

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
КАФЕДРА ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

**Шамсутдинова Лилия Рашидовна**

**Магистерская диссертация**

**Автоматическое распознавание стилевой  
принадлежности картин**

Направление 02.04.02

Фундаментальная информатика и информационные технологии  
Магистерская программа Технологии Баз Данных, 15.М09-пу

Научный руководитель,  
кандидат физ.-мат. наук,  
доцент  
Должиков В. В.

Санкт-Петербург

2017

# Содержание

Введение.....	4
Постановка задачи.....	7
Обзор литературы.....	8
Глава 1. Методология .....	10
1.1 Стили живописи .....	10
1.2 Визуальные характеристики .....	11
1.2.1 Classemes .....	12
1.2.2 PiCoDes.....	13
1.2.3 HOG .....	13
1.2.4 GIST .....	15
1.2.5 SSIM.....	15
1.2.6 SIFT.....	16
1.3 Определение понятия «стиль».....	17
Глава 2. Программная реализация.....	18
2.1 Составление коллекции картин .....	19
2.2 Выделение визуальных характеристик.....	20
2.3 Обработка векторов характеристик .....	22
2.4 Классификация и нейронные сети.....	23
Глава 3. Эксперименты.....	25
3.1 Сравнение с другими известными решениями .....	33
Глава 4. Прототип приложения .....	35
Выводы .....	38

Заключение .....	40
Список использованной литературы.....	41

## Введение

В течение последних нескольких лет число оцифрованных картин, которые находятся в свободном доступе, увеличивается все более быстрыми темпами. В связи с этим возрастает и потребность в разработке систем для хранения и обработки данной информации. Оцифрованные коллекции содержат классические (Wikiart, Arkyves, BBC Yourpainting) и современные (Artsy, Behance, Artnet) произведения искусства. Обычно данные коллекции, особенно самые первые, содержат метадату в форме аннотаций о художнике, стиле, дате и т.д. Для онлайн галерей, выставляющих современную живопись, высока потребность в разработке автоматической рекомендательной системы, способной выделить «похожие» картины, которые пользователь с большой вероятностью захочет приобрести [1]. Данный факт подчеркивает необходимость исследования различных метрик «похожести» картин.

Область компьютерного зрения находит свое применение в распознавании и категоризации объектов и сцен на картинах и видео [1]. Однако когда человек смотрит на картину, он не только распознает объекты, но также выделяет различные культурные и исторические аспекты [2]. Эксперт или даже непрофессионал может сделать предположение о стиле и жанре картины, художнике, написавшем ее. Возможность понимания сложной визуальной концепции является впечатляющей особенностью человеческого восприятия [3].

Художники используют различные подходы к описанию картины. Например, стилистические элементы, такие как пространство, текстура, форма, цвет, линии. Другие принципы включают в себя движение, единство, гармонию, баланс, контраст, пропорции. К этому могут быть добавлены физические атрибуты, например, мазки [4].

Для задачи анализа произведений искусства исследователи разработали различные визуальные характеристики, которые кодируют некоторые принципы. Цвет и мазки кодируются низкоуровневыми характеристиками, такими как статистика текстуры и цветовые гистограммы ([4], [5]). Цвет и текстура склонны к изменениям во время оцифровки картины [6]. Это подчеркивает необходимость в разработке визуальных характеристик, подходящих для анализа картин.

Очевидно, что разработка визуальных характеристик, кодирующих каждый художественный принцип, является сложнейшей задачей. Последние достижения в области компьютерного зрения показали преимущества «обучения» характеристик на данных по сравнению с их разработкой [1].

Принимая во внимание вышеупомянутые проблемы в разработке и обучении визуальных характеристик, была поставлена задача исследования различных современных подходов к решению данной проблемы. Затем, используя нейронные сети и методы классификации, определяется стиль картины.

Таким образом, цель данной работы – исследование возможностей автоматического определения стиля картины.

Актуальность работы заключается в следующем:

1. Автоматическая классификация картин способствует организации и структурированию больших коллекций.
2. Нет возможности дать математическое определение понятию «стиль» картины.
3. Необходимость разработки рекомендательной системы.
4. Возможность замены «живого» эксперта, что увеличивает скорость классификации картин.

5. Возможность получения новой информации о неизвестной картине, используя уже существующую базу картин.
6. В недавних разработках точность определения стиля около 60% [1].
7. В связи с повсеместным использованием смартфонов, появилась потребность в разработке приложения, способного автоматически идентифицировать картину для туризма и музеев [2].
8. Отсутствие приложений, решающих поставленную задачу.
9. В России нет исследований, посвященных данной тематике.

Одной из сложностей поставленной задачи является отсутствие публичной базы картин. В связи с этим, важным этапом является поиск источника оцифрованных произведений искусства.

## Постановка задачи

Данная работа сосредоточена на исследовании методов автоматического распознавания стилевой принадлежности картин. Вопросы, которые возникают после постановки данной задачи включают в себя, но не ограничиваются следующими: Какие визуальные характеристики должны быть использованы для кодирования информации о картине? Как добиться корректной меры схожести?

Для решения задачи был составлен следующий план работы:

1. Сбор данных.
2. Изучение собранных данных:
  - 2.1. Выявление наиболее значимых визуальных характеристик.
  - 2.2. Структуризация полученных данных.
3. Извлечение низкоуровневых и высокоуровневых характеристик.
4. Сокращение пространства характеристик, посредством выделения наиболее значимых.
5. Конкатенация различных характеристик.
6. Имплементация и тестирование нейронной сети.
7. Имплементация и тестирование различных методов классификации.
8. Сравнение полученных результатов.
9. Разработка веб-приложения для автоматического определения стиля картины.
10. Исследование возможностей дальнейшего улучшения разработанной системы.

## Обзор литературы

В области живописи компьютеры используются для широкого круга задач. Традиционно, методы обработки изображений использовались искусствоведами для пигментационного анализа, статистической оценки мазков и т.д. Данные подходы описаны в работах [7, 8].

Несколько исследований посвящены вопросу кодирования информации о картинах. Большинство из них используют низкоуровневые характеристики для кодирования цвета, теней, текстур, краев и т.д. Ломбарди в [9] представил исследование эффективности данных характеристик для задачи классификации картин по авторам. Он использовал небольшую выборку различных авторов и применял алгоритмы обучения с учителем и без.

Так как мазки являются некой подписью, способной помочь в идентификации автора, создание визуальных характеристик, кодирующих мазки, широко распространено. Следует отметить работы [5, 10, 11, 12, 13, 14]. Как правило, для этой цели используются статистические данные. Однако характеристики текстуры зависят от разрешения изображения. Исследователи также используют характеристики, основанные на гистограмме локальных краев, например SIFT [15] и HOG [16]. [17] использует SIFT характеристики в подходе Bag-of-words («мешок слов») для определения художника среди 8 возможных.

Анора и другие в работе [2] представили сравнительное исследование, которое оценивало низкоуровневые характеристики, такие как SIFT и Color SIFT [18] и семантические характеристики, такие как Classemes [19], которые кодируют присутствие объекта на изображении. Было определено, что семантические характеристики намного превосходят низкоуровневые в задаче классификации картин. Однако, оценка была проведена на небольшом объеме данных: всего около 500 картин. Карниерио и другие в работе [20] также пришли

к выводу, что низкоуровневые характеристики не эффективны из-за непоследовательности цветовых и текстурных шаблонов.

Одним из самых недавних исследований является работа Салеха и других [21], в которой описывается использование различных подходов к определению схожести художников, основываясь на их произведениях.

Как можно заметить, большинство работ в области классификации картин исследуют возможности определения художника. В работе [1] показано, что определение стиля является более сложной задачей, чем определение художника или жанра. Также в работах не рассматривается возможность сочетания низкоуровневых и высокоуровневых характеристик. Кроме того, отсутствуют программные продукты, реализующие поставленную задачу. Таким образом, обзор известных решений задачи показывает, что данная область актуальна и требует исследования.

# Глава 1. Методология

В данной главе будет описана методология выбора наиболее подходящих визуальных характеристик и наиболее точных метрик похожести картин. Полученные метрики используются для имитации способности эксперта определять стиль произведения искусства. Первым этапом является извлечение числовых визуальных характеристик из изображения. Данные визуальные характеристики ранжируются от низкоуровневых (например, края) до высокоуровневых (например, объекты на картине). Что еще более важно, на следующем шаге происходит настройка характеристик для задачи определения стиля, посредством обучения соответствующей метрики. Учитывая полученную метрику, картина проецируется в пространство с гораздо меньшей размерностью. Затем происходит обучение классификатора и проектирование нейронной сети.

## 1.1 Стили живописи

Стиль — 1) общность идейно-художественных особенностей произведений искусства определенной эпохи. Возникновение и смена стилей определяется ходом исторического развития общества (например, классицизм, барокко и др.); 2) национальная особенность искусства (китайский, мавританский стиль и т. д.). Говорят также о стиле группы художников или одного художника, если их творчество отличается яркими индивидуальными чертами [22].

Существует огромное разнообразие стилей и направлений в изобразительном искусстве. Часто они не имеют четко выраженных границ и плавно переходят из одного в другой, находясь в непрерывном развитии, смешении и противодействии. В рамках одного исторического художественного стиля всегда зарождается новый, а тот, в свою очередь, переходит в следующий

и т.д. Многие стили сосуществуют одновременно и поэтому «чистых стилей» вообще не бывает [23].

В таблице 1 представлены стили живописи, с которыми в дальнейшем будет происходить работа.

Абстракционизм
Поп-арт
Фигуратив
Гиперреализм
Уличное искусство

Таблица 1. Список стилей живописи в коллекции произведений искусства.

## 1.2 Визуальные характеристики

Визуальные характеристики либо разработаны и извлечены путем обучения без учителя (например, HOG, GIST), либо путем оптимизации конкретной задачи, как правило, категоризации объектов и сцен (например, характеристики, основанные на сверточной нейронной сети). Это приводит к вектору признаков высокой размерности, который не обязательно соответствует именуемым (семантическим) характеристикам изображения. Визуальные признаки можно разделить на низкоуровневые и высокоуровневые. Низкоуровневые характеристики – это визуальные дескрипторы, в которых для каждого измерения нет явного смысла. Тогда как высокоуровневые характеристики предназначены для описания некоторых понятий (как правило, объектов). В данной работе были исследованы самые передовые представители каждой категории.

Для представления низкоуровневых характеристик извлекаются GIST характеристики [24], которые представляют собой функции для категоризации

сцен. GIST характеристики – это вектор вещественных чисел размерности 512, который неявно описывает пространственную структуру изображения [25].

Для задачи семантического представления изображения извлекаются Classemes [19], PiCodes [26] и CNN-based характеристики [27]. Каждый элемент вектора представляет уверенность в присутствии объекта на изображении. Таким образом, характеристики представляют собой семантическое кодирование изображения. Classeme и PiCodes предназначены для описания множества объектов базового уровня (например, рамки и пересечение линий) [28].

Сверточная нейронная сеть (CNN) показывает огромную производительность в задаче категоризации изображений [27]. CNN имеет 4 сверточных слоя, а за ними 3 полностью связанных слоя. Комбинация выходов позволяет достичь превосходной производительности в задаче автоматической классификации картин по стилю [29].

### 1.2.1 Classemes

Classemes – это характеристики, основанные на классификаторах, которые впервые были представлены в [12]. В данном подходе базисные классы определяются непосредственно как объекты реального мира на тренировочном множестве  $D^s$ . Каждый базисный классификатор  $h_c$  представляет собой один-против-всех (One-vs-the-rest) бинарный классификатор, обученный распознавать  $c$ -ю категорию. Classemes используют  $LP - \beta$  модель классификации. Данная модель определяется как линейная комбинация  $M$  нелинейных SVM, каждый из которых обучен на векторе низкоуровневых характеристик  $f_m(x) \in R^{d_m}$ . Формально, каждый базисный классификатор  $h_c(x)$  определяется по формуле (1).

$$h_c(x) = \tau_c \left( \sum_{m=1}^M \beta_{m,c} [w_{m,c}^T \Psi_m(f_m(x)) + b_{m,c}] \right) \quad (1)$$

Явное отображение  $\Psi_m$  для характеристики  $m$  определяется как функция в формуле (2), где  $r \in Z^+$ .

$$\Psi_m : R^{d_m} \rightarrow R^{d_m(2r+1)} \quad (2)$$

Таким образом, что справедливо условие формулы (3).

$$\langle \Psi_m(f_m(x)), \Psi_m(f_m(x')) \rangle \approx K_m(f_m(x), f_m(x')) \quad (3)$$

Где  $K_m()$  - нелинейное ядерное расстояние.  $\tau_c$  - функция для квантования или масштабирования выхода классификатора. Согласно схеме  $LP - \beta$ , сначала находятся параметры  $\{w_{m,c}, b_{m,c}\}$  для каждой характеристики  $m$  независимо, путем обучения гипотезы, представленной в формуле (4), используя SVM на тренировочном множестве  $D_c^S$ .

$$h_{m,c}(x) = [w_{m,c}^T \Psi_m(f_m(x)) + b_{m,c}] \quad (4)$$

Вторым шагом, находится параметр  $\beta_c = [\beta_{1,c}, \dots, \beta_{m,c}]^T$ , такой что  $\sum_m \beta_{m,c} = 1$  [19].

### 1.2.2 PiCoDes

Для обучения PiCoDes базисных классификаторов используется бинарное кодирование меток тренировочного множества  $D^S$  : для каждого примера  $i$ , определяется  $y_i \in \{-1, +1\}^K$  – вектор, описывающий соответствующую метку категории, с  $y_{ik} = +1$ , если  $i$ -ый пример принадлежит классу  $k$ . Такое кодирование позволяет использовать стратегию один-против-всех (One-vs-the-rest) для обучения базисных классификаторов [26].

### 1.2.3 HOG

В основе метода HOG (Histogram of Oriented Gradients) лежит предположение, что вид распределения градиентов интенсивности изображения позволяет достаточно точно определить наличие и форму присутствующих на нем объектов [30].

При описании фрагмента изображения оно разбивается на несколько небольших участков, далее называемых ячейками. В ячейках вычисляются гистограммы  $h_i$  направленных градиентов внутренних точек. Обычно они объединяются в одну гистограмму  $h = f(h_1, \dots, h_k)$ , после чего она нормализуется по яркости ( $L_2$  или  $L_1$  норма) по формуле (5), где  $\varepsilon$  — некоторая малая константа.

$$h_{L_2} = \frac{h}{\sqrt{|h|_2^2 + \varepsilon}}, \quad h_{L_1} = \frac{h}{|h|_1 + \varepsilon}, \quad h_{\sqrt{L_1}} = \sqrt{h_{L_1}} \quad (5)$$

Таким образом, данный описатель содержит пространственную информацию о фрагменте и инвариантен к освещению.

При вычислении градиентов производится свертка изображения с ядрами  $[-1, 0, 1]$  и  $[-1, 0, 1]^T$ , в результате чего образуются две матрицы  $D_x$  и  $D_y$  производных вдоль осей  $x$  и  $y$  соответственно. Эти матрицы используются для вычисления углов и величин (модулей) градиентов в каждой точке изображения.

Пусть множество углов  $(-\pi, \pi]$  разбивается на  $n$  равных интервалов вида  $(-\frac{k-1}{n}\pi, \frac{k}{n}\pi]$ , где  $k = \{1, \dots, n\}$ . Каждому интервалу ставится в соответствие бин гистограммы. Тогда гистограмма ячейки заполняется так, что величина градиента в каждой внутренней его точке добавляется к величине бина, соответствующего интервалу, содержащему угол данного градиента. Пример ячейки изображен на рисунке 1.

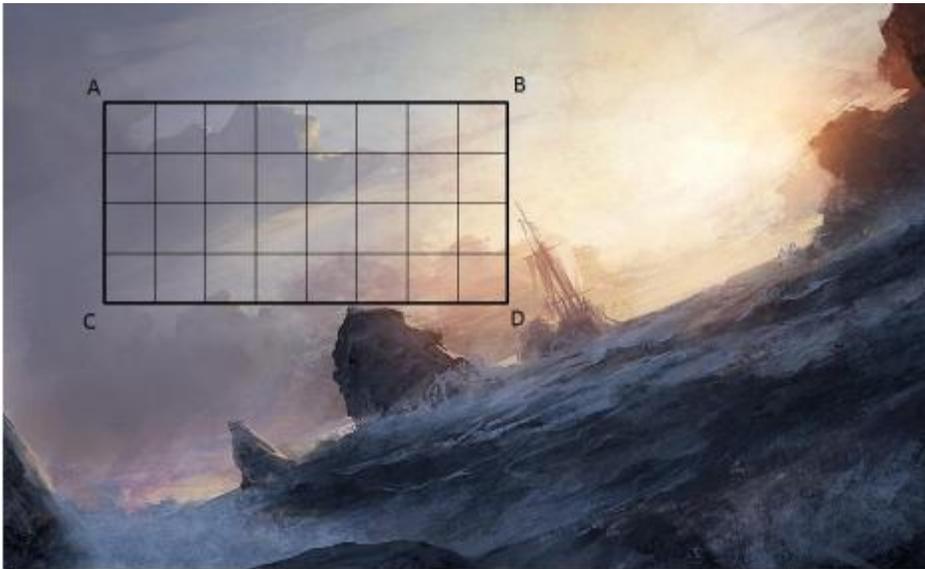


Рис. 1. Скользящее окно

### 1.2.4 GIST

GIST суммирует градиентную информацию (масштабы и ориентацию) для разных частей изображения. Вычисления происходят следующим образом:

1. Свертка изображения, используя 32 фильтра Габора в 4 масштабах, 8 ориентациях. В результате получаются 32 карты изображения.
2. Разделение каждой карты на 16 областей и усреднение значений в каждой области.
3. Конкатенация всех усредненных значений [31].

### 1.2.5 SSIM

SSIM (индекс структурного сходства) - метод измерения схожести между двумя изображениями путем полного сопоставления. Данный метод учитывает «восприятие ошибки» благодаря учёту структурного изменения информации. Идея заключается в том, что пиксели имеют сильную взаимосвязь, особенно когда они близки пространственно. Данные зависимости несут важную информацию о структуре объектов и о сцене в целом.

SSIM-метрика рассчитана на различные размеры окна. Разница между двумя окнами  $x$  и  $y$  имеющими одинаковый размер  $N \times N$  представлена в формуле (6).

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (6)$$

Где  $\mu_x$  - среднее  $x$ ,  $\mu_y$  - среднее  $y$ ,  $\sigma_x^2$  - дисперсия  $x$ ,  $\sigma_y^2$  - дисперсия  $y$ ,  $\sigma_{xy}$  - ковариация  $x$  и  $y$ ,  $c_1 = (0.01L)^2$ ,  $c_2 = (0.03L)^2$ ,  $L$  - динамический диапазон пикселей.

Приведённая формула применима только для яркости изображения, по которой и происходит оценка качества. Полученный SSIM-индекс лежит в пределах от  $-1$  до  $+1$ . Значение  $+1$  достигается только при полной аутентичности образцов [32].

### 1.2.6 SIFT

Для формирования дескриптора SIFT (Scale Invariant Feature Transform) сначала вычисляются значения магнитуды и ориентации градиента в каждом пикселе, принадлежащем окрестности особой точки размером  $16 \times 16$  пикселей. Магнитуды градиентов при этом учитываются с весами, пропорциональными значению функции плотности нормального распределения с математическим ожиданием в рассматриваемой особой точке и стандартным отклонением, равным половине ширины окрестности.

В каждом квадрате размером  $4 \times 4$  пикселя вычисляется гистограмма ориентированных градиентов путем добавления взвешенного значения магнитуды градиента к одному из 8 бинов гистограммы. Чтобы уменьшить различные «граничные» эффекты, связанные с отнесением похожих градиентов к разным квадратам (что может возникнуть вследствие небольшого сдвига расположения особой точки) используется билинейная интерполяция: значение магнитуды каждого градиента добавляется не только в гистограмму,

соответствующую квадрату, к которому данный пиксель относится, но и к гистограммам, соответствующим соседним квадратам. При этом значение магнитуды добавляется с весом, пропорциональным расстоянию от пикселя, в котором вычислен данный градиент, до центра соответствующего квадрата. Все вычисленные гистограммы объединяются в один вектор, размером, равным  $128 = 8$  (число бинов)  $\times 4 \times 4$  (число квадратов) [33].

### **1.3 Определение понятия «стиль»**

Одной из задач данной работы является исследование возможности описать, что есть стиль картины. Формальный подход к стилю предполагает, что искусство понимается в терминах: линия, цвет, форма и т.д., а не содержание само по себе. По двум причинам формальный подход к стилю является хорошей отправной точкой для автоматической классификации. Во-первых, формальные элементы картины, такие как линии и цвет, легко определяются компьютером. Во-вторых, многие стили, например абстракционизм, не содержат различимого содержания. Таким образом, стиль формально можно определить как определенное сочетание линий, цветов, форм и других формальных элементов картины. А различные формальные элементы описываются путем извлечения визуальных характеристик картины. Следовательно, «стиль» - это определенное сочетание визуальных характеристик.

## Глава 2. Программная реализация

Для получения коллекции произведений искусства был использован сайт “Artsy” [34], содержащий современные полотна. Большинство работ в области классификации произведений искусства работают с классическими произведениями. Данное исследование фокусируется на современных направлениях живописи, чем и обоснован выбор источника данных. Коллекция содержит полотна 5 различных стилей (абстракционизм, поп-арт, фигуратив, гиперреализм, уличное искусство). В таблице 2 приведено количество картин каждого стиля в коллекции.

Стиль	Количество картин
Абстракционизм	35 189
Поп-арт	6 543
Фигуратив	25 985
Гиперреализм	5 310
Уличное искусство	2 346

Таблица 2. Количество картин каждого стиля в коллекции.

Задачей является автоматическая классификация картин по стилям, используя визуальные характеристики. Характеристики автоматически извлекаются с помощью алгоритмов компьютерного зрения. Следует отметить, что задача распознавания стиля картин является одной из самых сложных. Существуют большие различия в плане визуального представления среди картин одного стиля. Данные различия гораздо более ограничены среди картин одного художника или жанра [1].

Алгоритм работы по определению стиля картины представлен на рисунке 2.



Рис. 2. Алгоритм определения стиля картины.

## 2.1 Составление коллекции картин

Для сбора коллекции картин был реализован автоматизированный алгоритм обхода сайта Artsy. На первом этапе происходил сбор ссылок на картины определенного стиля, а затем каждая картина загружалась в локальную директорию. Фрагмент кода по загрузке картины представлен на рисунке 3.

```

import os
import re
import sys
import lxml.html
from lxml.cssselect import CSSSelector
import requests
import wget

link = CSSSelector(".js-artwork-
images__images__image__display__img")
linkRegex = re.compile('img data-src="(.*?)"',
re.UNICODE)

# Загрузка страницы

```

```

def download_webpage(url):
    r = requests.get(url)
    if r.status_code != 200:
        raise Exception
    return r

# Парсинг страницы
def parsePage(requests_object):
    return lxml.html.fromstring(requests_object.text)

# Извлечение ссылки на изображение
def extractLink(tree):
    link_text =
lxml.html.tostring(link(tree)[0]).decode("utf-8")
    return linkRegex.search(link_text).group(1)

# Загрузка изображения
def downloadImage(link, pathToFile):
    filename = wget.download(link, pathToFile)
    return filename

```

Рис. 3. Фрагмент кода загрузки картины на языке Python.

## 2.2 Выделение визуальных характеристик

В качестве низкоуровневых характеристик были выбраны: GIST, PHOG, SSIM, SIFT. Алгоритмы извлечения каждого типа характеристик реализованы на языке программирования Python. Пример работы алгоритма извлечения гистограммы ориентированных градиентов представлен на рисунке 4.

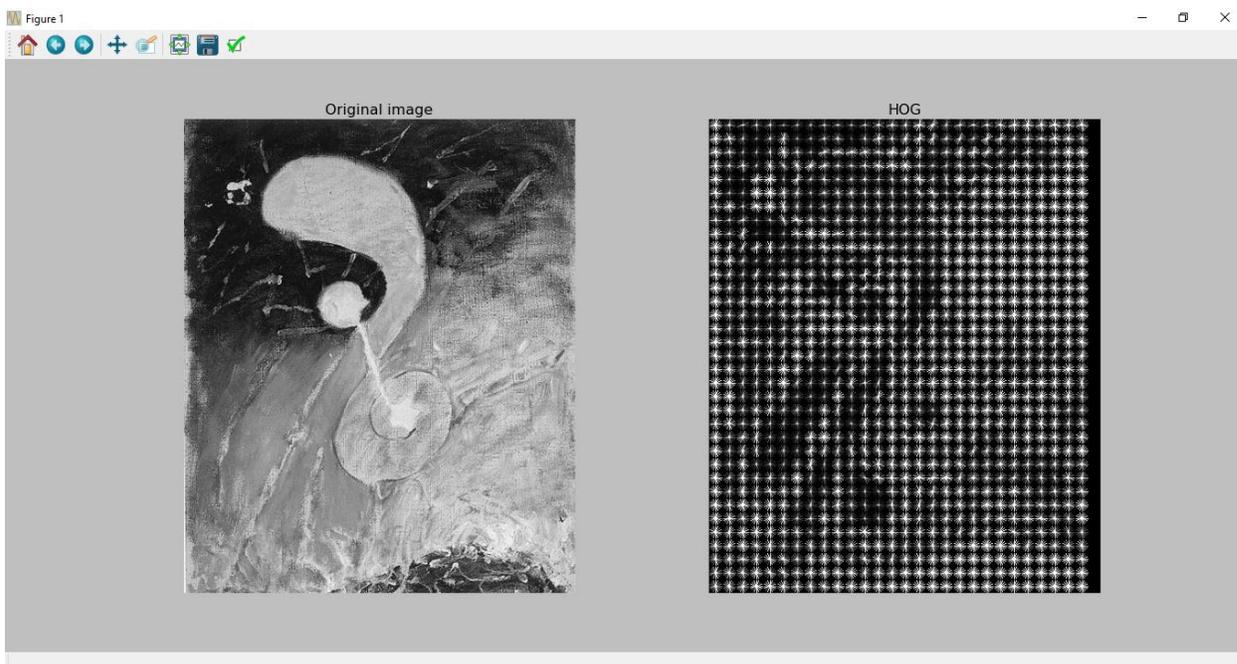


Рис. 4. Гистограмма ориентированных градиентов.

В качестве гистограммы ориентированных градиентов был выбран описатель PHOG, который отличается от стандартного тем, что состоит из нескольких независимых этапов скольжения с разным размером скользящего окна.

Высокоуровневые характеристики (Classemes, PiCoDes) были извлечены с использованием vlg extractor [35].

Таким образом, каждая картина представляется в виде вектора числовых характеристик. Размерности векторов представлены в таблице 3.

<b>Тип визуальных характеристик</b>	<b>Размерность вектора</b>
GIST	960
PHOG	3400
SSIM	6300
SIFT	10500
Classemes	2660
PiCoDes	2048

Таблица 3. Размерность векторов визуальных характеристик.

## 2.3 Обработка векторов характеристик

Как указано в таблице 23, размерности векторов варьируются от 960 до 10500. Для большей корректности сравнения каждый вектор был приведен к размеру 512, основываясь на критерии Фишера. Данный подход представлен на рисунке 5.

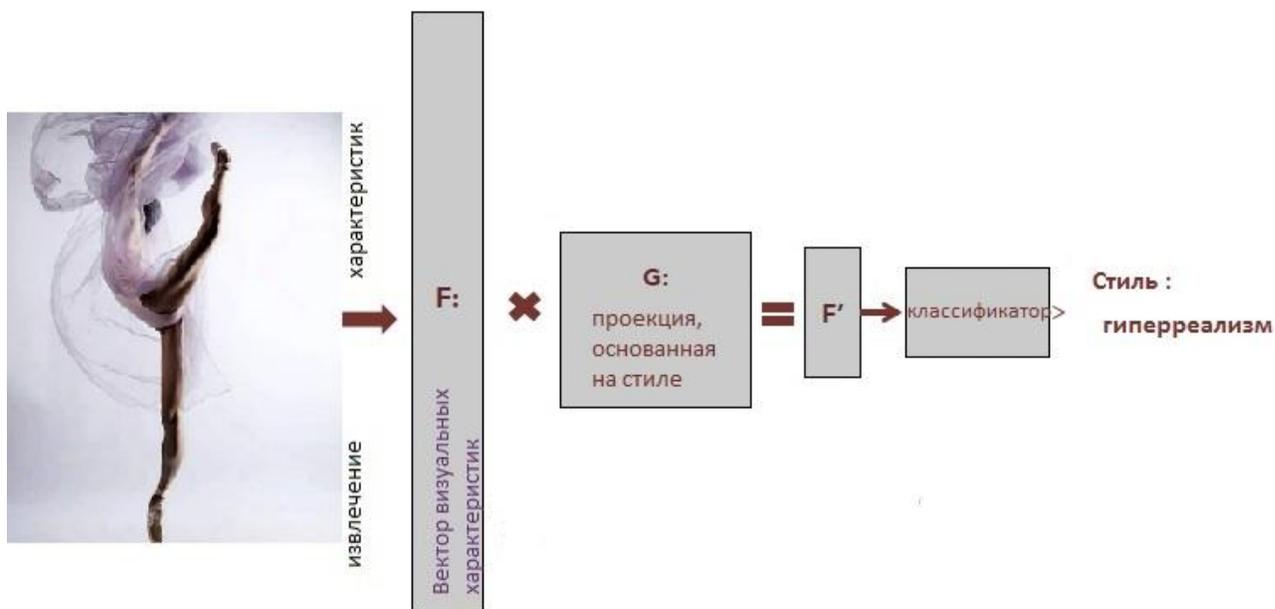


Рис. 5. Алгоритм работы с визуальными характеристиками. Извлечение, а затем приведение к вектору меньшей размерности.

Если в качестве входного вектора классификатора использовалось сочетание характеристик, то вектор, полученный путем конкатенации выбранных характеристик, приводился к размерности 512. Данный подход представлен на рисунке 6.

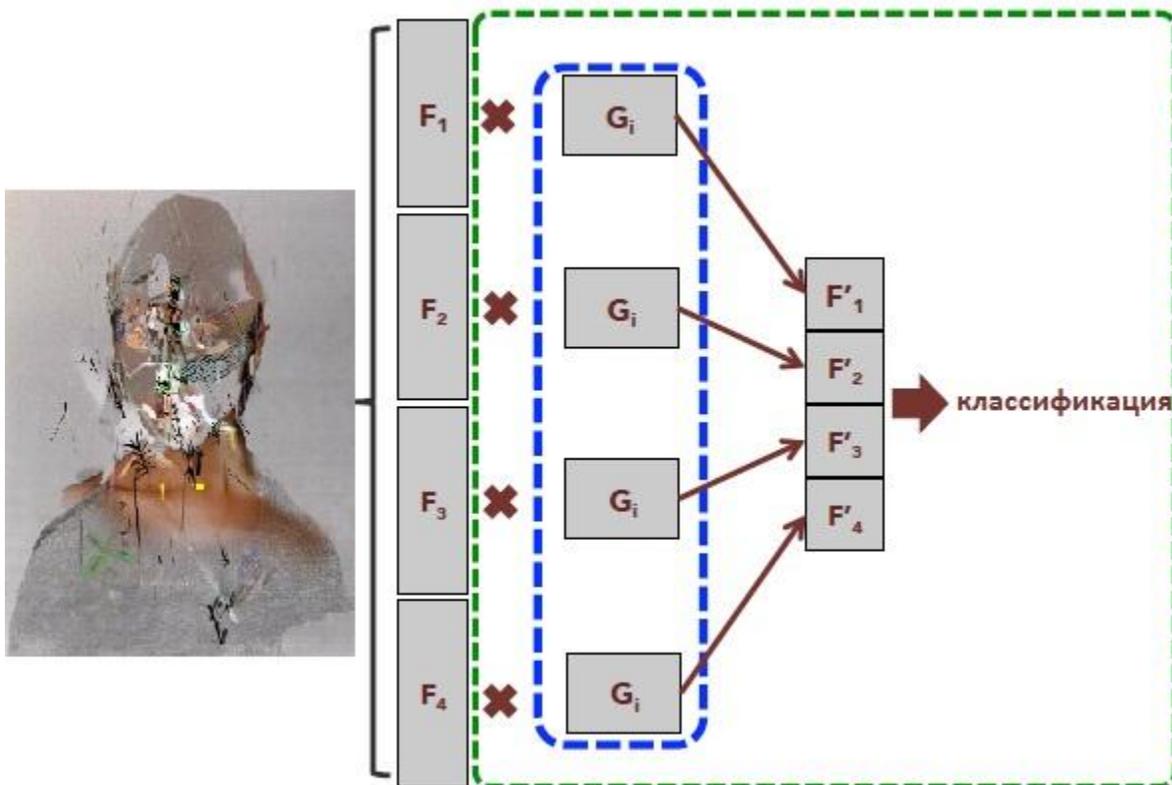


Рис. 6. Алгоритм работы с сочетанием визуальных характеристик. Конкатенация выбранных векторов визуальных характеристик, а затем приведение к вектору меньшей размерности.

## 2.4 Классификация и нейронные сети

В качестве подходов к классификации картин по стилю были выбраны следующие алгоритмы: Naïve Bayes, SVM, CNN, Random Forest, Logistic Regression. Достоверно известно, что не существует заранее определенного оптимального метода, и, в зависимости от задачи, различные методы работают лучше или хуже. Naïve Bayes – это простое и быстрое решение, которое, тем не менее, показывает высокие результаты. SVM – показывает высокую точность и устойчивость к переобучению. Так же, если выбрано подходящее ядро, алгоритм хорошо работает, даже если данные не линейно разделимы. Однако, Random Forest в большинстве случаев показывает лучшие результаты и является простым в интерпретации решением. Плюсы данного подхода — это скорость, масштабируемость и отсутствие параметров. Logistic Regression (логистическая

регрессия) имеет хорошую вероятностную интерпретацию, справляется с коррелирующими характеристиками. CNN (сверточная нейронная сеть) является одним из лучших алгоритмов по распознаванию и классификации изображений. Но большое количество параметров, влияющих на результат, делают работу с CNN достаточно сложной [36].

Для каждого из описанных алгоритмов входным вектором выступал вектор характеристик, извлеченных из картин на первом этапе работы. Для того, чтобы определить, что выбор классификаторов был оптимальным, были проведены эксперименты с различными наборами входных характеристик. Реализация алгоритмов находится в пакете Scikit-learn Python.

## Глава 3. Эксперименты

Для оценки вклада каждой характеристики был проведен тест: характеристика удалялась и происходила повторная классификация. Результаты данного теста представлены на рисунке 7.

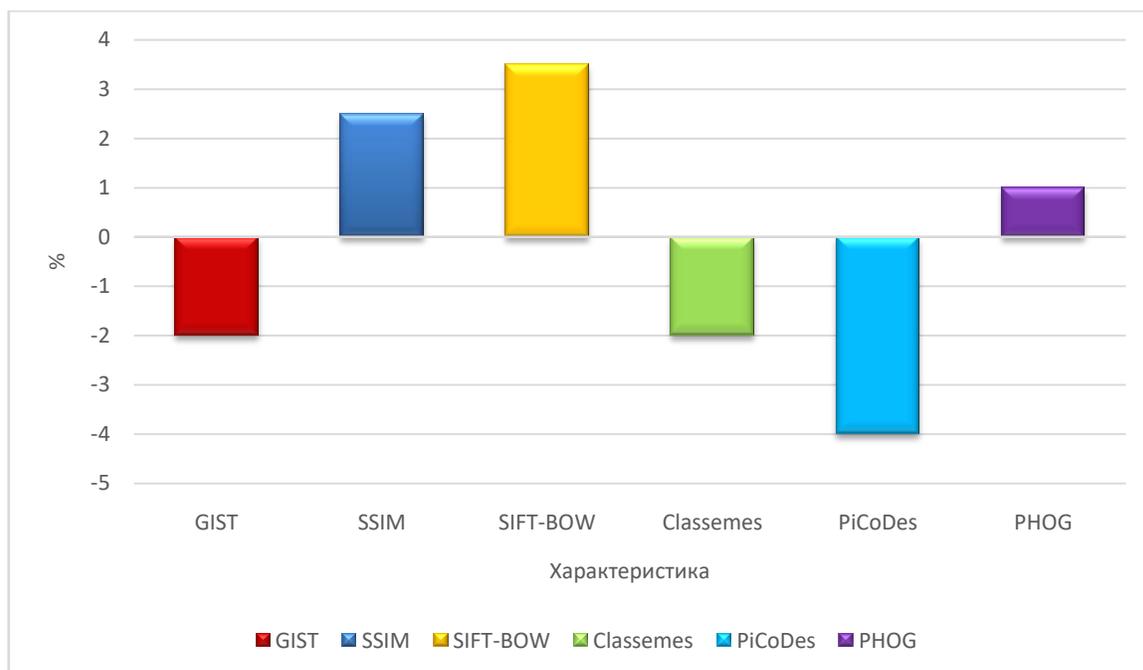


Рис. 7. Вклад каждой характеристики. Показывает насколько снижается точность, если характеристика удалена (большее значение показывает более значительный вклад).

Тестирование классификатора проводилось на коллекции картин, содержащей 75.373 полотна, принадлежащих 5 современным стилям живописи (абстракционизм, поп-арт, фигуратив, гиперреализм, уличное искусство). Была применена 10-fold кросс валидация. В качестве меры качества классификации была выбрана точность (Accuracy).

Алгоритм классификации	Naïve Bayes	SVM	CNN	Random Forest	Logistic Regression
Classemes	35%	46%	43%	43%	43%
PiCoDes	40%	44%	37%	40%	41%
GIST	30%	26%	32%	34%	31%

PHOG	32%	5%	33%	38%	24%
SSIM	33%	42%	34%	40%	36%
SIFT	31%	43%	32%	41%	35%
Classemes + GIST	46%	54%	53%	53%	53%
Classemes + PHOG	35%	39%	41%	42%	42%
PiCoDes + GIST	51%	5%	56%	55%	56%
PiCoDes + PHOG	44%	52%	55%	58%	58%
Classemes + PiCoDes	35%	45%	44%	44%	45%
GIST + PHOG+ SSIM + SIFT	60%	72%	71%	71%	67%
Classemes + SIFT	47%	56%	58%	58%	59%
GIST+PHOG	32%	29%	33%	38%	33%
GIST+SIFT-BOW	34%	42%	36%	41%	38%
GIST+SSIM	35%	45%	35%	45%	38%

Таблица 4. Результаты классификации картин по стилю, с использованием различных характеристик и их сочетаний.

Далее представлены более подробные результаты для исследованных сочетаний визуальных характеристик и методов классификации.

Наилучшую точность показывает SVM с RBF ядром и параметрами  $C=5$  (штраф),  $\Gamma=1$  (ширина ядра). Точность линейного ядра хуже примерно на 2.5%.

Для более глубокой оценки была составлена матрица неточностей (confusion matrix) для каждого апробированного сочетания визуальных характеристик и классификаторов. На рисунке 8 представлена матрица неточностей для SVM + Classemes.

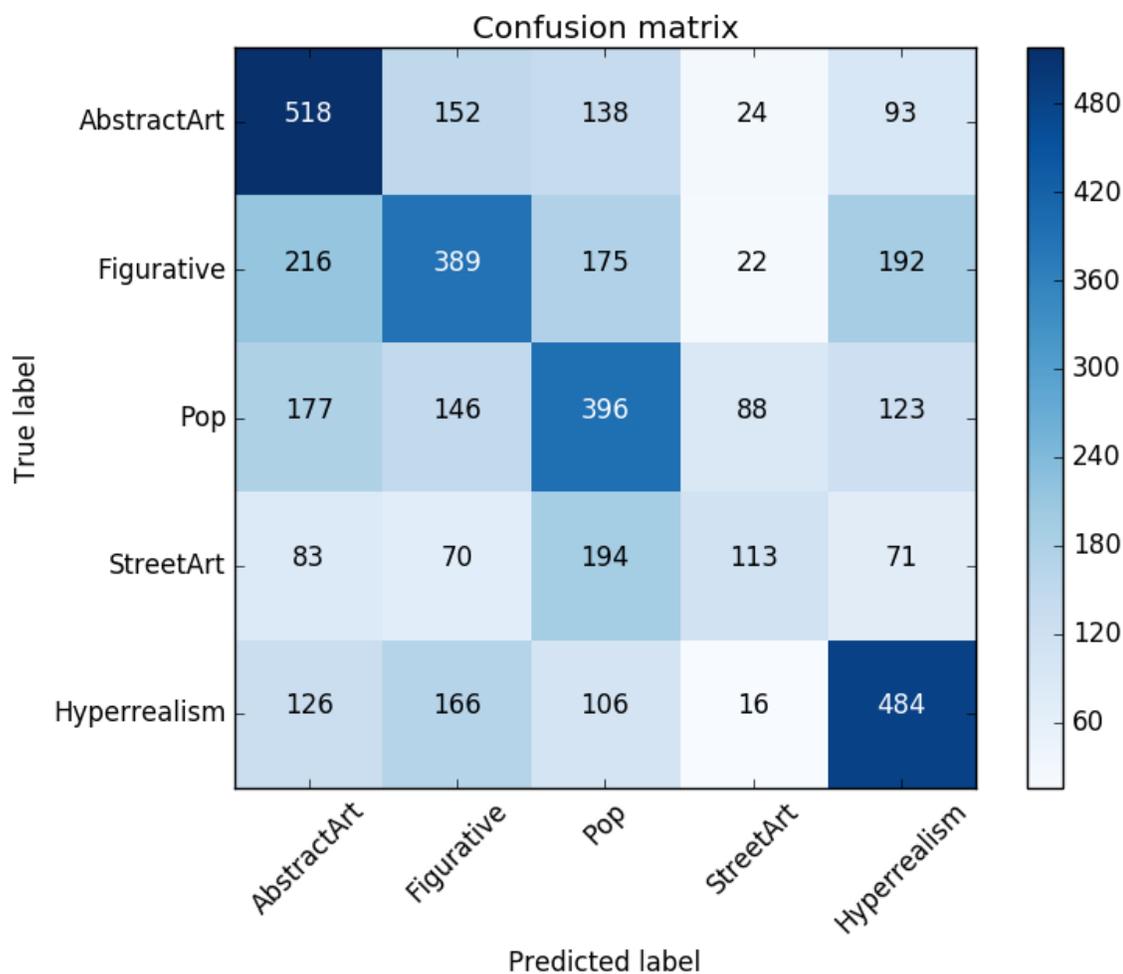


Рис. 8. Матрица неточностей для SVM с Classemes.

На рисунке 9 представлен классификационный отчет.

	precision	recall	f1-score	support
AbstractArt	0.47	0.51	0.49	851
Figurative	0.44	0.47	0.46	1011
Hyperrealism	0.52	0.54	0.53	961
Pop	0.39	0.45	0.41	953
StreetArt	0.50	0.14	0.22	502
avg / total	0.46	0.45	0.44	4278

Рис. 9. Точность, полнота и f-мера SVM с Classemes.

Из матрицы неточностей видно, что уличное искусство в большинстве случаев классифицируется как поп-арт. Это можно объяснить схожестью

стилей, а также малым количеством картин стиля уличное искусство, представленных в выборке.

Заметим также, что для большинства протестированных классификаторов, сочетание высокоуровневых характеристик (Classemes + PiCoDes) дает выигрыш в точности.

Затем было протестировано поведение системы при изменении количества классов. Как и ожидалось, при увеличении количества стилей, точность уменьшается, но результат остается приемлемым для всех стилей. Все возможные комбинации следующих стилей были протестированы: абстракционизм, гиперреализм, поп-арт, фигуратив, уличное искусство. На рисунке 10 представлены средняя, минимальная и максимальная точности для каждого возможного количества классов.

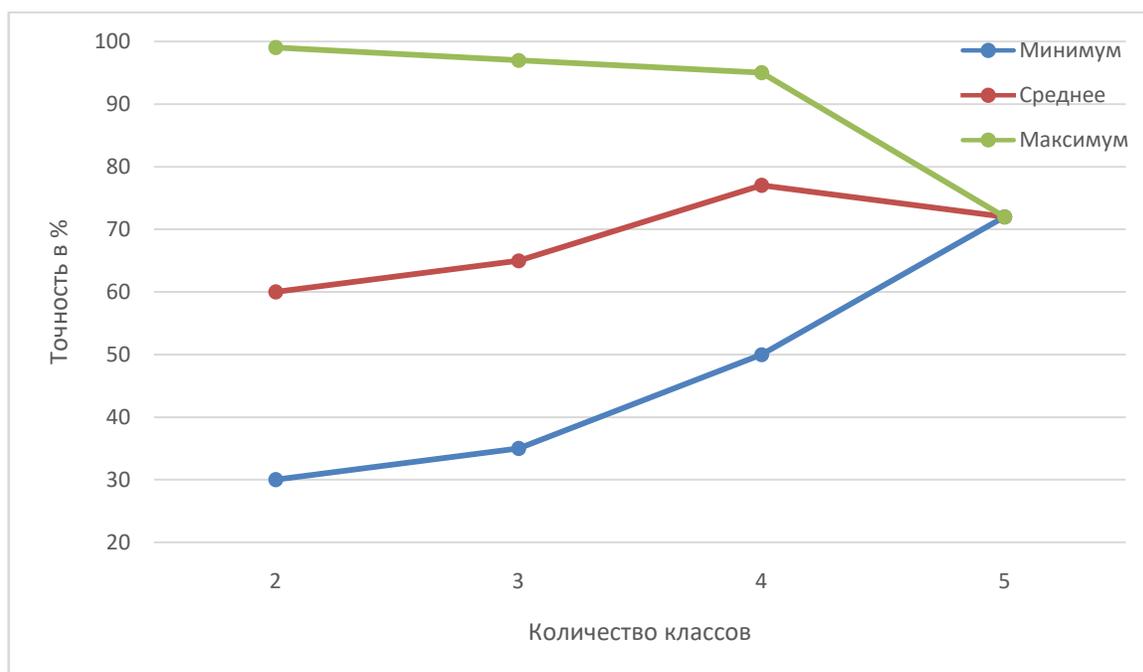


Рис. 10. Минимальная, средняя и максимальная точность в зависимости от количества классов.

Как видно из рисунка 10, для определенного количества классов разница между минимальной и максимальной точностью может быть значительной. Это

можно объяснить через стили, которые были выбраны. Если картины принадлежат похожим стилям, например, уличное искусство и поп-арт – 31.12%, то точность может быть меньше. Тогда как, если картины принадлежат различающимся стилям, например, гиперреализм и абстракционизм – 96.9%, то точность получается намного выше.

В таблице 5 изображены примеры некорректно классифицированных картин. В общем случае, некорректная классификация имела место при схожих стилях, которые сложно разделить даже для человека. Схожести в стилях также можно увидеть в матрице неточностей на рисунке 8.

<b>Поп-арт</b>	<b>Уличное искусство</b>
	





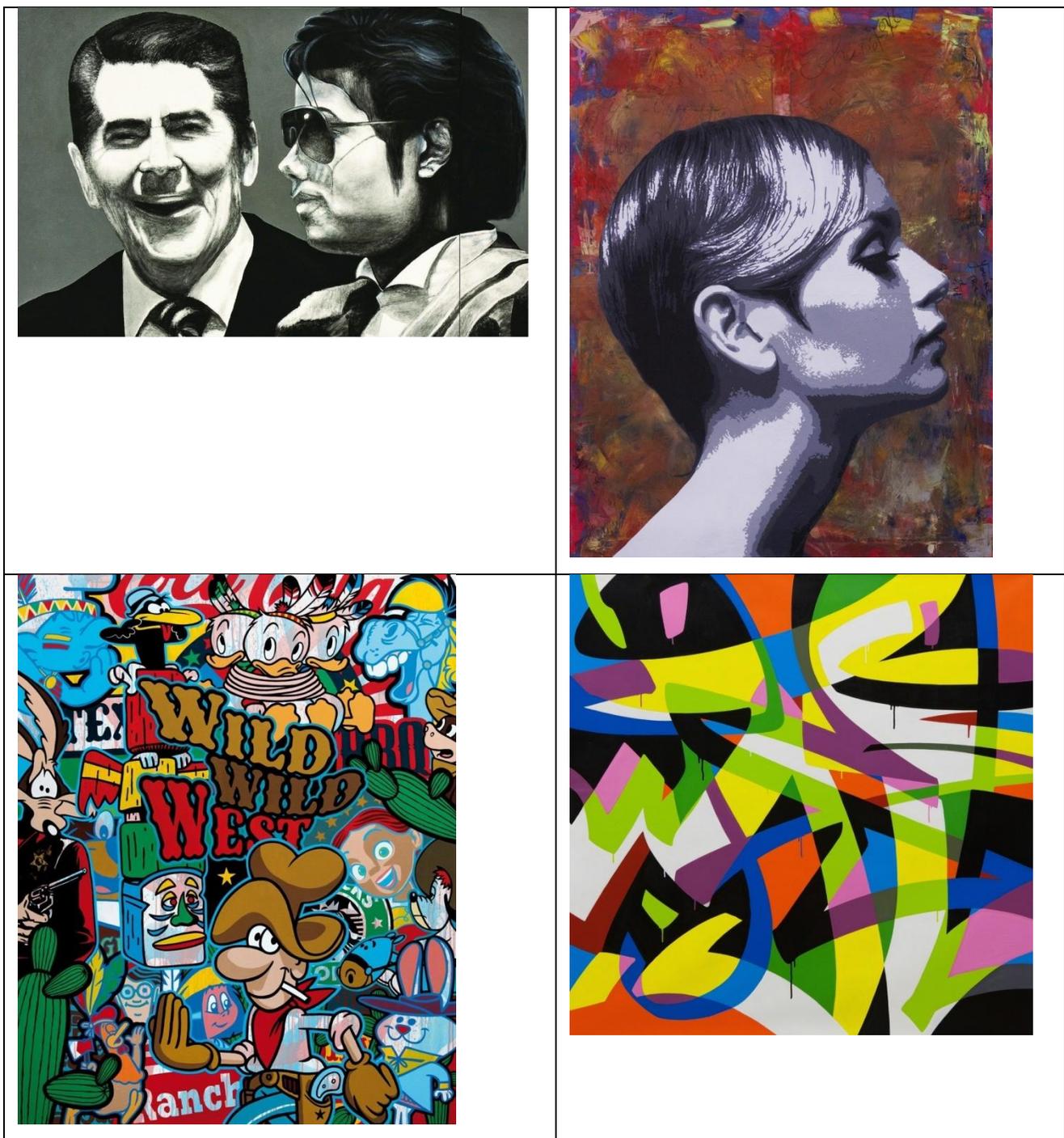


Таблица 5. Примеры некорректно классифицированных картин.

Следует отметить, что классификация с использованием низкоуровневых характеристик является наиболее точной. Данный результат опровергает утверждение в работе [1], что высокоуровневые характеристики всегда показывают наилучшие результаты в задаче классификации картин. Также

данный вывод показывает, что различие современных стилей основывается не на изображенных предметах, а в большей степени на цветах, мазках и т.д.

Тогда как время классификации на данный момент не является решающим фактором, заметим, что время классификации на компьютере с процессором Intel® Core™ i3 1.90GHz в среднем равно 10 секунд.

### 3.1 Сравнение с другими известными решениями

Были отмечены несколько важных проблем при попытке сравнения предложенного решения с другими: использование разных коллекций произведений, фокусировка на определение жанра и автора, а не стиля, а также отсутствие исходного кода. В связи с этим, важнейшими факторами остаются размер коллекции и средняя точность определения стиля.

Размер коллекции в данной работе больше, чем во всех других. Каждый исследователь использовал не только различное количество изображений в классе, но также и разное количество классов. В таблице 6 представлены результаты данной работы и других.

<b>Автор</b>	<b>Метод</b>	<b>Коллекция</b>	<b>Точность</b>
Данная работа	SVM+низкоуровневые характеристики	70.000	72%
[1]	Classemes + Boost	50.000	31.77%
[2]	Discriminative Semantic Model	490	65.4%
[29]	PiCoDes + NN	40,724	38%

Таблица 6. Средняя точность данной работы в сравнении с другими работами: метод классификации, количество картин. Более высокая точность означает более высокое качество классификации.

Для интерпретации полученных результатов, было произведено сравнение с результатами, описанными в работе [37] для ручной классификации, в которой сказано, что «приемлемая» точность автоматической классификации – 70%.

## Глава 4. Прототип приложения

На языке Java было реализовано веб-приложение для классификации картин по стилю. Домашняя страница приложения представлена на рисунке 11.

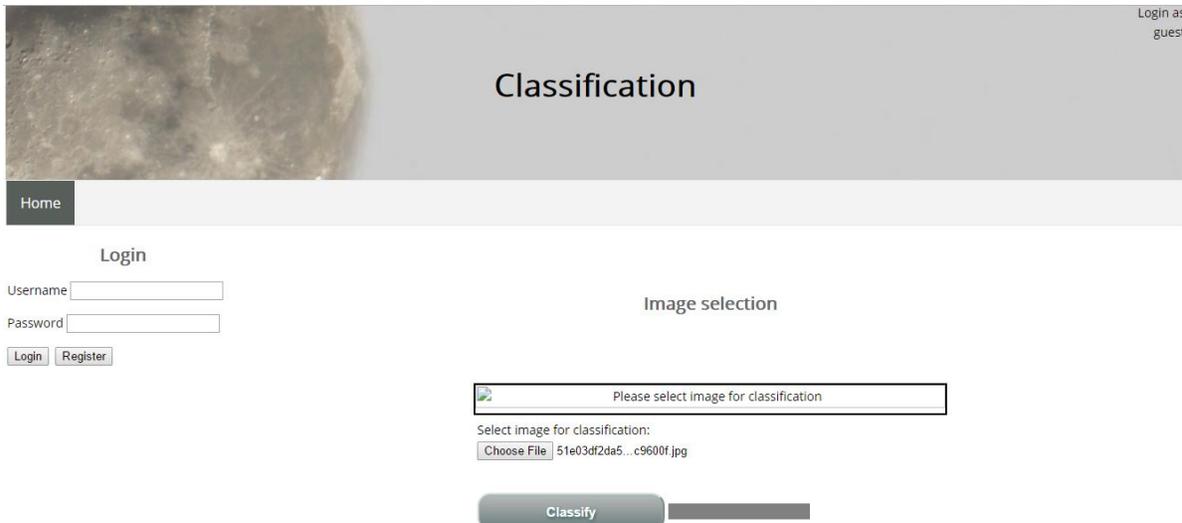


Рис 11. Домашняя страница прототипа на языке Java.

Приложение было создано в качестве «помощника» искусствоведа. Пользователи по желанию регистрируются в системе (по умолчанию каждый заходит как гость). Существуют администраторы. Пользователь имеет возможность зайти в систему по сети или с того же самого компьютера. При входе пользователь может использовать логин и пароль. Пользователь имеет возможность произвести классификацию картин по стилю и в качестве результата получить процент уверенности принадлежности картины каждому из стилей. Если пользователь зарегистрирован, то в личном кабинете он может просматривать ранее загруженные картины и результаты их классификации. Также в личном кабинете пользователь может удалять картины.

Администраторы имеют возможность добавлять картины в обучающую выборку, удалять картины из обучающей выборки и запрашивать тренировку классификатора.

Возможности Гостя:

1. Загрузка картины.

2. Определение стиля.

Возможности зарегистрированного пользователя:

1. Загрузка картины.
2. Определение стиля.
3. Просмотр ранее загруженных данным пользователем картин с результатами.

Возможности администратора:

1. Загрузка картин.
2. Запуск обучающего алгоритма на загруженных картинах.
3. Удаление картин.

Интерфейсы:

1. Для входа в систему.
2. Для регистрации пользователей.
3. Для загрузки изображения.
4. Для выдачи результатов.
5. Для просмотра ранее загруженных изображений.

Таблицы в базе данных MSSQL:

1. Пользователи.
2. Администраторы.
3. Картины: ссылки на изображения, стиль, дополнительная информация (автор, год, направление и т.д.), процент уверенности принадлежности определенному стилю, итоговый стиль (по большему проценту).
4. Стили.

Диаграмма классов представлена на рисунке 12.

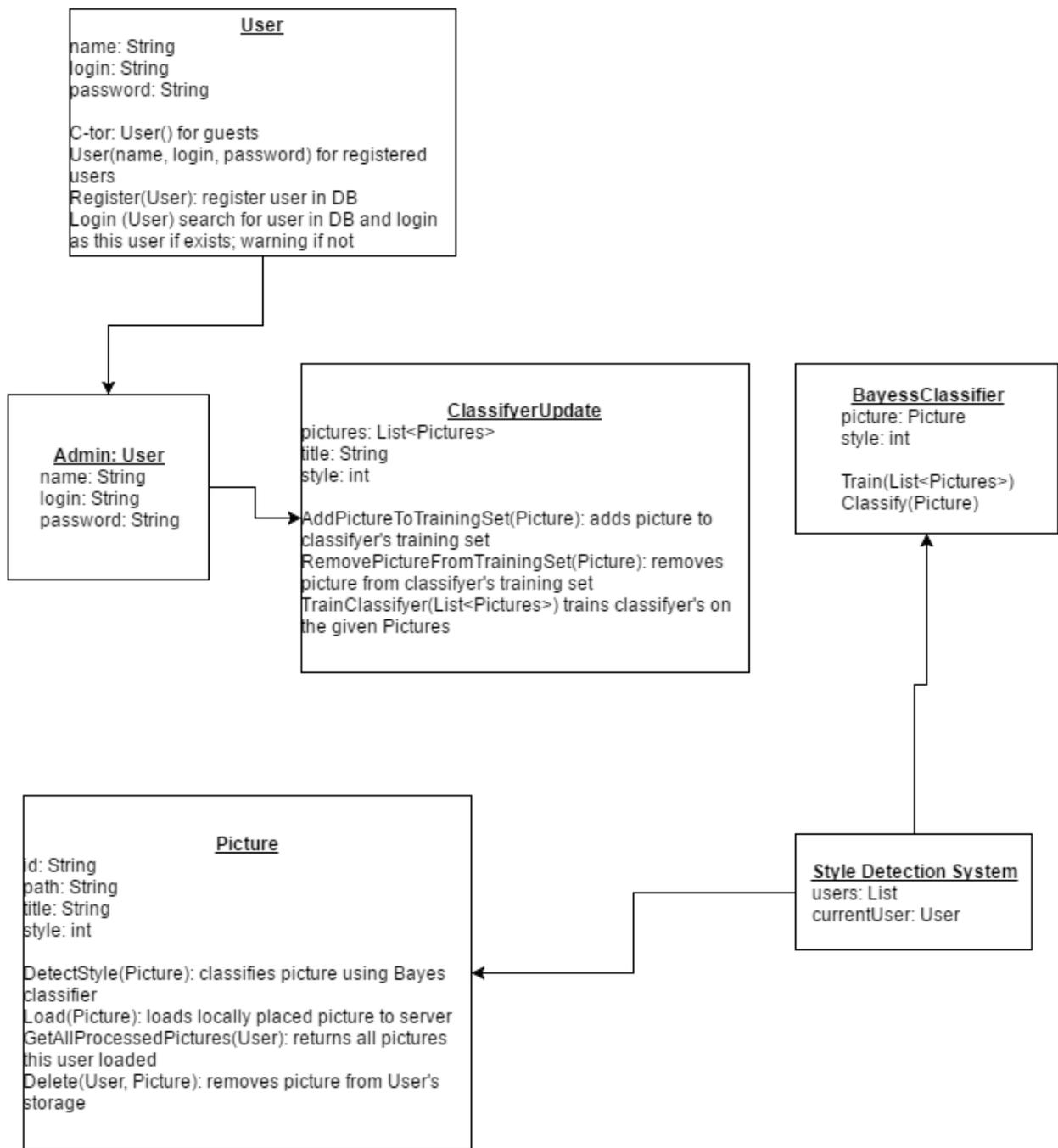


Рис. 12. Диаграмма классов на примере Наивного Байесовского классификатора.

Также пользователь имеет возможность согласиться или не согласиться с предложенным стилем картины. В данном случае Администратор может решить сделать повторную тренировку классификатора.

## Выводы

Подводя итог проделанной работе, следует подчеркнуть, что все цели, поставленные ранее, были достигнуты. Был реализован алгоритм автоматического сбора картин с сайта Artsy. Были изучены актуальные на данный момент алгоритмы извлечения визуальных характеристик изображения. Было дано определение понятию «стиль» картины. Были реализованы алгоритмы извлечения низкоуровневых и высокоуровневых характеристик. Каждый вектор характеристик проецировался в пространство с меньшей размерностью. Также были имплементированы и протестированы различные методы классификации и нейронные сети. В ходе экспериментов было выделено сочетание характеристик, дающее наибольшую точность в определении стиля. Полученные результаты сравнивались с результатами, описанными в других работах. Точность определения стиля в данной работе, а также в работах [1], [2], [29] представлены на рисунке 13.



Рис. 13. Сравнение точности определения стиля в различных работах.

Путем проведения различных экспериментов было доказано, что низкоуровневые характеристики более точно характеризуют картины современных стилей живописи. Данный результат опровергает утверждение в работе [1], что высокоуровневые характеристики всегда показывают наилучшие результаты в задаче классификации картин.

Также было разработано веб-приложение на языке Java. Подробности реализации описаны в главе 4. Разработанный программный модуль может быть применен для автоматизированного определения стиля картины.

Исходный код расположен по ссылке [38].

## Заключение

Исследование показало, что посредством применения различных методов классификации, можно успешно определять стиль картины. Данный процесс происходит полностью автоматически, что позволяет обрабатывать большие объемы данных за малый промежуток времени.

Для дальнейшего исследования выделены следующие направления:

1. Разработка метода извлечения характеристик из изображения. Как можно заметить, наибольшая точность достигается при комбинации низкоуровневых характеристик. Однако полученный вектор имеет довольно большую размерность. Исследование других возможностей перевода изображения в вектор, может дать выигрыш не только в точности, но и в скорости работы.
2. Изучение возможности классификации картин по другим характеристикам: жанр, художник, временной период. Не смотря на то, что множество работ посвящено определению художника, данное направление остается не до конца проработанным. Точность в определении художника составляет  $\approx 60\%$  в работе [1].
3. Доработка разработанного веб-приложения и его апробация.

Создание такого программного комплекса может стать объектом диссертации.

## Список использованной литературы

1. Elgammal A., Saleh B. Large-scale Classification of Fine-Art Paintings: Learning The Right Metric on The Right Feature // International Journal for Digital Art History, 2016. No 2, P. 71-95.
2. Elgammal A., Arora R. Towards Automated Classification of Fine-art Painting Style: a Comparative Study // Multimedia Tools and Applications, 2014. Vol. 75, No 7, P. 3565-3591.
3. Arnheim R. Visual Thinking. M.: University of California Press, 1969. P. 346.
4. Fichner-Rathus L. Foundations of Art and Design. M.: Cengage Learning, 2008. P. 318.
5. Hendriks E., Li J., Wang J. Z., Yao L. Rhythmic brushstrokes distinguish van gogh from his contemporaries: Findings via automated brushstroke extraction // IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012. Vol. 34, No 3, P. 1159-1176.
6. Brasoveanu A., Daubechies I., Jafarpour S., Hughes S., Polatkan G. Detection of forgery in paintings using supervised learning // Proceedings of the 16<sup>th</sup> IEE international conference on Image processing, 2009. P. 2885-2888.
7. Stork D. Computer vision and computer graphics analysis of paintings and drawings: An introduction to the literature // Computer Analysis of Images and Patters, 2009. Vol. 14, No 1, P. 9-24.
8. Bentkowska-Kafel A., Coddington J. Computer Vision and Image Analysis // Art: Proceedings of the SPIE Electronic Imaging Symposium, 2010. P. 18-22.
9. Lombardi T. The Classification of Style in Fine-Art Painting // ETD Collection for Pace University, 2005. No 3, P. 95-110.
10. Kammerer P., Sablatnig R., Zolda E. Hierarchical classification of paintings using face- and brush stroke models // Pattern Recognition, 1998. Vol. 14, No 2, P. 172-174.

11. Li J., Wang J. Z. Studying digital imagery of ancient paintings by mixtures of stochastic models // Image Processing, 2004. Vol. 13, No 3, P. 340-353.
12. Farid H., Lyu S., Rockmore D. A digital technique for art authentication // Proceedings of the National Academy of Sciences of the United States of America, 2004. Vol. 101, No 49, P. 17006-17010.
13. Berezhnoy I., Brevdo E., Daubechies I., Johnson C. R., Hendriks E., Hughes S. M., Li J., Postma E., Wang J. Z. Image processing for artist identification // Signal Processing Magazine, 2008. Vol. 25, No 4, P. 37-48.
14. Berezhnoy I. E., Herik H. J., Postma E. O. Automatic extraction of brushstroke orientation from paintings // Machine Vision and Applications, 2009. Vol. 20, No 1, P. 1-9.
15. Lowe D. G. Distinctive image features from scale-invariant keypoints // International Journal of Computer Vision, 2005. Vol. 5, No 2, P. 91-110.
16. Dalal N., Triggs B., Histograms of oriented gradients for human detection // International Conference on Computer Vision & Pattern Recognition, 2005. P. 886-893.
17. Fahad M. V., Khan S., Weijer J. Who painted this painting? // The CREATE Conference, 2010. P. 329-333.
18. Abdel-Hakim A. E., Farag A. A. A sift descriptor with color invariant characteristics // International Conference on Computer Vision & Pattern Recognition, 2006. P. 1978-1983.
19. Fitzgibbon A., Szummer M., Torresani L. Efficient Object Category Recognition Using Classemes // Computer Vision – ECCV 2010 Workshops, 2010. P. 776-789.
20. Bue A. D., Carneiro G., Costeira J. P., Silva N. P. Artistic image classification: An analysis on the printart database // Proceedings of the 12<sup>th</sup> European conference in Computer Vision, 2012. P. 143-157.

21. Abe K., Elgammal A., Saleh B. Knowledge discovery of artistic influences: A metric learning approach // 5<sup>th</sup> International Conference on Computational Creativity, 2014. P. 13-17.
22. Ключина А. Словарь специальных терминов в живописи. <http://cosmograph.ru/culture/libs-termin/>
23. Стили и направления живописи. <http://vertex-art.ru/statyi/11-stili-i-napravlenija-zhivopisi.html>
24. Oliva A., Torralba A. Modeling the shape of the scene: A holistic representation of the spatial envelope // International Journal of Computer Vision, 2001. Vol. 42, No 3, P. 145-175.
25. Amsaleg L., Douze M., Jegou H., Sandhawalia H., Schmid C. Evaluation of GIST descriptors for web-scale image search // Proceedings of the ACM International Conference on Image and Video Retrieval, 2009. P. 8-14.
26. Bergamo A., Fitzgibbon A. W., Torresani L. Picodes: Learning a compact code for novel-category recognition. In Advances in Neural Information // Proceedings of the 24<sup>th</sup> International Conference on Neural Information Processing Systems, 2011. P. 2088-2096.
27. Hinton G. E., Krizhevsky A., Sutskever I. Imagenet classification with deep convolutional neural networks // Proceedings of the 25<sup>th</sup> International Conference on Neural Information Processing Systems, 2012. P. 1097-1105.
28. Bengio Y., Bottou L., Haffner P., LeCun Y. Gradient-based learning applied to document recognition // Proceedings of the IEEE, 2002. Vol. 86, No 11, P. 2278-2324.
29. Bar Y., Levy N., Wolf L. Classification of Artistic Styles using Binarized Features Derived from a Deep Neural Network // Computer Vision – ECCV 2014 Workshops, 2014. P. 71-84.

30. Южаков Г. Б. Алгоритм быстрого построения дескрипторов изображения, основанных на технике гистограмм ориентированных градиентов // ТРУДЫ МФТИ, 2013. № 3. С. 84-91.
31. Zhang C. Computer Vision: What is a GIST descriptor?  
<https://www.quora.com/Computer-Vision-What-is-a-GIST-descriptor>
32. Wikipedia: SSIM. <https://ru.wikipedia.org/wiki/SSIM>
33. Национальный открытый Университет: Обзор дескрипторов ключевых точек. <http://www.intuit.ru/studies/courses/10621/1105/lecture/17983?page=2>
34. Artsy. <https://www.artsy.net/collect?medium=painting>
35. Bergamo A., Fang C., Torresani L. VLG extractor.  
[http://vlg.cs.dartmouth.edu/projects/vlg\\_extractor/vlg\\_extractor/Home.html](http://vlg.cs.dartmouth.edu/projects/vlg_extractor/vlg_extractor/Home.html)
36. Chen E. Choosing a Machine Learning Classifier.  
<http://blog.echen.me/2011/04/27/choosing-a-machine-learning-classifier/>
37. Cunningham D., Feixas M., Fleming R., Rigau J., Sbert M., Wallraven C. Computational Aesthetics 2008: Categorizing art: Comparing humans and computers // J. Computers and Graphics, 2009. Vol. 33, No 4, P. 484-495.
38. Github: Style classification. <https://github.com/LiloS/StyleClassification>