней серого тона (GLCM). Для ее вычисления задаются 2 параметра: расстояние d и угол φ. Определение лучших значений параметров d, φ является отдельной подзадачей в этом исследовании.

Тестирование алгоритмов машинного обучения. На основе извлеченных признаков осуществляется классификация болезней листьев растений. В данной работе будут рассмотрены следующие классификаторы: деревья решений (DTC), случайный лес (RF),

мультиклассовый метод опорных векторов (MSVM), k-ближайших соседей (KNN), одноуровневый персептрон (1 Layer MLP).

Определение информативного набора признаков. С точки зрения анализа данных, после извлечения полного набора признаков, будет полученная модель, которую сложно интерпретировать и сложно вычислять. Как правило, признаки довольно сильно зависят друг от друга и их одновременное наличие – избыточно. Поэтому нужно создать упрощенную модель, максимально хорошо решающую поставленную задачу. Снижение размерности пространства признаков является важной подзадачей в машинном обучении. В данной работе для анализа признаков и извлечения наиболее информативных будет использоваться простой и популярный метод главных компонент (МГК).

Вышеизложенные этапы являются базовыми и обязательными в задаче классификации болезней листьев растений. Качественная реализация каждого этапа позволит с высокой точностью диагностировать заболевание растения.

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 методами машинного обучения, используемыми в задачах классификации – DTC, RF, MSVM, KNN и 1 Layer MLP.
   1.3 Рассмотреть способы оценки качества работы классификатора.
   1.4 Ознакомиться с методом главных компонент.
2. Подготовить базу данных для обучения и тестирования классификаторов.
3. Извлечь статистические и текстурные GLCM признаки для всех изображений из базы данных, на основе которых будет выполняться классификация.
4. Исследовать качество работы различных классификаторов на извлеченных данных.
5. Выявить наиболее показательные признаки с минимальной потерей качества классификации. Определить лучшие параметры GLCM: расстояние d, угол .
6. Сделать выводы по возможности применения каждого из рассматриваемых классификаторов и признаков для и классификации болезни

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-occurence 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, φ), 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:

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

- 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 данного класса текстуры. Для этой цели Haralick, Shanmugam и Dinstein [8] предложили 14 мер. Позже Conners 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.

- *Корреляция*. Измеряет линейную зависимость интенсивностей пикселей относительно друг друга.

$$\text{корреляция} = \sum_{i,j} P_{ij} * \frac{(i - \mu_i) * (j - \mu_j)}{\sigma_i * \sigma_j} \quad (1)$$

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

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

- *Энергия*. Измеряет текстурную однородность.

$$\text{энергия} = \sqrt{\sum_{i,j} P_{ij}^2} \tag{2}$$

- *Энтропия*. Измеряет беспорядок или сложность изображения.

$$\text{энтропия} = \sum_{i,j} \mathrm{P_{ij}} * (-\ln \mathrm{P_{ij}}) \tag{3}$$

- *Контраст*. Определяет локальные изменения интенсивности.

$$\text{контраст} = \sum_{i,j} \mathrm{P_{ij}} * (\mathrm{i} - \mathrm{j})^2 \tag{4}$$

- *Однородность*. Измеряет близость распределения элементов GLCM к диагонали.

$$\text{однородность} = \sum_{i,j} \frac{\mathrm{P_{ij}}}{1 + (\mathrm{i} - \mathrm{j})^2} \tag{5}$$

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 Дерево решений

Дерево решений (DTC) – 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.

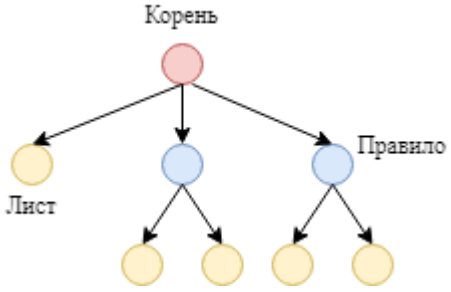

Рисунок 1. Структура дерева решений

В корневом узле располагается множество входных данных, в листовых узлах – значение целевой функции, в остальных узлах – правило перехода, определяющее по какому из ребер идти. Правило перехода, или правило разбиения данных на подмножества на

текущем узле, определятся из условия минимизации среднего значения энтропии внутри этих подмножеств.

3.2.2 Случайный лес

В основе метода *случайного леса* (RF) 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».

3.2.3 Мультиклассовый метод опорных векторов

Основная цель *метода опорных векторов* (SVM) 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* (MSVM).

3.2.4 Метод k-ближайших соседей

Метод K-ближайших соседей (KNN) — 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.

3.2.5 Одноуровневый персептрон

Одноуровневый персептрон (1 Layer MLP) представляет собой нейронную сеть прямого распространения. Структура 1 Layer MLP:

1. Входной слой. Во входной слой поступают исходные данные, и передаются дальнейшим слоям.
2. Один скрытый слой. В нем происходит взвешенное суммирование выходных сигналов предыдущего слоя и формирование выхода посредством нелинейной функции активации.
3. Выходной слой. Вывод результата. В данной работе выходом будет распределения вероятностей принадлежности к определённому классу.

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

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*):

$$\text{Accuracy} = \frac{\text{N}_{\text{correct}}}{\text{N}_{\text{total}}} \tag{6}$$

где $\text{N}_{\text{correct}}$ - число верных предсказаний алгоритма, N_{total} - число объектов в тестовой выборке. Однако стоит отметить, что 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:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{7}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{8}$$

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:

$$\text{F-score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

3.4 Метод главных компонент

Пусть заданы признаки $X^{(j)}, j = \overline{1,k}$ и матрица данных X:

$$\mathrm{X} = \begin{bmatrix} \mathrm{X}_1^{(1)} & \cdots & \mathrm{X}_1^{(\mathrm{k})} \\ \vdots & \ddots & \vdots \\ \mathrm{X}_\mathrm{n}^{(1)} & \cdots & \mathrm{X}_\mathrm{n}^{(\mathrm{k})} \end{bmatrix}$$

В связи с тем, что анализ набора признаков часто проводится в стандартизованной форме, ниже рассматриваются определение и свойства главных компонент, извлекаемых на основе матрицы корреляций признаков R.

Собственные числа матрицы R упорядочим по убыванию и обозначим через $\lambda_j, j = \overline{1,k}$:

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_\mathrm{k} > 0 \qquad (10)$$

Собственный вектор, соответствующий числу λ_j, обозначим через $W^{(j)}$ и запишем как вектор-столбец в матрицу собственных векторов W:

$$\mathrm{W} = \left[\mathrm{W}^{(1)}, \mathrm{W}^{(2)}, \ldots, \mathrm{W}^{(\mathrm{k})}\right] = \begin{bmatrix} \mathrm{W}_1^{(1)} & \cdots & \mathrm{W}_1^{(\mathrm{k})} \\ \vdots & \ddots & \vdots \\ \mathrm{W}_\mathrm{k}^{(1)} & \cdots & \mathrm{W}_\mathrm{k}^{(\mathrm{k})} \end{bmatrix}$$

где через $W_s^{(j)}$ обозначена s-я координата собственного вектора с номером j (номер вектора указан верхним индексом, а номер его координаты – нижним). Без ограничения общности считаем, что векторы $W^{(j)},\ j = \overline{1,k}$, ортонормированы.

Главными компонентами набора признаков $X^{(j)}, j = \overline{1,k}$, выделяемыми на основе корреляционной матрицы R, называют новые признаки $C^{(s)}, s = \overline{1,k}$, определяемые на основе ортонормированных собственных векторов матрицы R и стандартизованных признаков $Z^{(j)}, j = \overline{1,k}$, по формуле:

$$\mathrm{C} = \mathrm{ZW} \qquad (11)$$

В частности,

$$\mathrm{C}^{(\mathrm{s})} = \mathrm{W}_1^{(\mathrm{s})}\mathrm{Z}^{(1)} + \cdots + \mathrm{W}_\mathrm{k}^{(\mathrm{s})}\mathrm{Z}^{(\mathrm{k})} \qquad (12)$$

Каждая из главных компонент $C^{(s)}, s = \overline{1,k}$, является линейной комбинацией стандартизованных признаков $Z^{(j)}, j = \overline{1,k}$, причем координаты собственного вектора с номером s определяют главную компоненту с номером s.

Одно из основных геометрических свойств главных компонент формулируется как свойство сохранения суммарной дисперсии.

$$\sum_{\mathrm{j}=1}^{\mathrm{k}} \mathrm{Var}\left(\mathrm{Z}^{(\mathrm{j})}\right) = \sum_{\mathrm{s}=1}^{\mathrm{k}} \mathrm{Var}\left(\mathrm{C}^{(\mathrm{s})}\right) = \sum_{\mathrm{j}=1}^{\mathrm{k}} \lambda_\mathrm{j} = \mathrm{k} \qquad (13)$$

В целях снижения размерности пространства признаков главные компоненты, имеющие наименьшую дисперсию, исключают из рассмотрения. Принятые к рассмотрению первые m главных компонент есть компоненты $C^{(s)}, s = \overline{1,m}$. Они соответствуют собственным векторам $W^{(s)}, s = \overline{1,m}$, и имеют дисперсии $\lambda_s, s = \overline{1,m}$, соответственно. Тогда величина

$$\delta = \frac{\lambda_1 + \lambda_2 + \cdots + \lambda_m}{k} \tag{14}$$

показывает долю суммарной дисперсии, объясненной на основе m выделенных главных компонент. Далее будем называть ее *суммарная объясненная дисперсия*. Чем ближе значение δ к единице, тем лучше пространство размерности m воспроизводит разброс объектов, имеющий место для пространства размерности k, и тем более оправдано снижение размерности.

В частности, доля *объясненной дисперсии* s-й главной компонентой определяется как

$$\delta^{(s)} = \frac{\lambda_s}{k} \tag{15}$$

Чем ближе значение $\delta^{(s)}$ к единице, тем данные более разбросаны в проекции на компоненту $C^{(s)}$, тем больше информации содержится в ней.

В силу того, что собственные векторы ортонормированы $W^{-1} = W^T$. Из (11) получим

$$Z = CW^T \tag{16}$$

После снижения размерности формируются «части» исходных стандартизованных признаков:

$$\overline{Z^{(j)}} = W_j^{(1)} C^{(1)} + \cdots + W_j^{(m)} C^{(m)}, j = \overline{1,k} \tag{17}$$

Доля дисперсии признака $Z^{(j)}$, объясняемая с помощью m выделенных компонент, называется *общностью* признака $Z^{(j)}$ и обозначается как h_j^2

$$\begin{aligned} h_j^2 &= \left[W_j^{(1)}\right]^2 \mathrm{Var}\left(C^{(1)}\right) + \cdots + \left[W_j^{(m)}\right]^2 \mathrm{Var}\left(C^{(m)}\right) = \\ &= \left[W_j^{(1)}\right]^2 \lambda_1 + \cdots + \left[W_j^{(m)}\right]^2 \lambda_m, \qquad j = \overline{1,k} \end{aligned} \tag{18}$$

Каждая из общностей $h_j^2, j = \overline{1,k}$, находится в границах $0 \le h_j^2 \le 1$. Чем ближе значение h_j^2 к единице, тем лучше пространство размерности m воспроизводит разброс объектов по признаку $Z^{(j)}$, имеющий место для пространства размерности k, и тем более оправдано оставить этот признак при снижении размерности.

В данной работе снижение размерности пространства признаков будет происходить за счет удаления признаков с наименьшими показателями общности.

4. Исследовательская часть

4.1 Описание базы данных

База данных изображений больных и здоровых листьев томатов была взята из открытого источника Plant Village [5]. Она включает в себя 6000 отсегментированных изображений, которые подразделяются на 6 классов, описанных в таблице 1.

Таблица 1. Описание классов исследуемой базы данных

Класс	Тип заболевания	Кол-во изображений
Healthy – здоровые листья	-	1000
Bacterial spot – бактериальная пятнистость	Бактериальное	1000
Early blight – ранняя гниль	Грибковое	1000
Late blight – поздняя гниль	Грибковое	1000
Septoria leaf spot – септория	Грибковое	1000
Yellow Leaf Curl Virus – вирус желтого скручивания листьев	Вирусное	1000

Описание изображений:

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

4.2 Инструменты исследования

Разработка программного комплекса ведется на языке *Python*. Такой выбор языка программирования обусловлен наличием готовых библиотек для работы с многомерными массивами (тензорами). В частности, в данной работе используются библиотеки *numpy* и *pytorch,* который позволяет обернуть процесс извлечения признаков в единую модель.

Для вычисления GLCM и ее признаков, описанных в пункте 3.1 «Матрица GLCM», используются функции библиотеки *scikit-image*.

Обучение и тестирование классификаторов, а также снижение размерности пространства признаков происходит с помощью инструментов библиотеки *scikit-learn*.

4.3 Извлечение признаков

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.

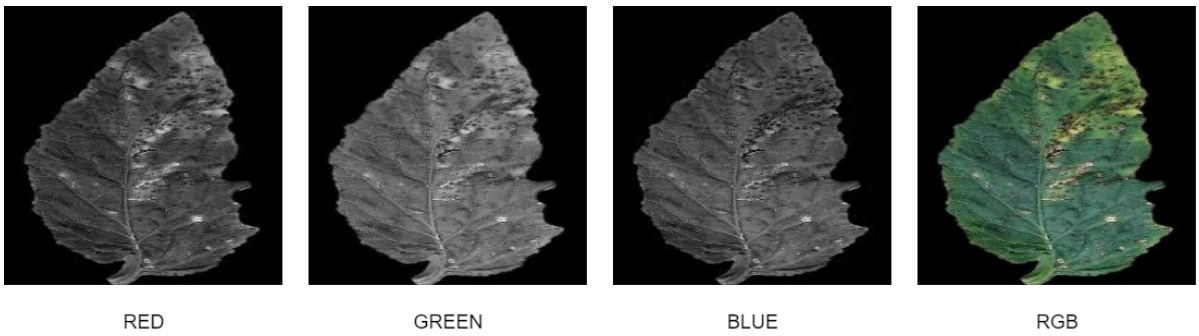

Рисунок 2. Разложение по каналам изображения листа томата, пораженного заболеванием Bacterial spot

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. На основе работ [12] в данном исследовании будет предпринята попытка вычислить NDVI образы для всех изображений, в качестве дополнительного признака.

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* понимаются признаки, извлеченные неким оператором над всем изображением сразу. Под *локальными* признаками понимаются признаки, извлеченные под малой, по сравнению с размерами изображения, маской инструмента.

4.3.1 Метод квантования изображений

Извлечение текстурных признаков для изображений, имеющих 256 градаций серого, требует большой вычислительной работы. Как правило точность признаков, полученных

таким способом, является избыточной. Поэтому, для уменьшения вычислительной нагрузки в данной работе было решено сжать изображения путем их квантования на 5 уровней серого. Предполагается, что такой точности признаков хватит для классификации.

Для квантования изображений использовались фиксированные уровни дискретизации, полученные из распределения пикселей с ненулевой интенсивностью среди всех изображений в красном канале:

$$\mathrm{meanR}^* = 85.29, \qquad \mathrm{stdR}^* = 53.73, \tag{19}$$

где meanR*, stdR* - среднее и стандартное отклонение яркости ненулевых пикселей по всей выборке. В качестве адаптивного алгоритма квантования применялись следующие границы:

$$\mathrm{Q} = \begin{cases} [0] - \text{0-й уровень} \\ [1, \mathrm{meanR}^* - \mathrm{stdR}^*) - \text{ 1-й уровень} \\ [\mathrm{meanR}^* - \mathrm{stdR}^*, \mathrm{meanR}^*) - \text{ 2-й уровень} \\ [\mathrm{meanR}^*, \mathrm{meanR}^* + \mathrm{stdR}^*) - \text{ 3-й уровень} \\ [\mathrm{meanR}^* + \mathrm{stdR}^*, 255] - \text{ 4-й уровень} \end{cases}, \tag{20}$$

Для нулевого уровня яркости выделен отдельный уровень квантования. Это сделано для того, чтобы не учитывать вклад фоновых (нулевых) пикселей в гистограмму, подаваемую на вход классификатору, поскольку они не несут полезной информации.

4.3.2 Описание полного вектора признаков

Для анализа изображений выбран следующий вектор признаков:

$$\mathrm{features} = [\mathrm{STAT}, \mathrm{HIST}, \mathrm{GLCM}] \tag{21}$$

составленный из:

1. Статистических характеристик изображения в R-канале:

$$\mathrm{STAT} = \left[\frac{\mathrm{meanR}}{\mathrm{meanR}^*}, \frac{\mathrm{stdR}}{\mathrm{stdR}^*}, \frac{\mathrm{maxR} - \mathrm{meanR}}{\mathrm{stdR}^*}, \frac{\mathrm{meanR} - \mathrm{minR}}{\mathrm{stdR}^*}\right], \tag{22}$$

   где *meanR*, *maxR*, *minR* и *stdR* - среднее, максимальное, минимальное значения и стандартное отклонение значений яркостей изображения в красном канале, *meanR** и *stdR** обозначены в равенстве (19)

2. Нормированной гистограммы квантованного изображения без учёта нулевых элементов:

$$\mathrm{HIST} = [\mathrm{N}(\mathrm{Q_i})], \mathrm{i} = \overline{1,4} \tag{23}$$

   где $N(Q_i)$ - нормированное значения уровня Q_i (20) на рассматриваемом участке изображения

3. Признаков, полученных из GLCM-матрицы квантованного изображения в красном канале, для различных расстояний d и углов φ:

$$\mathrm{GLCM} = [\mathrm{contrast}, \mathrm{homogeneity}, \mathrm{energy}, \mathrm{corrrelation}, \mathrm{entropy}], \quad (24)$$

$$\mathrm{d} = \{1\mathrm{px}, 2\mathrm{px}, 4\mathrm{px}\}$$

$$\varphi = \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\} \quad (25)$$

GLCM-матрица для каждого угла и расстояния квантованного изображения имеет размер 5x5. Всего в работе рассмотрено 3 расстояния и 4 направления (25). Поэтому каждый из текстурных признаков (1-5) представлен 12 числами и dim(GLCM) = 60.

Вектора *features* глобальных и локальных признаков далее будем обозначать как $features_{glob}$ и $features_{loc}$ соответственно.

4.3.3 Алгоритм извлечения вектора локальных признаков

Для вычисления локальных признаков размер маски был взят равным 17x17. Алгоритм извлечения локальных признаков *features* (23) для конкретного изображения пошагово описан в таблице 2.

Таблица 2. Алгоритм извлечения признаков

Шаг	Действие	Размерность
1. Подготовка данных и вычисление статистических характеристик STAT (22)		
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 в каждом окне.	[4, 61, 61]
1.6	Квантование изображения на 5 уровней (22). Получение квантованного изобр. *Q*.	[1, 61, 61, 17, 17]
1.7	Изменение размера *Q*.	[3721, 17, 17]
2. Вычисление характеристик HIST (23) и GLCM (24) в каждом из 3721 окон		
2.1	Вычисление нормированной гистограммы HIST для *Q*.	[4]
2.2	Вычисление нормированной, симметричной *GLCM*-матрицы.	[5, 5, 3, 4]
2.3	Вычисление признаков GLCM	[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]

Блок операций 2 выполняется по всему изображению параллельно. Модель извлечения признаков реализована средствами библиотеки *pytorch*.

На рисунках 6 и 7 изображены примеры исходного изображения здорового и больного Bacterial spot листьев томатов, а рядом с ними результаты извлечения 68 признаков $features_{loc}$ под маской инструмента. Данные изображения возможно получить в п. 3.3 (табл. 2) выполнения модели извлечения локальных признаков из оригинальных изображений.

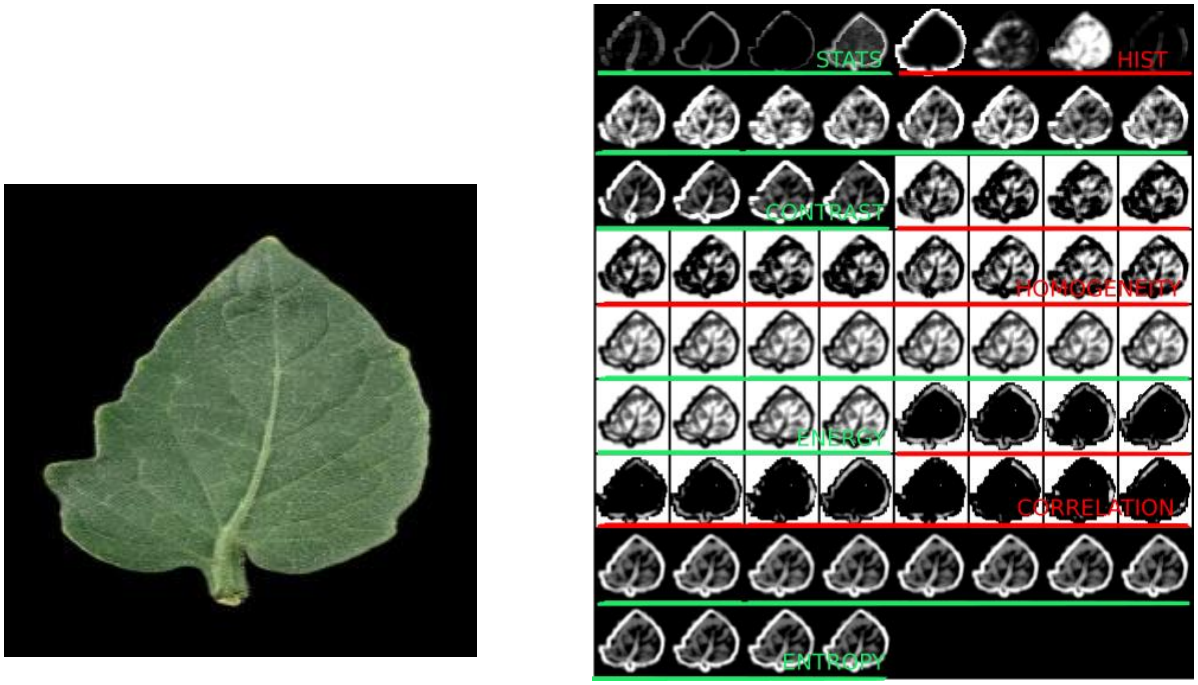

Рисунок 6. Исходное изображение и вектор изображений features для здорового листа

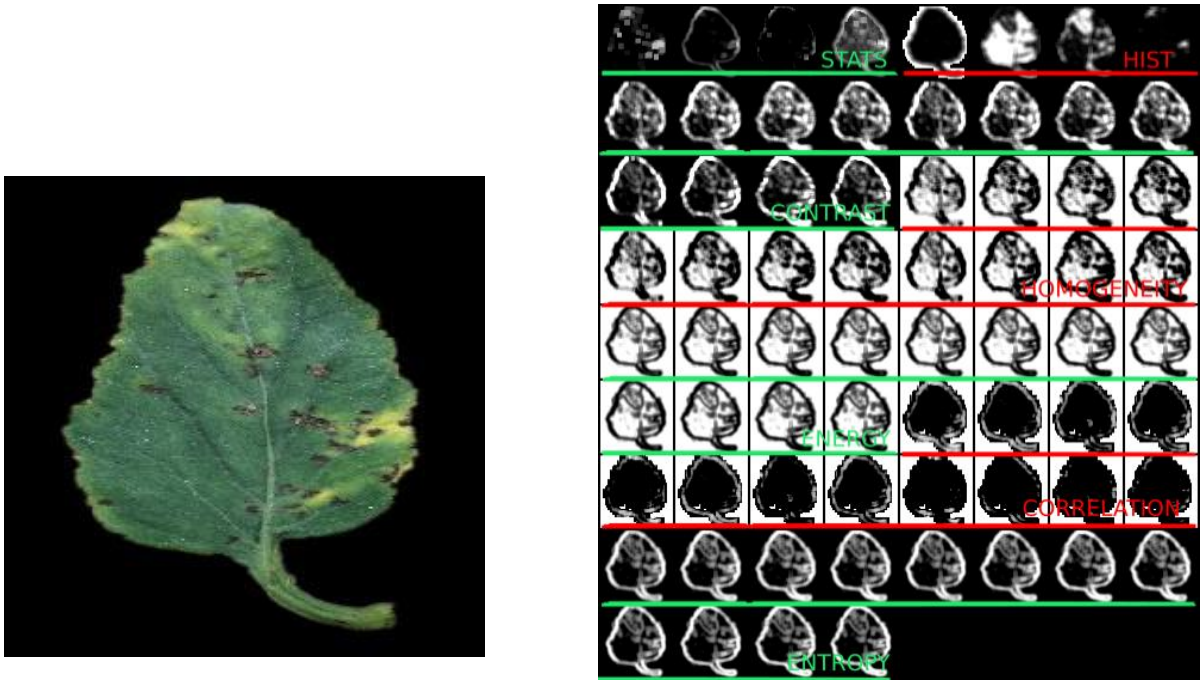

Рисунок 7. Исходное изображение и вектор изображений features для больного Bacterial spot листа

4.3.4 Результаты работы классификаторов на глобальных и локальных признаках

Результаты экспериментов по применению различных алгоритмов классификации к наборам локальных $features_{loc}$ и глобальных $features_{glob}$ признаков описаны в предыдущей работе [13], здесь они будут продублированы для сравнений с дальнейшими результатами экспериментов.

Для изучения были выбраны следующие алгоритмы: дерево решений (DTC), метод k-ближайших соседей (KNN), случайный лес (RF), метод опорных векторов (MSVM) и одноуровневый персептрон (1 Layer MLP). К тестовой выборке применялись алгоритмы с параметрами, дающими наилучшие результаты при кросс-валидации на тренировочной выборке. Параметры выбранных алгоритмов представлены в таблице 3.

Таблица 3. Параметры алгоритмов классификации на векторе *features*

Название метода	Параметры для scikit-learn
DTC	criterion='entropy', max_depth=10
KNN	n_neighbors=9, metric='euclidean', weights='uniform'
RF	n_estimators=300, criterion='entropy', max_depth=10
MSVM	C = 5, kernel='rbf', gamma='auto'
1 Layer MLP	hidden_layer_sizes=(220), activation='relu', max_iter=10000, shuffle=True, solver='adam'

Эти алгоритмы применялись как ко всем извлечённым признакам *features*, так и к их частям: STAT, HIST и GLCM.

В предыдущей работе [13], установлено, что процедура стандартизации признаков в среднем позволяет выиграть несколько процентов качества, поэтому в этом исследовании будут рассматриваться только стандартизованные признаки.

Результаты мультиклассовой классификации на векторах $features_{loc}$ и $features_{glob}$ предоставлены в Приложении А. Выделены 3 лучших и 3 худших алгоритма для каждого из векторов.

Анализ результатов классификации (см. Приложение А.), позволяет сделать следующие выводы:

- Дерево решений показало один из худших результатов. Хороший результат даёт лишь ансамбль деревьев решений – случайный лес.
- Стабильно лучшие результаты показывали одноуровневый персептрон и метод опорных векторов над полным вектором признаков *features*.
- Алгоритмы классификации на основе характеристик STAT+HIST, показывают низкое качество. Текстурные характеристики GLCM вносят заметный вклад в улучшение работы алгоритмов.

- Использование локальных признаков $features_{loc}$ до 4% улучшает качество работы всех классификаторов.
- Классификация на локальных признаках STAT+HIST приобретает до 15% качества относительно классификации на глобальных признаках STAT+HIST.
- Классификация на глобальных признаках GLCM приобретает до 3% качества относительно классификации на локальных признаках GLCM. Правда при этом она всё ещё остаётся хуже классификации на полном наборе глобальных признаков *features*.

На основе последних трех пунктов, можно сказать, что глобальные признаки $features_{glob}$ проигрывают в качестве за счёт потерь на признаках STAT + HIST, поскольку глобальные GLCM признаки в рамках рассмотренных моделей наоборот, показывают себя лучше локальных. Поэтому ставится предположение что можно применять комбинированные локальные и глобальные признаки без потери точности.

О совместном использовании глобальных и локальных признаков пойдет речь в следующей главе.

4.4 Комбинирование глобальных и локальных признаков

В данном разделе исследуется возможность заменить некоторые локальные признаки глобальными, так чтобы не произошло большой потери точности работы классификаторов. Использование глобальных признаков позволит значительно уменьшить вычислительную сложность их извлечения. Вектор признаков, в котором используются и локальные, и глобальные данные будем называть *комбинированным* и обозначим как $features_{comb}$.

4.4.1 Формирование вектора комбинированных признаков

После извлечения, для каждого типа признаков – STAT, HIST, GLCM полезно узнать какие их представления окажутся более информативными: глобальные или локальные. Для этого используем МГК.

Определим число главных компонент m, обеспечивающее высокую суммарную дисперсию (более 80%). Глядя на рис. 8 можно сказать, что локальные признаки при малом m описываются лучше, чем глобальные. Это означает что облако локальных данных имеет более простую структуру, т.е. лучше проецируется на пространства малой размерности.

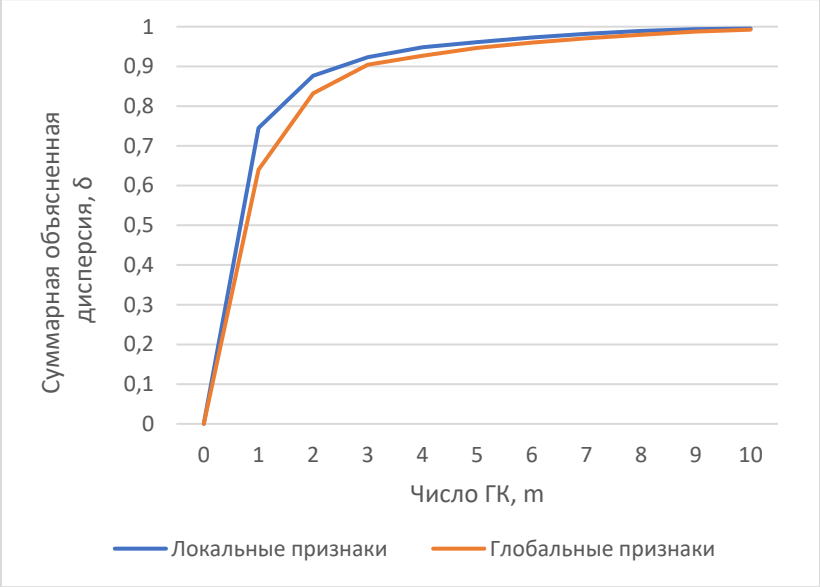

Рисунок 8. График сохранения информации главными компонентами

В качестве лучшего *m*, обеспечивающего высокую суммарную дисперсию, выбирается m=3, поскольку «потеря информации» для глобальных и локальных признаков будет одного порядка и составит ≈9%.

С помощью МГК признаки из 68-мерного пространства признаков проецируются на 3-мерное пространство в базисе главных компонент. Эта процедура проводится отдельно над пространствами глобальных и локальных признаков. Полученные в результате сжатия размерности показатели общностей для каждого из признаков отражают насколько хорошо тот или иной признак позволяет сжать пространство.

Локальные признаки обладают достаточно бóльшей общностью лишь для признаков STAT, HIST и correlation (см. Приложение D, рис. D1). Поэтому, для уменьшения вычислительной нагрузки целесообразно все остальные признаки вычислять глобально.

Заметим, что, если усреднить полученные на рис. С1 общности для каждой группы признаков STAT, HIST, GLCM, то результат (рис. 9), совпадает с предположением, сделанным в предыдущей работе [13] о том, что совместное использование локальных статистических STAT + HIST и глобальных текстурных GLCM признаков возможно даст лучшие результаты (во всяком случае не хуже).

Таким образом, выделен комбинированный вектор признаков *features_comb*

$$\begin{gathered}\text{features}_{\text{comb}} = [\text{STAT}_{\text{loc}}, \text{HIST}_{\text{loc}}, \text{GLCM}_{\text{comb}}], \\ \text{GLCM}_{\text{comb}} = [\text{contrast}, \text{homogeneity}, \text{energy}, \text{entropy}]_{\text{glob}} + [\text{correlation}]_{\text{loc}},\end{gathered} \tag{26}$$

который включает в себя локальные признаки STAT, HIST и correlation. Все остальные признаки: contrast, homogeneity, energy, и entropy – глобальные.

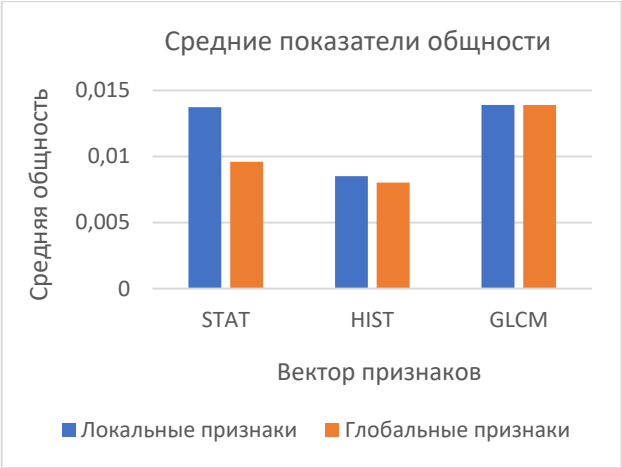

Рисунок 9. Средние показатели общности для каждой группы признаков

4.4.2 Результаты использования вектора комбинированных признаков

Проведем анализ точности работы классификаторов на комбинированном векторе признаков $features_{comb}$. Показатели F-Score для комбинированного вектора $features_{comb}$ и для вектора локальных данных $features_{loc}$ оказались идентичными (см. Приложение В). Точность классификации изменилась в пределах 1%.

За счет отсутствия упадка качества работы классификаторов на комбинированном векторе признаков, становится возможным использования вместо локальных GLCM признаков contrast, homogeneity, energy, и entropy их глобальные аналоги. В связи с этим уменьшится вычислительная нагрузка извлечения признаков из исходных изображений.

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

4.5 Внедрение NDVI статистики

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, в качестве дополнительного признака к вектору $features_{comb}$.

NDVI рассчитывается в красном и инфракрасном диапазоне следующим образом:

$$\mathrm{NDVI} = \frac{\mathrm{NIR} - \mathrm{RED}}{\mathrm{NIR} + \mathrm{RED}}, \tag{27}$$

NDVI принимает свои значения в диапазоне [-1, 1]. Природные объекты, не связанные с растительностью, имеют отрицательное значение NDVI. Отсутствие зеленых растений на изображении дает NDVI близкий к 0, а близкое к +1 (0,7–0,9) значение показателя указывает на максимально возможную густоту зеленых здоровых листьев. Поэтому рамках исследуемой задачи, будут рассматриваться области с малым значением показателя NDVI.

4.5.1 Извлечение NDVI образов изображений

Для вычисления NDVI необходимо обладать информацией о листе в инфракрасном диапазоне, но поскольку мы работаем с RGB изображениями, то в чистом виде информации об этом спектральном диапазоне нет.

Для преобразования RGB изображения в NIR используем метод, предложенный в работе H.G.Dietz «RGB+NIR extraction from a single RAW image» [12]. Способ применения этого метода описан в таблице 4 «Алгоритм извлечения NDVI изображения». Используемые матрицы преобразования цветовых пространств:

- Матрица преобразования RGB_NIR в RGB:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} -1.382995 & 0.249841 & -0.159960 & 1.289882 \\ -0.265864 & -1.457404 & 0.933098 & 0.809020 \\ -1.423661 & 0.167074 & 1.381977 & -0..097471 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \\ NIR \end{bmatrix} \tag{28}$$

  В данной работе используется обратное преобразование RGB в RGB_NIR, поэтому вместо матрицы преобразования (28) используется ее псевдообратная.

- Матрица преобразования RGB_NIR в NIR:

$$[NIR] = [15.097626 \quad 12.247668 \quad -9.044717 \quad -8.382289] \begin{bmatrix} R \\ G \\ B \\ NIR \end{bmatrix} \tag{29}$$

Таблица 4. Алгоритм извлечения NDVI изображения

Шаг	Операция
1	Извлечение красного канала из RGB изображения
2	Преобразование изображения из RGB в RGB_NIR используя псевдообратную матрицу для матрицы A
3	Преобразование изображения из RGB_NIR в NIR используя коэффициенты вектора B
4	Нормировка на 256 уровней NIR изображения.
5	Вычисление NDVI образа изображения по формуле (29)

Чтобы убедиться в корректности алгоритма извлечения NDVI изображений (табл. 4) рассмотрим примеры полученных RED, NIR и NDVI изображений для здорового листа томата, и листа, страдающего заболеванием Bacterial Spot (рис. 10-11).

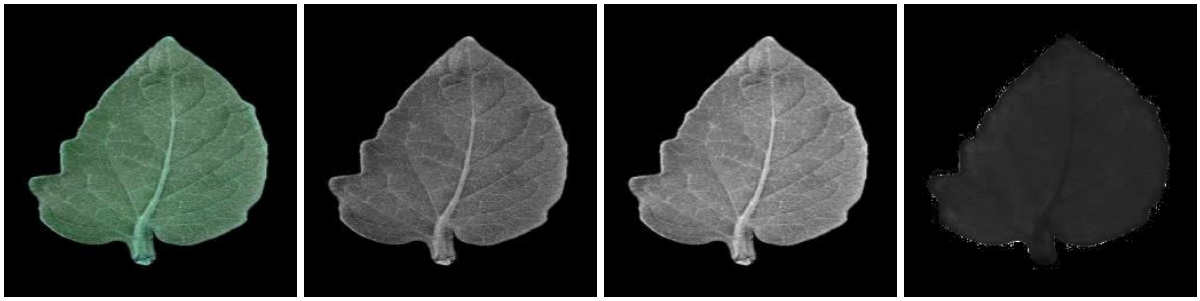

Рисунок 10. RGB, RED, NIR и NDVI изображения здорового листа томата

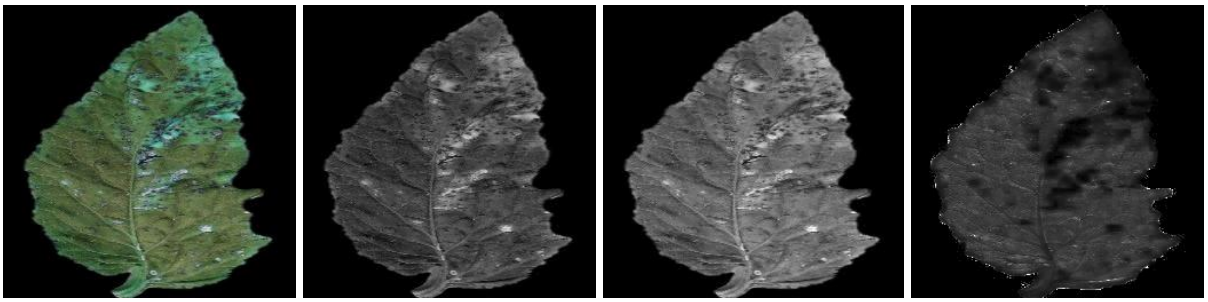

Рисунок 11. RGB, RED, NIR и NDVI изображения больного Bacterial spot листа томата

Проанализируем рисунки 10-11 с точки зрения физики.

Поскольку пигмент листьев растений, хлорофилл, сильно поглощает видимый свет (от 0,4 до 0,7 мкм), 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 сильно отражает свет в ближнем инфракрасном диапазоне (от 0,7 до 1,1 мкм). Поэтому для NIR изображения здоровые области листа характеризуются светлыми участками, «омертвевшие» – темными.

Полученные изображения (рис. 10-11) корректно описывают вышеописанные физические явления. Как и ожидалось на NDVI изображениях область заболевания описывается темными участками, то есть малыми значением индекса NDVI.

4.5.2 Результаты использования вектора NDVI признаков

Поскольку текстура зараженной области листа «съедается» на NDVI изображении (см. рис. 11), то для классификации будет использован только статистический вектор признаков: STAT (24) и HIST (25), вычисленный локально на NDVI изображении. Такой вектор в дальнейшем будем называть *вектором NDVI признаков* (30). Размер вектора NDVI признаков dim(NDVI) = 8.

$$NDVI_{loc} = [STAT_{ndvi}, HIST_{ndvi}] \quad (30)$$

Поскольку к вектору $features_{comb}$ добавляются новые признаки - $NDVI_{loc}$, перед тестированием классификаторов была проведена кросс-валидация на тренировочных данных. Параметры алгоритмов представлены в таблице 5.

Таблица 5. Параметры алгоритмов классификации на векторе features+NDVI

Название метода	Параметры для scikit-learn
DTC	criterion='entropy', max_depth=10
KNN	n_neighbors=9, metric='manhattan', weights='distance'
RF	n_estimators=320, criterion='entropy', max_depth=15
MSVM	C = 7, kernel='rbf', gamma='auto'
1 Layer MLP	hidden_layer_sizes=(240), activation='relu', max_iter=10000, shuffle=True, solver='adam'

Результаты классификации доступны в приложении С. При добавлении $NDVI_{loc}$ признаков точность классификации на полном векторе данных упала только у одноуровневого персептрона (на 4%). Выполняя классификацию только на NDVI признаках можно достичь не более 16% F-score, что катастрофически мало. Эти признаки оказались бесполезны в задаче классификации.

Поскольку показатель NDVI напрямую используется для оценки качества растительного покрова, то прежде, чем исключать $NDVI_{loc}$ признаки из рассмотрения, интересно узнать, какой именно вклад они вносят на полном наборе признаков. Для этого в следующей главе оценим информативность каждого из признаков вектора $features_{comb} + NDVI_{loc}$.

4.6 Понижение размерности пространства признаков

На данном этапе исследования получен вектор признаков $features_{comb} + NDVI_{loc}$ длинной в 76 элементов. Это достаточно большой набор данных, который сложно интерпретировать и сложно вычислять. Он содержит в себе GLCM признаки для всех значений параметров d и φ (25), а изначально ставилась задача поиска наиболее оптимальных из них. Также, в главе 4.5 «Внедрение NDVI статистики» показано, что признаки $NDVI_{loc}$ не дали большого прироста точности алгоритмов классификации, в связи с этим ставится вопрос о значимости NDVI признаков в решении задачи.

В качестве предварительной обработки дальнейшего анализа сначала проверим *релевантность*, а затем *избыточность* всех измерений.

Проверка на релевантность, подразумевает исключение признаков с нулевой дисперсией, поскольку они не содержат информации, которая помогает отличать объекты друг

от друга. В нашем наборе данных такие признаки отсутствуют. Дисперсии лежат в диапазоне [0.02; 0.67].

Проверка на избыточность, заключается в анализе корреляционной зависимости признаков и при необходимости удалении сильно коррелированных признаков. Такая операция не всегда приводит улучшению работы алгоритмов классификации, поэтому требуется провести проверку результатов классификации на векторе без коррелированных признаков.

В данной работе, сильной принята корреляция более чем 0.9. Сильная корреляция оказалась у признаков (см. Приложение Е. рис Е2):

- ENTROPY и ENERGY при любых параметрах d и φ. Оставим в векторе признаки ENERGY, поскольку они проще в вычислении.
- HOMOGENEITY и CONTRAST. Однако не для всех параметров d и φ. Решение об удалении одного из них, примем после определения конкретных значений d и φ.
- STAT_STD с признаками STAT_MAX и STAT_MIN. Поэтому удалим STAT_STD.
- NDVI_STAT_MEAN и NDVI_STAT_MIN, NDVI_STAT_STD и NDVI_STAT_MAX. Оставим первые из них.

Тестирование классификаторов на векторе без коррелированных признаков проводилось при параметрах моделей согласно таблице 5. Поскольку набор измерений изменился только для векторов STAT, GLCM и $NDVI_{loc}$, то результаты классификации будут рассматриваться только для них (см. Приложение F, табл. F1).

У всех классификаторов наблюдается упадок точности на векторе STAT, в среднем на 12%. Следовательно, несмотря на сильную корреляцию, признак STAT_STD вносит существенный вклад при классификации заболеваний. Показатели F-score на наборах GLCM и $NDVI_{loc}$ признаков не изменились.

Лучшим решением будет удаление только сильно коррелированных GLCM и NDVI признаков: ENTROPY, NDVI_STAT_MIN и NDVI_STAT_MAX. Полученный, в результате их удаления, вектор признаков обозначим как $(features_{comb} + NDVI_{loc})'$.

Отметим, что признаки ENTROPY и ENERGY внутри своей группы параметров d и φ коррелированы для каждой пары (d, φ). Вероятно, при классификации конкретный выбор параметров (27) на этих признаках не имеет большого значения.

Благодаря проверке на избыточность размер вектора данных был снижен с 76 признаков, до 62 признаков. Для дальнейшего сжатия используем МГК в пространстве $(features_{comb} + NDVI_{loc})'$ признаков.

Число главных компонент m выберем из условия сохранения суммарной объясненной дисперсии δ более 80%. Согласно рис. 11 в качестве такого m можно взять m=3, при этом доля суммарной объясненной дисперсии составит $\delta \approx 0{,}82$.

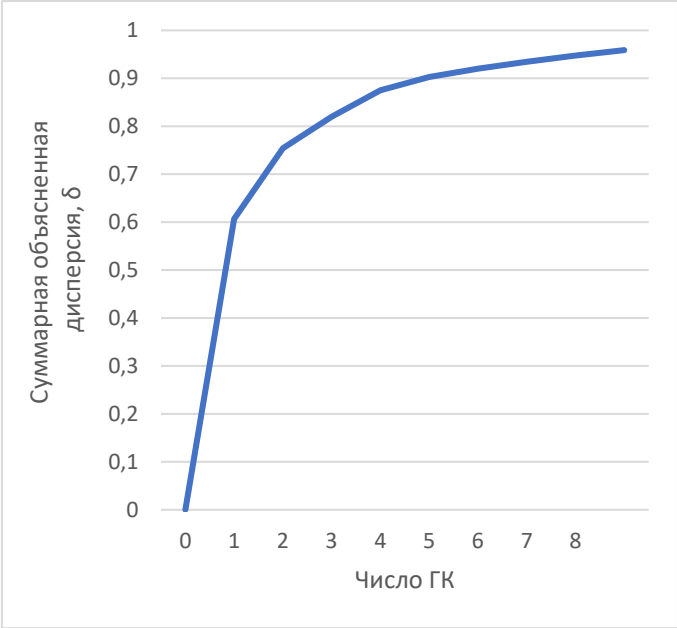

Рисунок 11. График, отражающий долю сохраненой информации features_comb+NDVI'.

Анализ показателей общности для каждого признака из вектора $(features_{comb} + NDVI_{loc})'$ при сжатии пространства с использованием 3-х главных компонент (см. Приложении G), позволяет сделать следующие выводы:

- Все локальные признаки STAT содержат много информации. Поэтому ни один из них не будет удален.
- Полезными из набора HIST оказались признаки HIST_1 и HIST_3. Следовательно, квантованное изображение хорошо описывается усреднением локальных показателей уровней HIST_1 = [meanR* - stdR*, meanR*) и HIST_3 = [meanR* + stdR*, 255].
- Все текстурные признаки оказались важными. Однако, по величине общностей, все-таки можно выделить наиболее оптимальные из них (см. табл. 6).

Таблица 6. Лучшие GLCM параметры d и φ

Признак	Чувствительность к		Лучший d	Лучший φ
	Расстоянию d	Направлению φ		
ENERGY	-	-	1, 2, 4	0, π/4, π/2, 3π/4
HOMOGENEITY	+	+	1, 2	π/4, 3π/4
CONTRAST	+	+	1, 2	π/4
CORRELATION	+	+	1, 2, 4	π/2, π/4

- NDVI признаки, извлеченные способом, описанным в Главе 4.5 «Внедрение NDVI статистики», не играют роли при делении образцов на классы, то есть не помогают разделить образцы на классы. В связи с этим, исключим их из набора признаков.
- При тестировании будут использованы (d, φ) = (1, π/4) для всех текущих GLCM-признаков: ENERGY, HOMOGENEITY, CONTRAST и CORRELATION.

Выбор конкретных параметров GLCM d и φ для каждого их признаков energy, homogeneity, contrast и correlation происходил с учетом результатов табл. 6 и минимизации потери точности алгоритмов классификации.

Таким образом выделен *оптимальный* вектор признаков (см. табл. 7) минимальной длины, максимально описывающий всю полезную информацию $features_{opt}$.

Таблица 7. Описание вектора $features_{opt}$

Признак	Способ извлечения	Параметры	Размер
STAT	локально	-	4
HIST_1 и HIST_3	локально	-	2
ENERGY	глобально	(d, φ) = (2, π/4)	1
HOMOGENEITY	глобально	(d, φ) = (1, 3π/4)	1
CONTRAST	глобально	(d, φ) = (2, π/4)	1
CORRELATION	локально	$(d, \varphi)_1 = (1, \pi/2)$, $(d, \varphi)_2 = (2, \pi/2)$ $(d, \varphi)_3 = (4, \pi/4)$	3

Тестирование классификаторов проводилось на параметрах моделей, указанных в таблице 5. Сравним показатели F-score при классификации на векторе $features_{opt}$ с вектором $features_{comb}$, который показал лучшие результаты. Обратимся к приложению F, табл. F1.

При удалении признаков из набора HIST, показатели F-score для набора STAT + HIST упали незначительно (менее 2%). Усечение набора текстурных признаков GLCM привело к значительной потере точности при классификации только по GLCM на 10% для SVM и 1 Layer MLP. Теперь классификация на статистических признаках выигрывает в среднем на 6% в точности, чем классификация на текстурных признаках вектора $features_{opt.}$ На полном наборе признаков показатели F-score уменьшились в пределах 6%, что не критично, с учетом того, что вектор $features_{comb}$ длинной в 68 признаков сжат до 12 признаков.

Таким образом, в данной главе с помощью проверки на избыточность и МГК выявлены лучшие GLCM параметры d и φ (см. табл. 6) и определены наиболее информативные признаки (см. табл. 7), образующие вектор $features_{opt}$. Так, вектор признаков $features_{comb} + NDVI_{loc}$ длинной в 76 элементов сжат до 12 элементов. При этом для всех классификаторов потеря точности на полном наборе признаков не превосходит 6%, по сравнению с наилучшими достигнутыми результатами на полном векторе признаков.

5. Результаты исследования

Одной из задач работы являлась проверка возможностей различных алгоритмов классификации на одном и том же наборе данных с целью выбрать наилучший из них. В процессе исследования из полного набора статистических и текстурных признаков *features* (21) выделен вектор $features_{opt}$ (см. табл. 8) минимальной длины, максимально описывающий полезную информацию. Поэтому далее для каждого класса больных растений описаны результаты эксперимента по применению алгоритмов классификации DTC, MSVM, RF, KNN и 1 Layer MLP на векторе признаков $features_{opt}$.

В приложении I представлена таблица с подробными результатами мультиклассовой классификации. После выполнения тестирования различных классификаторов можно сделать несколько выводов:

- Классификация на статистические признаках STAT + HIST выигрывает в среднем
- Лучшими классификаторами остались MSVM и 1 Layer MLP, которые дают 80% F-Score в среднем по всем классам. Высокий показатель для SVM говорит о наличии явных границ принятия решений непосредственно из обучающих данных. То есть в пространстве признаков данные и классах пересекаются не критично.
- Худший результат, чуть выше 50% F-Score дает дерево решений. Следовательно, на признаках $features_{opt}$ нельзя создать простое правило классификации, основанное на нескольких измерениях.
- Одинаковые результаты дали KNN и RF. Невысокая точность RF, объясняется тем, что RF – объединение результатов множества деревьев решений, а само дерево решений показало низкий результат. В случае KNN, информация о соседях, в манхэттенской метрике (см. табл 5.), также не позволяет с высокой точностью определить класс образца, значит образцы в пространстве признаков слабо подвержены кластеризации.
- Лучше всех подвергаются классификации здоровые листья и листья, пораженные заболеваниями *Yellow Leaf Curl Virus* и *Bacterial spot*. Точность их выявления не менее 80%. Классы *Late blight* и *Septoria leaf spot* классифицируется хуже, с показателями точности порядка 70%. *Early blight* является самым трудным для классификации, его показатели составляют 64% в лучшем случае.

На текущий момент осталась незатронутая информация, содержащаяся в синем и зеленом каналах изображения. Возможно использование этих данных позволит повысить результаты классификации и сделать модель более интерпретируемой.

6. 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 и метод снижения размерности пространства признаков МГК. Получены базовые знания в области машинного обучения. Данные темы кратко описаны в главе 3. «Теоретическая часть». Помимо теории, освоен с нуля язык программирования Python. Улучшены навыки работы с многомерными данными и их анализа.

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 (см. п. 4.1 «Описание базы данных»).

Одним из наиболее трудоёмких моментов в работе, с точки зрения реализации, был процесс извлечения признаков из изображений. Всего было получено 2 типа признаков: статистические и текстурные. При этом исследовалось влияние на результат способа получения этих признаков: глобального и локального. Как показали результаты экспериментов (см. прил. А), глобальные признаки проигрывают в качестве в среднем на 3 – 4% при классификации на полном наборе данных. Это происходит в основном за счёт потерь на признаках STAT (22) и HIST (23). Однако глобальные GLCM (24) признаки в рамках рассмотренных моделей наоборот, показывают себя лучше усреднённых локальных. Этот факт привел к созданию вектора комбинированных признаков $features_{comb}$ (26), в котором, с помощью МГК, локальные текстурные признаки, были заменены глобальными аналогами, обладающими большей информативностью.

В процессе исследования была предпринята получить NDVI образы изображений и извлечь из них статистические признаки STAT и HIST, предполагая, что их внедрение должно повысить точность результатов классификации. Однако такого не произошло, и эти признаки оказались непригодны для данной задачи классификации. Доказательства этих фактов находятся в разделе 4.5 «Внедрение NDVI статистики».

Так же в ходе выполнения работы было предположено, что для диагностирования болезни достаточного использовать только красный канал изображений. Высокие результаты в приложении А, доказывают справедливость этого предположения.

В главе 4.6 выполнена задача выявления наиболее информативных признаков. С использованием проверки на избыточность и МГК набор признаков $features_{comb}$ (26) был сжат до 12 признаков в вектор $feature_{opt}$ (см. табл. 7) и выделены лучшие параметры GLCM d и φ (см. табл. 6).

В итоге лучшим классификатором оказалась простейшая нейронная сеть – одноуровневый персептрон, показатель F-score которого составляет 78% на векторе. При этом для достижения такой точности ему понадобился весь вектор признаков $feature_{opt}$.

Вторым по качеству алгоритмом машинного обучения оказался метод опорных векторов, который дает 76% F-score, при этом используя весь вектор признаков $feature_{opt}$.

В дальнейшем, для повышения показателей точности классификации, в дополнение к $feature_{opt}$, вычисленному в красном канале, предлагается использовать информацию, содержащуюся в синем и зеленом каналах изображения. Также возможно внедрение статистических признаков матрицы GLCM.

Таким образом, все поставленные задачи выполнены. Установлен лучший классификатор болезней листьев томатов – одноуровневый персептрон и определен минимальный набор из лучших признаков – $feature_{opt}$ (см.табл. 6). Цель исследования достигнута.

7. Список литературы

1. M.A.F. Azlah, L.S. Chua, F.R. Rahmad, F.I. Abdullah, S.R Wan Alwi, “Review on Techniques for Plant Leaf Classification and Recognition”, Computers 2019, *8*, 77. 22 pp, doi: 10.3390/computers8040077
2. P.K. Sethy, N.K. Barpanda, A.K. Rath, “Detection & Identification of Rice Leaf Diseases using Multiclass SVM and Particle Swarm Optimization Technique”, International Journal of Innovative Technology and Exploring Engineering (IJITEE), ISSN: 2278-3075, Volume-8 Issue-6S2, April 2019, pp 108-120.
3. 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.
4. United Nations, Department of Economic and Social Affairs, Population Division (2019). World Population Prospects 2019: Highlights (ST/ESA/SER.A/423).
5. Dataset of images. Available: https://arxiv.org/abs/1511.08060
6. G. Mathur, H. Purohit, “Performance Analysis of Color Image Segmentation using K-Means Clustering Algorithm in Different Color Spaces”, IOSR Journal of VLSI and Signal Processing (IOSR-JVSP), Vol. 4, Issue 6, Ver. III (Nov - Dec. 2014), pp 1-4. Available: www.iosrjournals.org.
7. G. Jeon, “Color Image Enhancement by Histogram Equalization in Heterogeneous Color Space", International Journal of Multimedia and Ubiquitous Engineering (IJMUE), Vol. 9, No. 7 (2014), pp 309-318, doi: http://dx.doi.org/10.14257/ijmue.2014.9.7.26
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. Conners 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. Балабанов А.С., Стронгина Н.Р. «Анализ данных в экономических приложениях». Учебное пособие. Н.Новгород: Изд-во Нижегородского госуниверситета им Н.И. Лобачевского, 2004. 135с. ISBN 5-85746-760-8
12. H. G. Dietz, “RGB+NIR Extraction from a single RAW image”, Department of Electrical and Computer Engineering Center for Visualization & Virtual Environments University of Kentucky, Lexington, KY 40506-0046, Apr, 2006. URL: http://aggregate.org/DIT/RGBNIR/

13. Максимова И. И, «Методы машинного обучения в задаче обнаружения и классификации болезней листьев томатов». Курсовая работа. Н. Новгород, 2020, 26с.

8. Приложение

Приложение А. Показатели F-Score, полученные на векторах *feaures*$_{glob}$ *и* *features*$_{loc}$

Таблица А1. Показатели F-Score, полученные на *feaures*$_{glob}$ и *features*$_{loc}$ векторах

Классификатор	Вектор данных	Набор признаков			
		STAT	STAT + HIST	GLCM	FEATURES
DTC	glob	0,38	0,47	0,56	0,56
	loc	0,50	0,51	0,54	0,60
KNN	glob	0,41	0,53	0,61	0,63
	loc	0,55	0,60	0,60	0,65
RF	glob	0,43	0,57	0,65	0,65
	loc	0,57	0,61	0,64	0,67
MSVM	glob	0,44	0,58	0,74	0,76
	loc	0,60	0,70	0,71	0,78
1 layer MLP	glob	0,45	0,60	0,78	0,79
	loc	0,59	0,72	0,77	0,84

Рисунок А2. Диаграмма показателей F-Score, полученных на feaures$_{glob}$ и features$_{loc}$ векторах

Приложение B. Показатели F-Score, полученные на векторах $feaures_{loc}$ и $features_{comb}$

Таблица B1. Показатели F-Score, полученные на $feaures_{loc}$ и $features_{comb}$ векторах

Классификатор	Вектор данных	Набор признаков			
		STAT	STAT + HIST	GLCM	FEATURES
DTC	loc	0,50	0,51	0,54	0,60
	comb	0,51	0,51	0,55	0,60
KNN	loc	0,55	0,60	0,60	0,65
	comb	0,55	0,60	0,60	0,64
RF	loc	0,57	0,61	0,64	0,67
	comb	0,57	0,61	0,63	0,68
MSVM	loc	0,60	0,70	0,71	0,78
	comb	0,60	0,70	0,72	0,79
1 layer MLP	loc	0,59	0,72	0,77	0,84
	comb	0,60	0,71	0,77	0,84

Приложение C. Показатели F-Score, полученные на векторах $features_{comb}$ и $features_{comb}$+$NDVI_{loc}$

Таблица C1. Показатели F-Score, полученные на векторах $features_{comb}$ и $features_{comb}$ + $NDVI_{loc}$

Классификатор	Вектор данных	Набор признаков				
		NDVI	STAT	STAT + HIST	GLCM	FEATURES
DTC	comb	-	0,51	0,51	0,55	0,60
	comb + NDVI	0,14	0,51	0,51	0,55	0,59
KNN	comb	-	0,55	0,60	0,60	0,64
	comb + NDVI	0,15	0,56	0,62	0,58	0,63
RF	comb	-	0,57	0,61	0,63	0,68
	comb + NDVI	0,16	0,57	0,64	0,64	0,68
MSVM	comb	-	0,60	0,70	0,72	0,79
	comb + NDVI	0,15	0,59	0,70	0,73	0,79
1 layer MLP	comb	-	0,60	0,71	0,77	0,84
	comb + NDVI	0,16	0,60	0,72	0,78	0,80

Приложение D. Показатели общностей $features_{loc}$ и $features_{glob}$ признаков

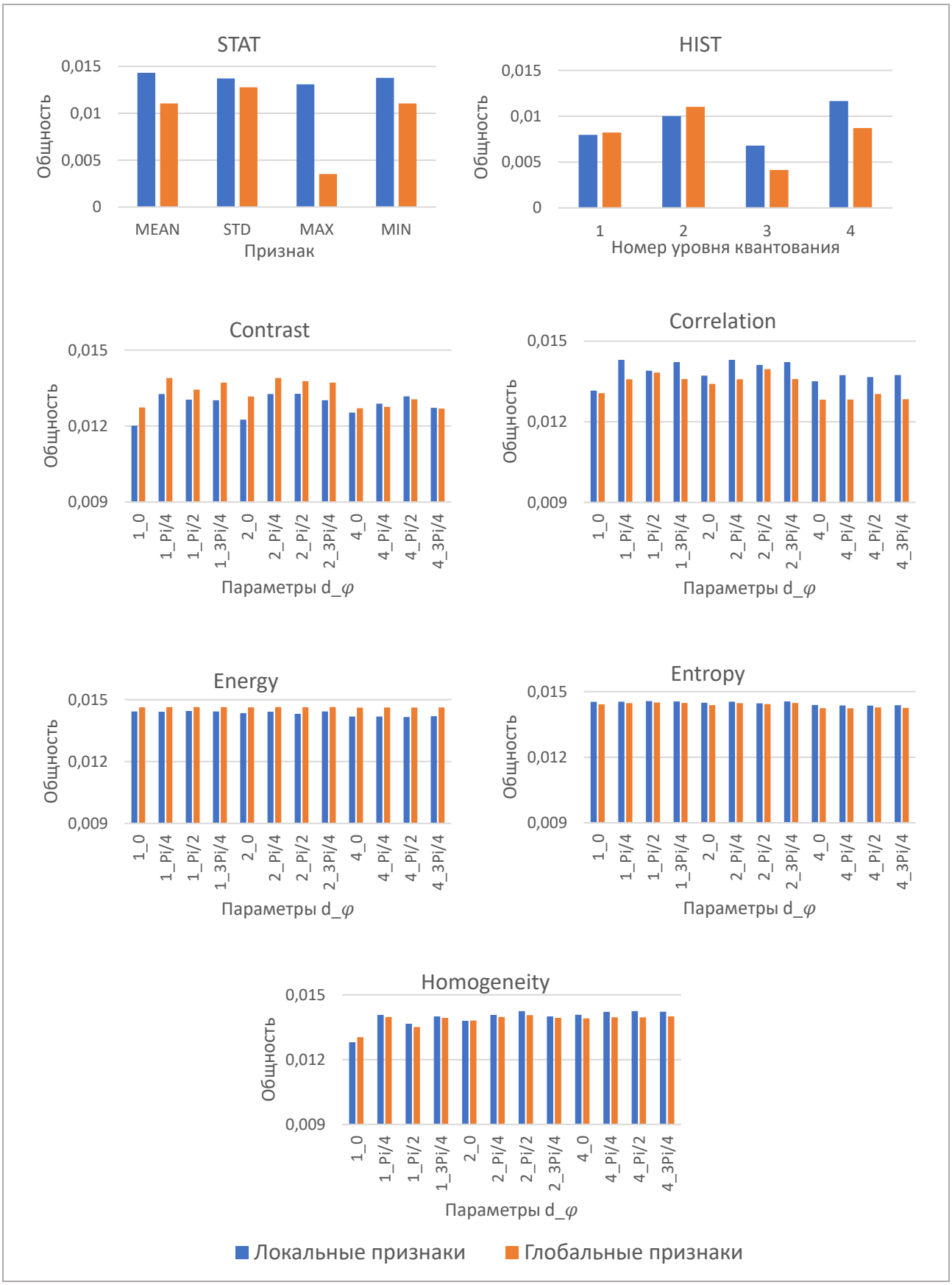

Рисунок D1. Показатели общностей для каждого из $features_{loc}$ и $features_{glob}$ признаков при m=3

Приложение Е. Матрица корреляции признаков $features_{comb}$ + $NDVI_{loc}$

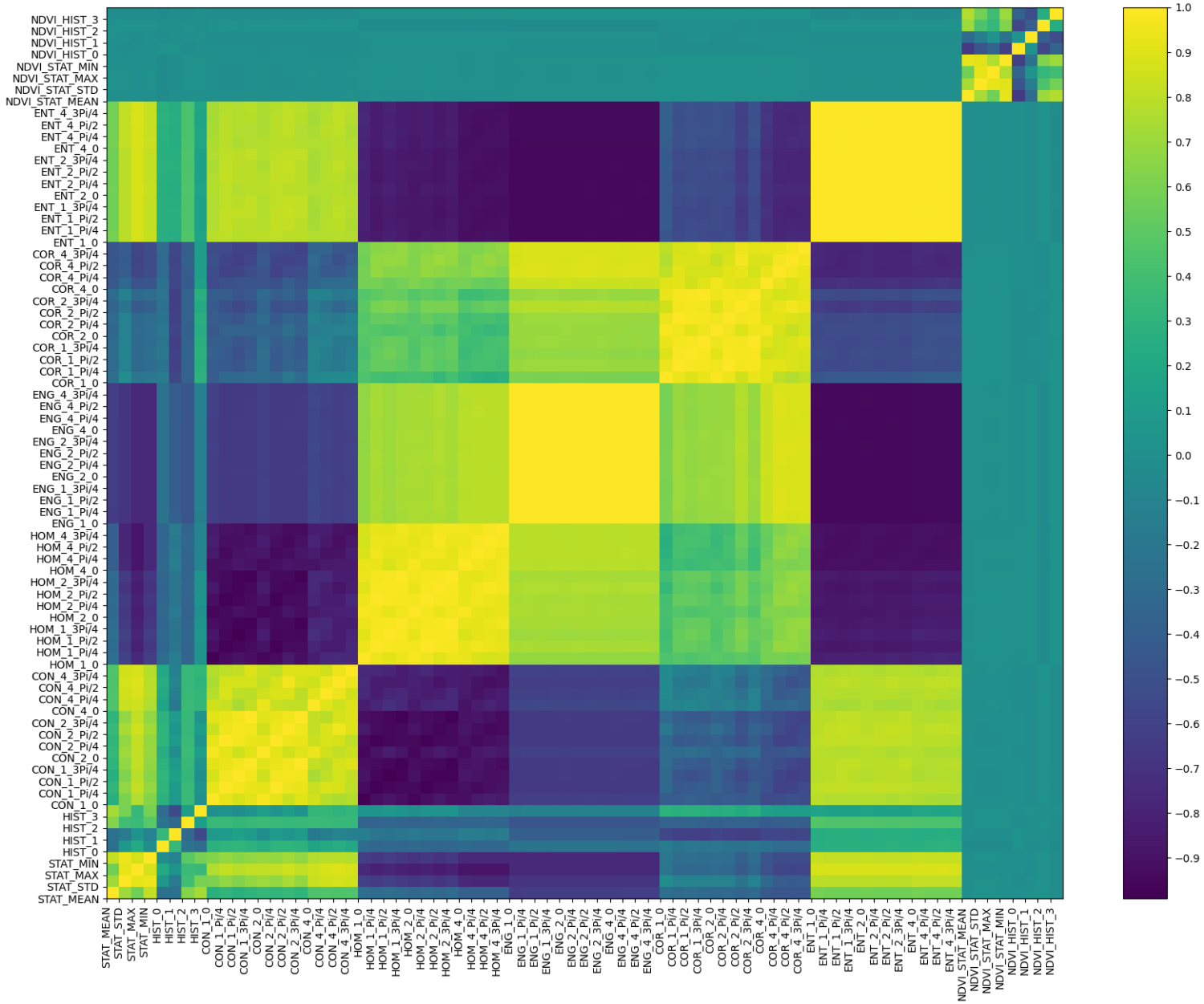

Рисунок Е1. Корреляционная матрица признаков $features_{comb}$ + $NDVI_{loc}$

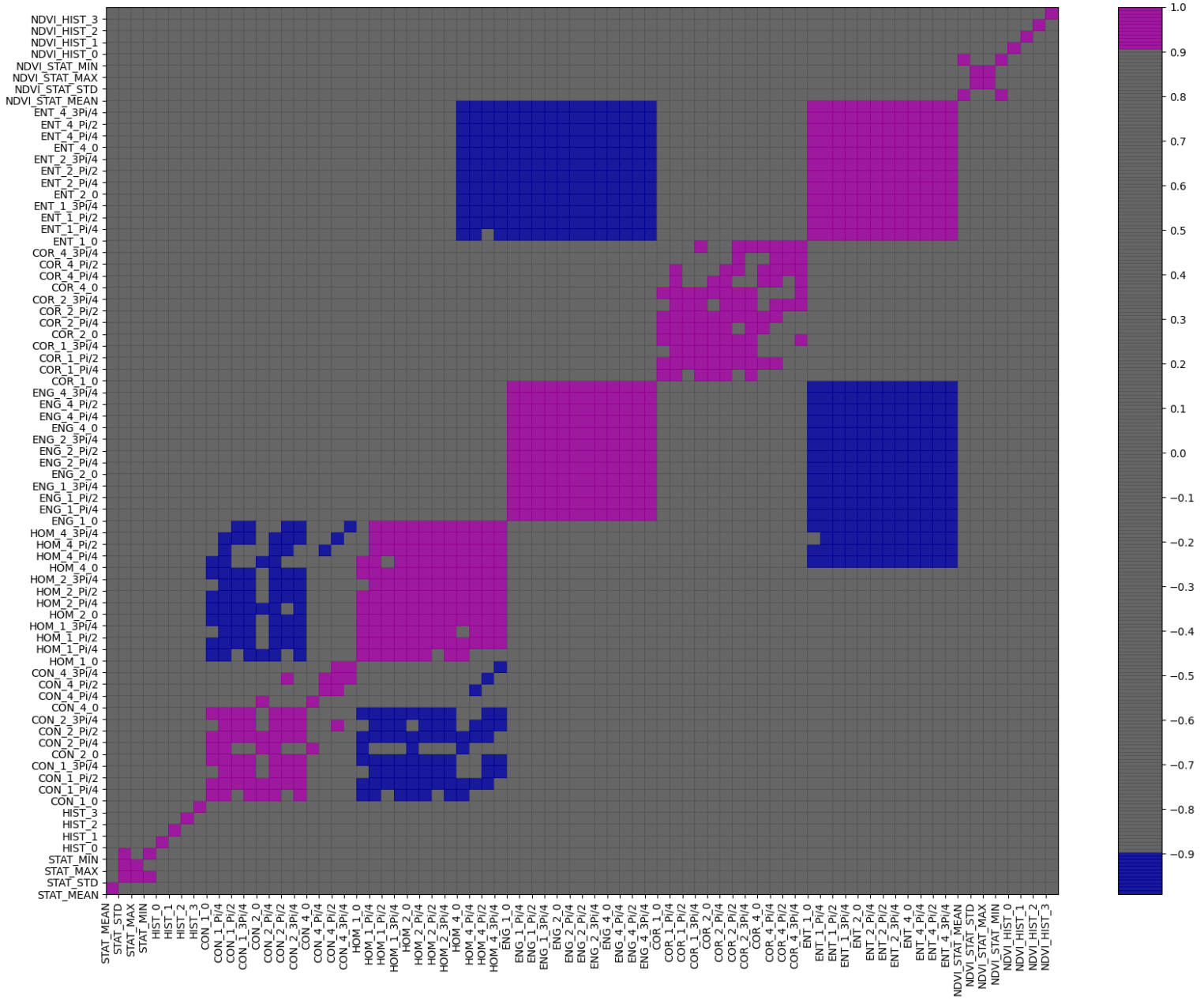

Рисунок Е2. Выделение признаков $features_{comb}$ + $NDVI_{loc}$ с показателем корреляции > 0.9

Приложение F. Показатели F-Score, полученные на векторе $features_{comb}$+ $NDVI_{loc}$ и после удаления из него сильнокоррелированных признаков

Таблица F1. Показатели F-Score, полученные на векторе $features_{comb}$+$NDVI_{loc}$ и после удаления из него сильнокоррелированных признаков

Классификатор	Вектор признаков	Набор признаков			
		NDVI	STAT	GLCM	FEATURES
DTC	comb + NDVI	0,14	0,51	0,55	0,59
	[comb + NDVI]*	0,16	0,41	0,54	0,60
KNN	comb + NDVI	0,15	0,56	0,58	0,63
	[comb + NDVI]*	0,15	0,44	0,58	0,63
RF	comb + NDVI	0,16	0,57	0,64	0,68
	[comb + NDVI]*	0,15	0,45	0,64	0,69
MSVM	comb + NDVI	0,15	0,59	0,73	0,79
	[comb + NDVI]*	0,16	0,46	0,72	0,79
1 layer MLP	comb + NDVI	0,16	0,60	0,78	0,80
	[comb + NDVI]*	0,16	0,47	0,78	0,81

Приложение G. Показатели общности признаков вектора ($features_{comb}$ + $NDVI_{loc}$)’

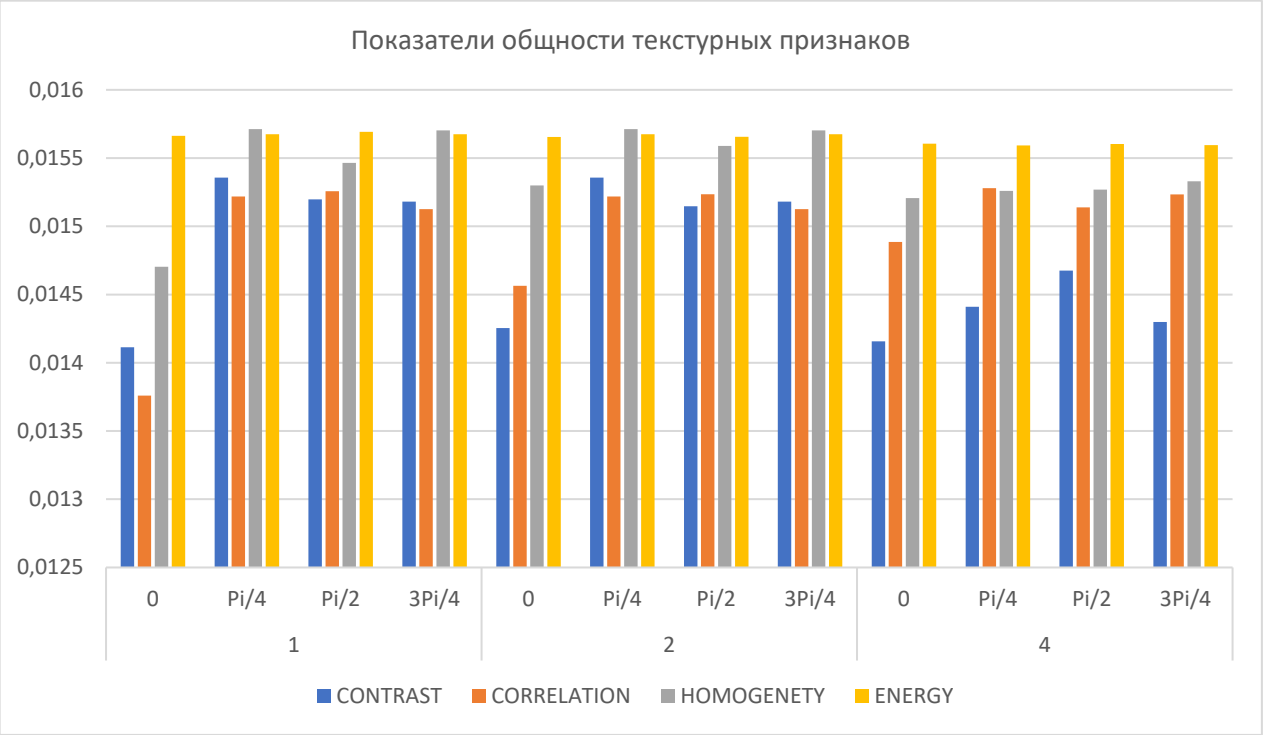

Рисунок G1. Показатели общности текстурных признаков при m=3

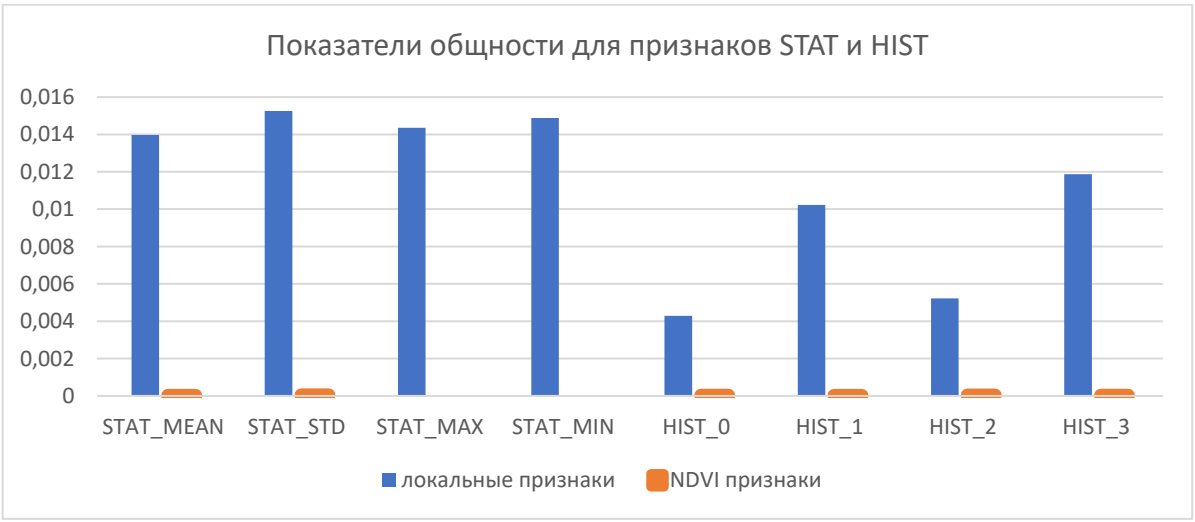

Рисунок G2. Показатели общности признаков STAT и HIST при m=3

Приложение H. Показатели F-Score, полученные на векторах $features_{opt}$ и $features_{comb.}$

Таблица H1. Показатели F-Score, полученные на векторах $features_{opt}$ и $features_{comb}$

Классификатор	Вектор признаков	Набор признаков			
		STAT	STAT + HIST	GLCM	FEATURES
DTC	opt	0,50	0,51	0,49	0,57
	comb	0,51	0,51	0,55	0,60
KNN	opt	0,55	0,59	0,54	0,65
	comb	0,55	0,60	0,60	0,64
RF	opt	0,57	0,61	0,56	0,65
	comb	0,57	0,61	0,63	0,68
MSVM	opt	0,59	0,68	0,60	0,76
	comb	0,60	0,70	0,72	0,79
1 layer MLP	opt	0,59	0,70	0,63	0,78
	comb	0,60	0,71	0,77	0,84

Приложение I. Показатели F-Score, полученные на $features_{opt}$ для каждого класса.

Классификатор	Early blight	Septoria leaf spot	Late blight	Bacterial spot	Yellow Leaf Curl Virus	Healthy	Среднее F-Score
DTC	0,41	0,43	0,50	0,68	0,68	0,74	0,57
KNN	0,47	0,55	0,53	0,75	0,77	0,84	0,65
RF	0,47	0,50	0,55	0,75	0,79	0,85	0,65
MSVM	0,62	0,67	0,65	0,79	0,91	0,90	0,76
1 Layer MLP	0,64	0,70	0,68	0,80	0,91	0,92	0,78