

Содержание

Введение	6
1 Аналитический обзор	8
1.1 Обзор предметной области	8
1.2 Обзор исходных данных	9
1.3 Обзор аналогичных систем	10
1.4 Требования к разрабатываемой системе	15
2 Решение поставленной задачи	16
2.1 Поиск и генерация набора данных для обучения нейронной сети	16
2.2 Подготовка набора данных предобработанных изображений	19
3 Краткая характеристика программных средств	25
3.1 Разработка системы распознавания	25
3.2 Тестирование разработанной системы	27
Заключение	32
Список литературы	33

					МИВУ 01.03.02-00.000 ПЗ			
Из	Лист	№ докум.	Подп.	Дата				
Студент	Лазарева Е.Ю.				Разработка системы распознавания точечной маркировки на поверхностях черных металлов. Пояснительная записка	Лит	Лист	Листов
Руков.	Астафьев А.В.					У	5	33
Конс						МИ ВлГУ ПМИ-117		
Н.контр.	Рыжкова М.Н.							
Утв.								

Введение

В рамках данной выпускной квалификационной работы представлен процесс разработки системы распознавания точечной маркировки на цифровых изображениях черных металлов. В настоящее время, в связи с необходимостью развития промышленности, требуется введение новых наукоемких технологий по контролю движения продукции. Реализация такого контроля возможна с помощью внедрения систем автоматической идентификации и распознавания. Для контроля движения продукции используются маркировки промышленных изделий. На новейших промышленных предприятиях контроль движения продукции осуществляется специализированными средствами - системами автоматической идентификации маркировки (САИМ). Основная сложность при использовании САИМ заключается в задаче локализации промышленной продукции, которую усложняют наличие неравномерной освещенности и резкости (засвечивание изображения искусственным и солнечным светом, неверная настройка фокуса), присутствие шумов и размытия, прочих лишних объектов. Автоматизация этой сферы деятельности является мало проработанной и, чаще всего, выполняется вручную. Этим обусловлена актуальность данной работы.

Цель данной выпускной квалификационной работы – разработать систему распознавания точечной маркировки на цифровых изображениях черных металлов для автоматизации производства трубопроводной продукции, протестировать ее и сделать выводы о ее работе с предобработанными и непредобработанными изображениями.

Задачи данной выпускной квалификационной работы:

- 1). провести анализ предметной области, проанализировать аналогичные системы, произвести краткий обзор исходных данных, предъявить требования к разрабатываемой системе;
- 2). произвести математическое моделирование разрабатываемой системы;
- 3). проанализировать алгоритмы, используемые при разработке системы и описать их;
- 4). найти либо сгенерировать набор данных для обучения нейронной сети;

					МИВУ 01.03.02-00.000 ПЗ	Лист
						6
Из	Лист	№ докум.	Подп.	Дата		

- 5). подготовить набор данных с предобработанными изображениями для тестирования эффективности работы системы;
- 6). реализовать систему для распознавания маркировки на изображении;
- 7). протестировать разработанную систему и сделать выводы о ее работе с непредобработанными и предобработанными изображениями.

Таким образом, общая структура данной выпускной квалификационной работы состоит из трех глав и выглядит следующим образом:

- 1). в первой главе производится анализ технического задания, а именно:
 - обобщается предметная область;
 - определяются исходные данные к проекту;
 - кратко обобщаются некоторые аналогичные алгоритмы;
 - выставляются требования к разрабатываемой информационной системе
- 2). во второй главе:
 - приводится описание используемых в разработке алгоритмов;
 - описывается процесс поиска и генерации набора данных для обучения и тестирования работы нейросети;
 - производится процесс подготовки набора данных предобработанных изображений для тестирования эффективности работы нейросети;
- 3). в третьей главе описывается процесс реализации разработанной системы распознавания, а также приводятся результаты тестирования работы системы с использованием непредобработанных и предобработанных изображений.

1 Аналитический обзор

1.1 Обзор предметной области

Маркировка красками или специальными чернилами - один из старейших способов маркировки. Наибольшее распространение сегодня маркировка красками имеет в легкой, электронной и пищевой промышленности, где используется оборудование типа IMAGE и DOMINO. Эти маркеры оборудованы специальными головками с соплами, с помощью которых возможно нанесение любого вида маркировки. Контрастность изображения обеспечивается различным количеством точек на единицу площади изображения.

Однако клеймение горячего металла красками невозможно: лучшие образцы лакокрасочной продукции выдерживают температуру не более 1000°C. Плюс к этому в условиях металлургических предприятий и горячих цехов трудно обеспечить необходимую чистоту и приемлемый температурный режим, требуемые для безотказной работы краскоструйных сопел.

Затрудняет использование краскоструйных маркеров и характер поверхности многих металлоизделий. Окалина, пыль, неровности, и т.п. делают маркировку красками фрагментарной и недолговечной. Все перечисленные факторы делают невозможным или экономически невыгодным использование краскоструйных маркеров в горячих цехах и при машинах непрерывного литья заготовок.

Краски используют для маркировки труб горячекатаных (на выходе из прокатного стана, когда температура изделия становится приемлемой), маркировки слябов и блюмов и т.п.

Оборудование для маркировки краской состоит из следующих узлов:

- головка с 7 или 9 соплами;
- манипулятор для перемещения головки с соплами;
- специальная панель для подготовки краски;
- пульт управления;
- терминал для оператора, с помощью которого можно вручную задавать параметры маркировки;
- электронная система управления (программное обеспечение).

					МИВУ 01.03.02-00.000 ПЗ	Лист
						8
Из	Лист	№ докум.	Подп.	Дата		

Краскосопла имеют увеличенный диаметр отверстий, что препятствует их засорению, и приспособлены для операции самоочищения. Минимальная высота наносимой маркировки - 19 миллиметров, что может служить ограничением в использовании такого оборудования для некоторых видов металлоизделий. С помощью оборудования для маркировки краской можно наносить не только буквенно-цифровые коды, но и штрих-коды. Однако, на данном оборудовании можно нанести только бинарные коды (содержащие минимум информации о металлоизделии), для считывания которых нужна специальная аппаратура (промышленные сканеры).

Оборудование для нанесения маркировки краской имеет высокую степень надежности. Однако, при использовании маркировки в металлургии с помощью красок и специальных чернил необходимо учитывать, что для полного высыхания краски требуется определенное время, при этом чем горячее металлоизделие, тем больший срок требуется для высыхания краски. При высокой температуре маркируемых металлоизделий также повышаются эксплуатационные расходы на краскоструйное оборудование: увеличивается расход красок, требуется больше времени для технического обслуживания агрегата и т.п.

Основные характеристики:

- постоянные метки для идентификации;
- высокая степень разборчивости символов для обычного или автоматического считывания в условиях неповреждения метки;
- разборчивая маркировка на грубых и неровных поверхностях;
- высокая скорость маркировки [3].

Основное оборудование, необходимое для нанесения маркировки – это многопистолетные установки нанесения краски или чернил.

1.2 Обзор исходных данных

В качестве входных данных были определены следующие:

1). Google Colab — это бесплатный облачный сервис на основе Jupyter Notebook, он предоставляет всё необходимое для машинного обучения прямо в браузере, даёт бесплатный доступ к невероятно быстрым GPU и TPU;

					МИВУ 01.03.02-00.000 ПЗ	Лист
						9
Из	Лист	№ докум.	Подп.	Дата		

2). язык программирования Python, так как данный язык является языком высокого уровня, что значительно упрощает работу над разработкой информационной системы, и позволяет реализовать возможность создания графического интерфейса;

3). библиотека Pandas - программная библиотека на языке Python для обработки и анализа данных;

4). библиотека OpenCV - это библиотека с открытым исходным кодом, разработанная на языке C++, которая используется разработчиками и исследователями при решении задач, связанных с компьютерным зрением;

5). библиотека Numpy – это также библиотека с открытым исходным кодом, которая позволяет работать с массивами и вычислять сложные арифметические операции;

6). библиотека Tensorflow - открытая программная библиотека для машинного обучения, разработанная компанией Google для решения задач построения и тренировки нейронной сети с целью автоматического нахождения и классификации образов, достигая качества человеческого восприятия;

7). библиотека Matplotlib - библиотека на языке программирования Python для визуализации данных двумерной графикой, получаемые изображения могут быть использованы в качестве иллюстраций в публикациях.

Выходными данными будет являться текст распознанной точечной маркировки.

1.3 Обзор аналогичных систем

К аналогичным системам можно отнести программы для распознавания почтовых индексов и шрифта Брайля (тактильный рельефно-точечный шрифт, который предназначен для чтения и письма слабовидящими и totalmente незрячими людьми). Точечную маркировку планируется наносить по формату почтового индекса (по образцу кодового штампа, рисунок 1). Также, точечная маркировка визуально схожа с текстом, написанным при помощи шрифта Брайля, так как каждая буква изображается в виде точек (рисунок 2).

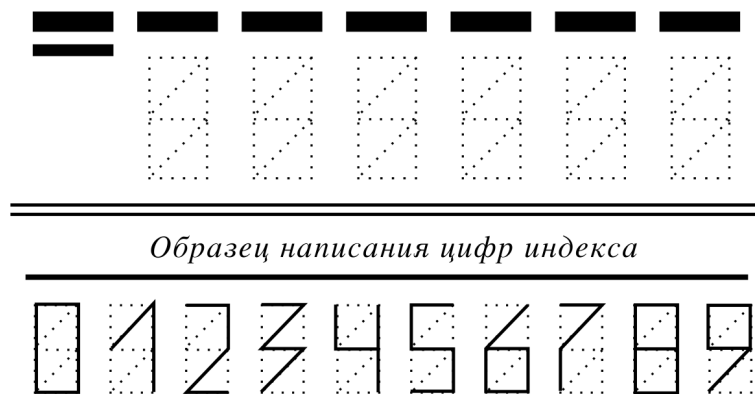


Рисунок 1 – Кодовый штамп

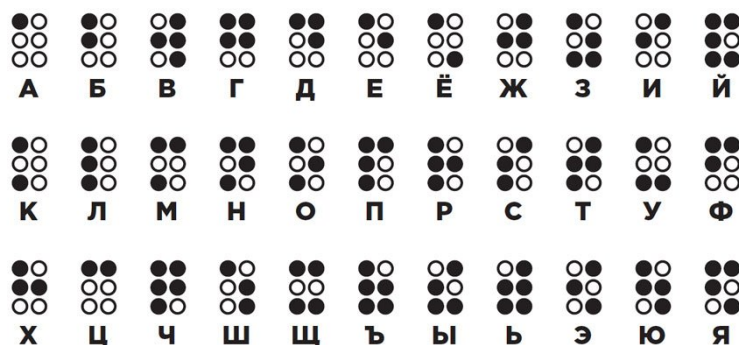


Рисунок 2 – Русский алфавит шрифтом Брайля

1.3.1. Алгоритм распознавания кодового штампа

В данном алгоритме на вход системе поступает отсканированное изображение кодового штампа, на котором вручную был написан почтовый индекс [7]. Авторами алгоритма предлагается следующая последовательность действий:

- 1). нахождение угла наклона последовательности цифр;
- 2). коррекция изображения с целью расположить цифры без наклона (поворот изображения на нужный угол и в нужную сторону);
- 3). сегментирование последовательности цифр (в результате каждая цифра становится отдельным элементом общей последовательности);
- 4). нормирование изображения с целью установления его размера согласно общему стандарту (например, 600x800 точек) – это поможет увеличить качество распознавания;
- 5). проведение анализа нормированного изображения по контрольным точкам (детекторам), таким образом нужно отследить, закрашены ли точки или нет;
- 6). анализ полученных комбинаций – сравнение полученного результата с обучающей выборкой (данное действие производится при помощи нейросети).

1.3.2. Алгоритм распознавания шрифта Брайля

В качестве примера была взята программа распознавания шрифта Брайля авторов Конобеевой Н.Н, Полуниной А.А., Круглякова А.В. и Белоусова А.В.[7].

В данном алгоритме рассматривается метод сравнения с прототипом. В процессе распознавания шрифта Брайля на подготовительном этапе определяется полнота группы ячеек, на основе чего получаются данные о ширине и высоте группы, и, зная расстояние между группами, становится возможным проведение анализа для распознавания шрифта Брайля.

Полнота группы – группа, удовлетворяющая полноте первой и второй степени. Первая степень полноты группы – группа, в которой максимальная ширина достигается суммой двух ячеек (в группе должны присутствовать ячейки в 1-м или 2-м или 3-м секторе и в 4-м или 5-м или 6-м секторе). Вторая степень полноты группы – группа, в которой максимальная высота достигается суммой трех ячеек (в группе должны присутствовать ячейки в 1-м или 4-м секторе, во 2-м или 5-м секторе и в 3-м или 6-м секторе).

Стандартная страница с текстом Брайля вмещает до 25 строк по 40–43 ячейки. Наличие или отсутствие точек в ячейке дает определенный символ. Высоты точки в 0,5 мм достаточно для ее распознавания на ощупь.

Для написания шрифта Брайля используют специальный трафарет с подложкой, в который вкладывается специальный лист плотной бумаги, который хорошо держит форму. На рисунке 3 представлен порядок чтения и написания точек в ячейке.

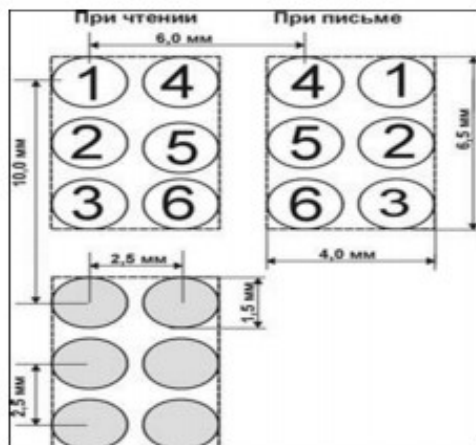


Рисунок 3 – Нумерация мест положения точек в ячейке

Рассмотрим основные этапы процесса распознавания шрифта Брайля.

Для начала необходимо найти и проанализировать все ячейки, двигаясь относительно анализируемой ячейки вправо, влево, вверх и вниз (рис. 4).

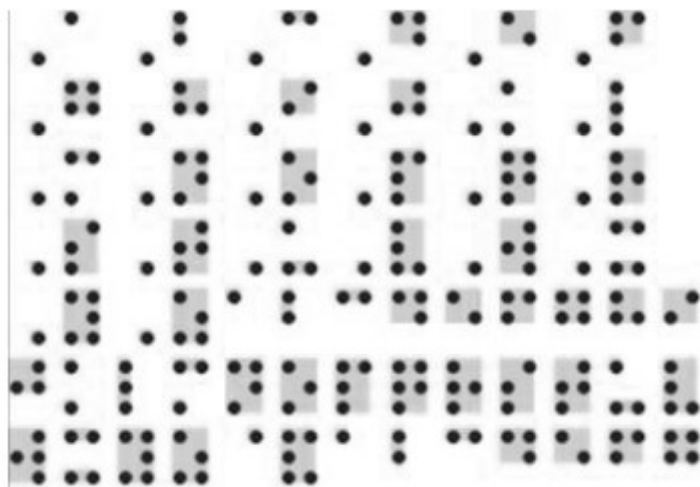


Рисунок 4 – Начало анализа

Затем произвести группировку ячеек, объединяя в группу ячейки, находящиеся в непосредственной близости друг от друга. Проверить полученные группы на полноту.

Анализируя группу, удовлетворяющую полноте первой степени, нужно найти расстояние по горизонтали между группами ячеек (рис. 5).

Анализируя группу, удовлетворяющую полноте второй степени, нужно найти расстояние по вертикали между группами ячеек (рис. 6).

Найти группу, удовлетворяющую правилу полноты. На основе найденной группы определить высоту и ширину группы.

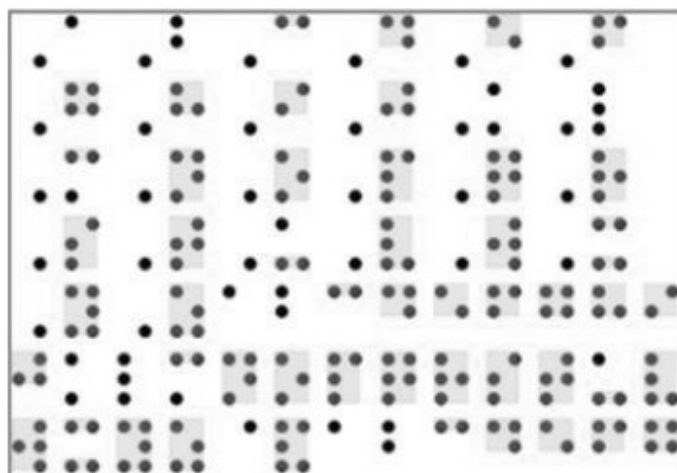


Рисунок 5 - Полнота первой степени

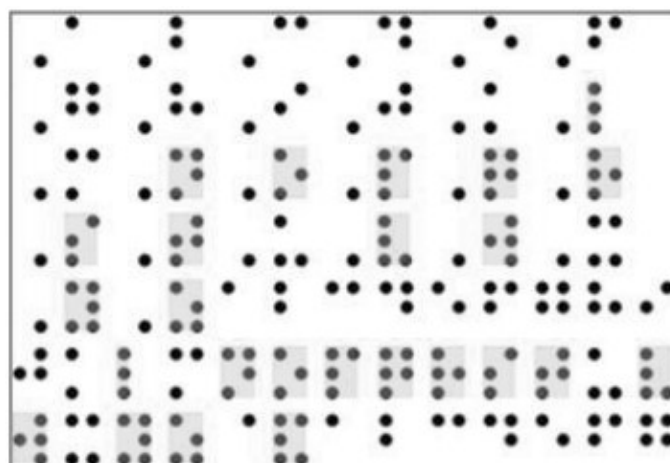


Рисунок 6 - Полнота второй степени

Имея данные о ширине и высоте группы, а также зная расстояние между группами, нужно наложить аналитическую сетку для последующего распознавания шрифта Брайля. Сверяя список всех ячеек, с аналитической сеткой нужно произвести перегруппировку групп. Анализируя группы, производится распознавание шрифта Брайля. В результате работы программы текст успешно распознан и выведен на экран (рис. 7).

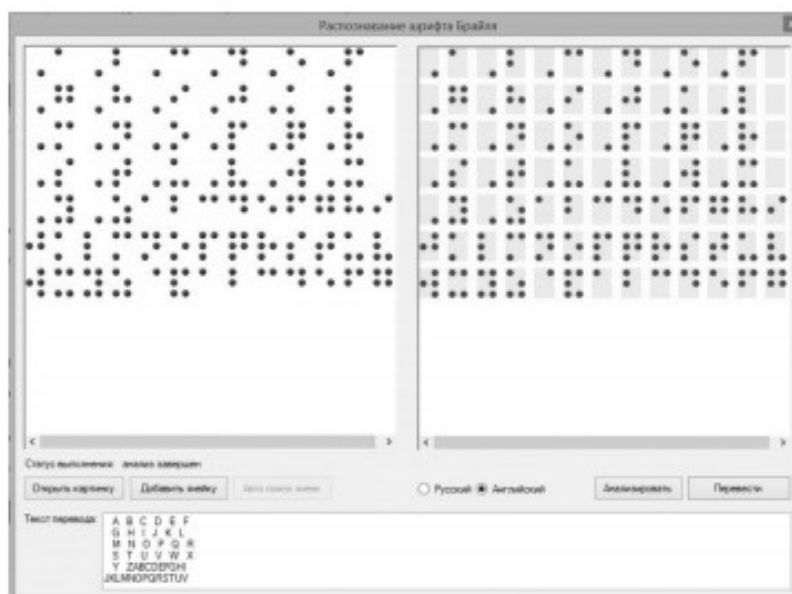


Рисунок 7 - Результат работы программы

Необходимые условия работы алгоритма следующие:

- ячейки не должны быть полыми внутри.
- текст должен соответствовать определенному стандарту, символы и ячейки должны находиться на одинаковом расстоянии друг между другом.
- одна из групп должна удовлетворять полноте.

Из	Лист	№ докум.	Подп.	Дата

1.4 Требования к разрабатываемой системе

Требования к разрабатываемой системе следующие:

- разрабатываемая система должна оперировать со всеми входными данными и получать в результате текст распознанной точечной маркировки;
- каждый алгоритм разрабатываемой системы должен состоять из отдельных элементарных шагов, работающих системно;
- последовательность шагов каждого алгоритма должна быть детерминирована, т.е. после каждого шага должно указываться, какой шаг следует выполнять дальше, либо указываться, когда следует работу алгоритма считать законченной;
- каждый алгоритм должен обладать результативностью, т.е. останавливаться после конечного числа шагов (зависящего от исходных данных) с выдачей результата.

2 Решение поставленной задачи

2.1 Поиск и генерация набора данных для обучения нейронной сети

Для поиска подходящего набора данных для обучения и тестирования разрабатываемой нейронной сети была использована платформа Kaggle.

Kaggle — система организации конкурсов по исследованию данных, а также социальная сеть специалистов по обработке данных и машинному обучению [2].

Среда организована как публичная веб-платформа, на которой пользователи и организации могут публиковать наборы данных, исследовать и создавать модели, взаимодействовать с другими специалистами по данным и инженерами по машинному обучению, организовывать конкурсы по исследованию данных и участвовать в них. В системе размещены наборы открытых данных, предоставляются облачные инструменты для обработки данных и машинного обучения.

Набора данных, содержащего в себе именно изображения черных металлов с нанесенными на них точечными маркировками найти не удалось, поэтому было принято решение генерировать такой набор данных самостоятельно.

В качестве основы был выбран набор данных изображений черных металлов Severstal: Steel Defect Detection, он содержит 12867 изображений [2]. Каждое изображение имеет размер 1600 на 256 пикселей.

«Северсталь» является лидером в области эффективной добычи и производства стали. Они считают, что будущее металлургии требует развития экономических, экологических и социальных аспектов отрасли, и они серьезно относятся к корпоративной ответственности. Компания недавно создала крупнейшее в стране озеро промышленных данных с петабайтами данных, которые ранее были отброшены. Сейчас «Северсталь» использует машинное обучение, чтобы улучшить автоматизацию, повысить эффективность и поддерживать высокое качество своей продукции. Сегодня «Северсталь» использует изображения с высокочастотных камер для работы алгоритма обнаружения дефектов [2].

Пример изображения черного металла из заявленного набора данных представлен на рисунке 8:

					МИВУ 01.03.02-00.000 ПЗ	Лист
						16
Из	Лист	№ докум.	Подп.	Дата		



Рисунок 8 - Изображение из набора данных Severstal: Steel Defect Detection

Перед генерацией набора данных было произведено исследование качества изображений, предложенных заводом «Северсталь». Некоторые изображения оказались непригодны, так как содержали менее 70% полезной площади изображения (площади с изображением именно текстуры металла). Такие изображения были отсортированы в процессе генерации набора данных.

Следующим этапом является задача генерации набора данных изображений черных металлов с нанесенными на них точечными маркировками [9].

Точечная маркировка обладает следующими характеристиками:

- состоит из 3-8 цифр, стоящих рядом друг с другом;
- разрешение каждой цифры – 9 к 16, размер цифры варьируется от 27 на 48 пикселей до 81 на 144 пикселя;
- каждая цифра имеет одинаковый размер относительно остальных цифр в конкретной маркировке;
- способ написания точечной маркировки схож с методом написания почтового индекса;
- обеспечивается 100%-е вхождение маркировки в кадр.

Алгоритм генерации набора данных следующий:

1). по примеру почтового индекса подготавливается основа для генерации цифры, она выглядит следующим образом (рисунок 9):

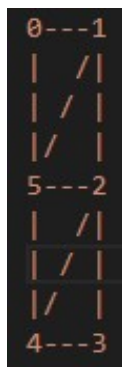


Рисунок 9 - Основа для генерации цифр

- 2). составляется список линий, по которым будут генерироваться те или иные цифры, например, цифра 0 – это линии 0-1, 1-2, 2-3, 3-4, 4-5, 5-0;
- 3). случайным образом задается размер цифры в маркировке;
- 4). случайным образом происходит генерация числа, которое будет нанесено на изображение;
- 5). случайным образом происходит генерация координат расположения маркировки и нанесение ее на изображение;
- 6). сохранение изображения [4].

Блок-схема этого алгоритма представлена на рисунке 10.

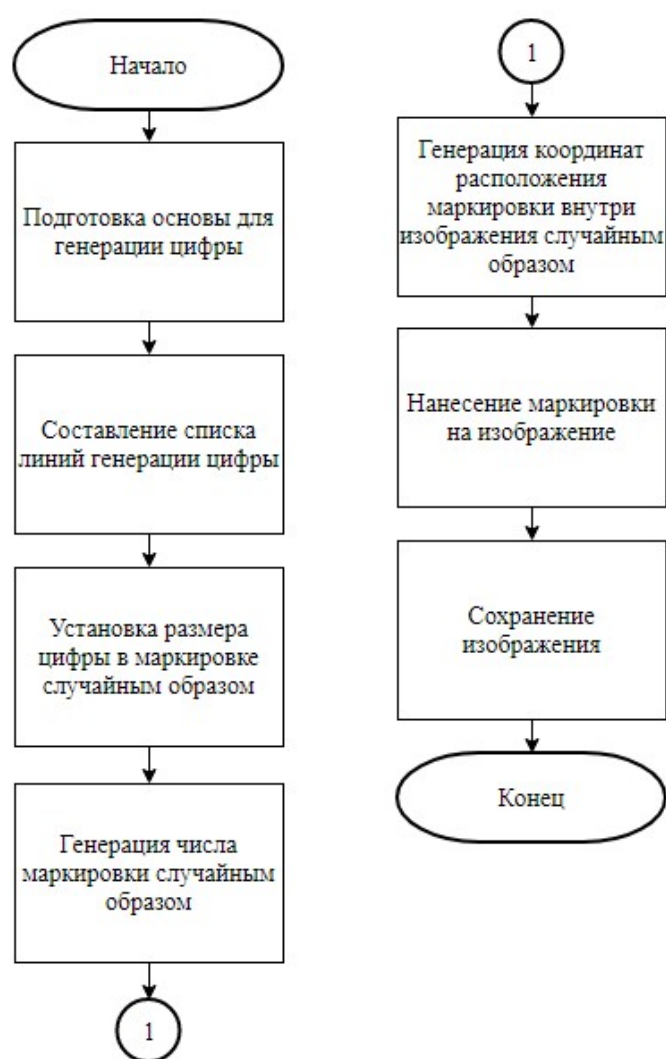


Рисунок 10 - Блок-схема алгоритма генерации маркировки

Таким образом необходимо произвести данные действия с каждым изображением из исходного набора данных. Пример изображения со сгенерированной маркировкой представлен на рисунке 11:

Рисунок 11 - Изображение со сгенерированной маркировкой

Готовый набор данных будет разделен на тренировочную выборку и тестовую выборку. Часть изображений из тестовой выборки будет предобработана алгоритмами, описанными в пункте 2.2 [6].

В процессе работы над нейронной сетью для распознавания цифр на маркировках, помимо набора данных изображений черных металлов с нанесенными на них маркировками, необходимо также создать 2 файла в формате csv: это файл классов и файл аннотаций. Количество классов в данном случае – 10 (цифры от 0 до 9). Файл аннотаций содержит в себе следующие данные по каждой цифре в маркировке:

- название изображения;
- координаты цифры (координаты верхней левой и нижней правой точек);
- принадлежность к классу.

В дальнейшем будет произведено исследование разработанной системы, в котором будет произведено сравнение качества распознавания в случаях, если на вход поступают предобработанные и непредобработанные изображения [1].

2.2 Подготовка набора данных предобработанных изображений

В числе изображений черных металлов в сгенерированном наборе данных имеются такие, на которых маркировка тяжело читается человеческим глазом. Это происходит из-за влияния внешних факторов – наличия примесей в металле, шумов на изображении и световых бликов [5].

Алгоритм предобработки изображений состоит из следующих частей:

- повышение контраста изображения;
- удаление световых бликов;
- удаление шума.

Блок-схема данного алгоритма представлена на рисунке 12.

					МИВУ 01.03.02-00.000 ПЗ	Лист
						19
Из	Лист	№ докум.	Подп.	Дата		

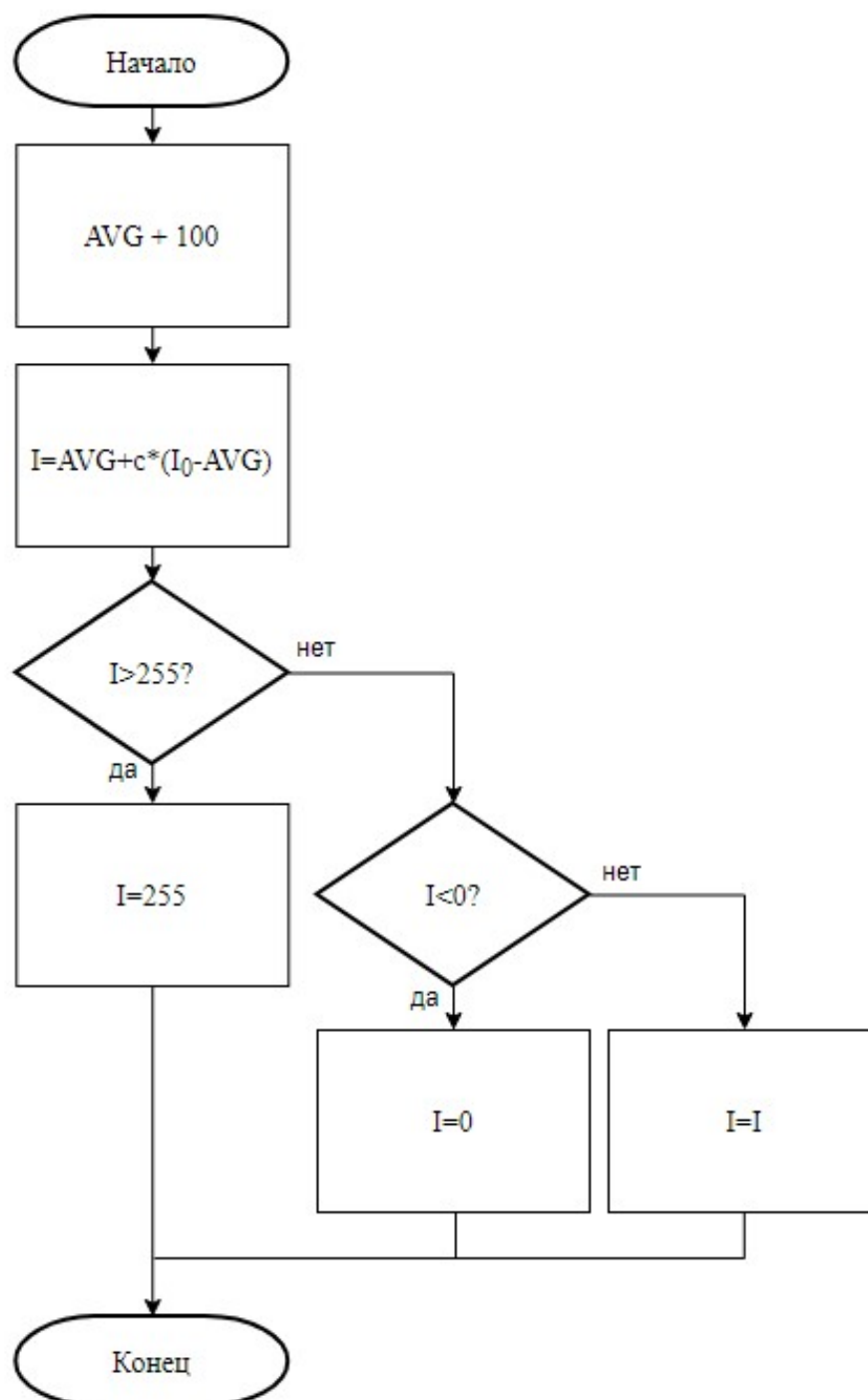


Рисунок 12 - Блок-схема алгоритма повышения контраста изображения

Повышение контраста изображения будет производиться следующим образом. На первом шаге нужно будет найти среднее арифметическое всего количества пикселей на черно-белом изображении и прибавить к ним 100, так как это пороговое значение позволит более качественно отделить «полезные» пиксели от фоновых [6].

Затем нужно рассчитать новые значения яркостей пикселей по формуле:

$$I = AVG + c \cdot (I_0 - AVG),$$

где AVG – это значение, полученное на предыдущем шаге, c – это коэффициент контраста (по умолчанию устанавливается равным 2), I_0 – текущая яркость пикселя.

Если яркость пикселя выше 255, его значение нужно поставить равным 255. Если яркость пикселя получилась меньше 0, то значение его яркости устанавливается равным 0. Иначе яркость устанавливается равной вычисленному значению.

Световые блики занимают довольно большую часть изображения. В случаях, когда они своим расположением не пересекаются с маркировкой, их можно легко локализовать и удалить при помощи следующего алгоритма:

- 1). бинаризация изображения [8];
- 2). применение медианной фильтрации с ядром 3 на 3;
- 3). дилатация – это свертка изображения ядром 7 на 7;
- 4). поиск контуров изображения;
- 5). определение площади контуров;

6). установление порогового значения равным 5000 пикселей (если площадь найденных контуров будет больше, чем 5000 пикселей, то часть изображения внутри этого контура будет закрашена фоновым цветом, так как эта область будет распознана как блик);

- 7). заполнение найденной области фоновым цветом [3].

Алгоритм удаления шума состоит из следующих шагов:

- 1). поиск контуров;
- 2). определение площади контуров;
- 3). определение количества найденных контуров - если площадь контура больше 0, то счетчик количества контуров увеличивается на 1;

4). для всех найденных контуров, кроме самого большого по площади (контура маркировки), яркость пикселей площадей устанавливается равной яркости фона.

Таким образом происходит удаление шума и размытых дефектов на изображении (например, мелкого мусора на изображении металла)[5].

Тестовая выборка из предобработанных изображений будет генерироваться при помощи системы, разработанной в рамках курсового проекта по дисциплине

«Проектирование информационных систем», на рисунках 13 и 14 приведены примеры изображений, взятых для проведения данного эксперимента:

Рисунок 13 – Исходное изображение

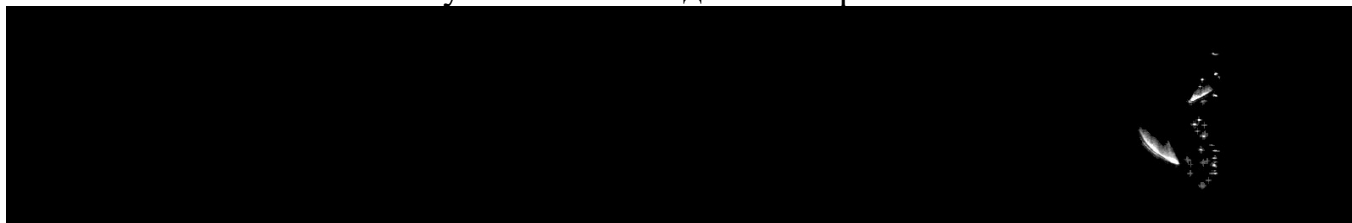


Рисунок 14 – Изображение после предобработки

В результате был сделан вывод о том, что алгоритм предобработки необходимо видоизменить таким образом, чтобы на данном этапе маркировка максимально сохраняла свою целостность.

Путем поэтапного исследования работы блока алгоритмов предобработки изображения, было доказано, что целостность маркировки на изображении теряется в 2-х случаях:

- на этапе удаления мелких областей от примесей в металле;
- на этапе удаления больших областей – засветов на изображении.

Таким образом, было принято следующее решение:

- в алгоритме удаления мелких областей была настроена маска ядра для дилатации – она была уменьшена с 7 на 7 пикселей до 2 на 2 пикселей для первой итерации (очистка изображения от шумов) и с 9 на 9 пикселей до 3 на 3 пикселя для 2-4 итераций (удаление наиболее больших областей, в т.ч. примесей в металле);
- в алгоритме удаления больших областей (световых бликов на изображениях) размер максимальной области был уменьшен с 5000 пикселей до 3000 пикселей (так как изображение из нового набора данных состоит из реальных изображений черных металлов, а не сгенерированы, как это было сделано для системы, реализованной в курсовом проекте)[2].

Блок-схемы алгоритмов удаления шума и световых бликов на изображениях черных металлов представлены на рисунках 15 и 16.

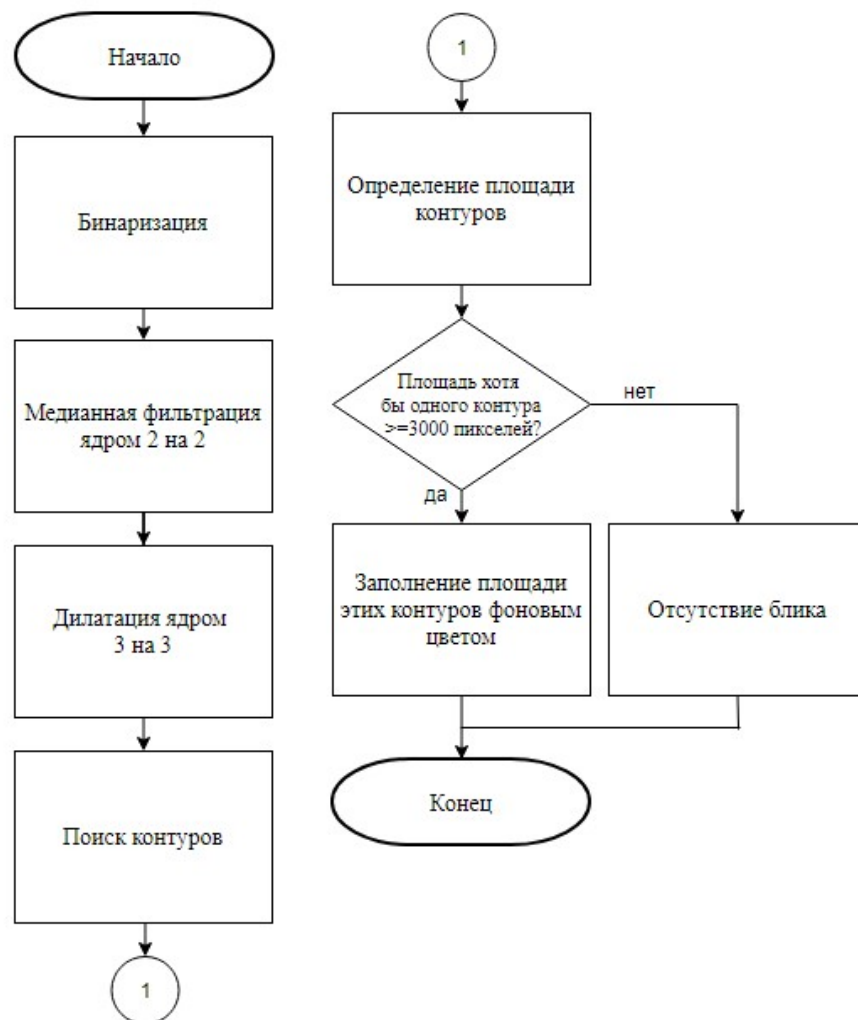


Рисунок 15 - Блок-схема алгоритма удаления световых бликов

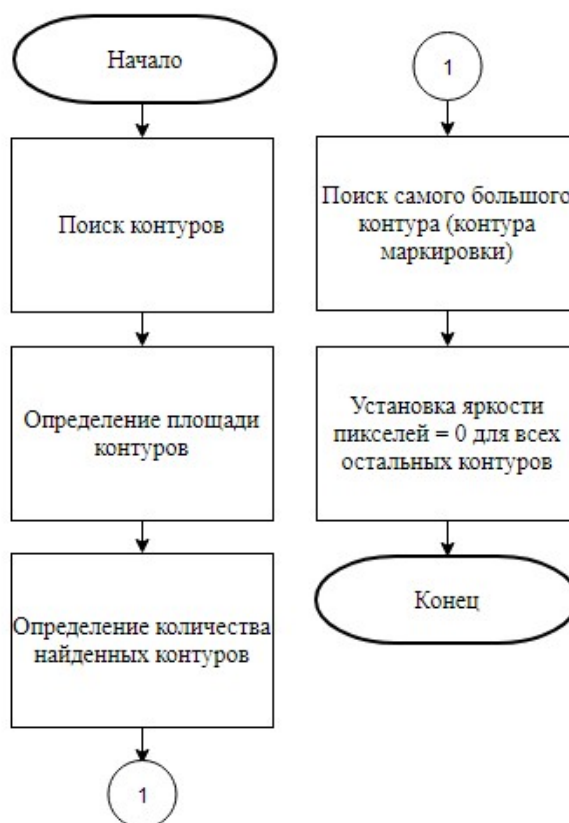


Рисунок 16 - Блок-схема алгоритма удаления шума на изображениях

Пример изображения, полученного в результате работы алгоритмов блока предобработки, представлен на рисунке 17:

Рисунок 17 – Изображение после предобработки

Таким образом, целостность маркировки была сохранена более, чем на 60%.
Следующий этап – разработка системы распознавания маркировки.

3 Краткая характеристика программных средств

3.1 Разработка системы распознавания

По заданию научного руководителя, разработка системы производилась в Google Colab.

Согласно официальной документации библиотеки машинного обучения TensorFlow для решения задач класса object detection подходят следующие модели свёрточных нейронных сетей:

- RetinaNet;
- Mask R-CNN;
- ShapeMask;
- SpineNet [1].

В работе была использована модель RetinaNet (рис. 18) [1].

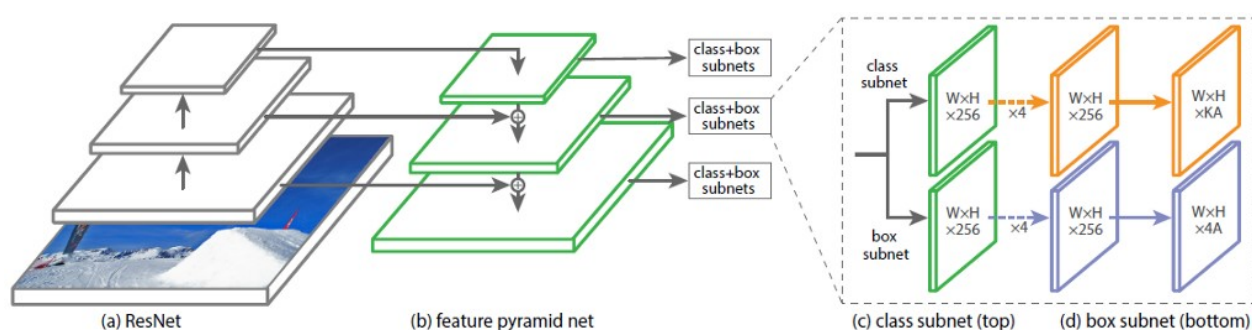


Рисунок 18 - Схема модели RetinaNet

Для обучения использовался предобученный граф весов на популярном наборе данных COCO (применяется при создании графовых нейросетей)[4]. Схема обучения представлена на рисунке 19:

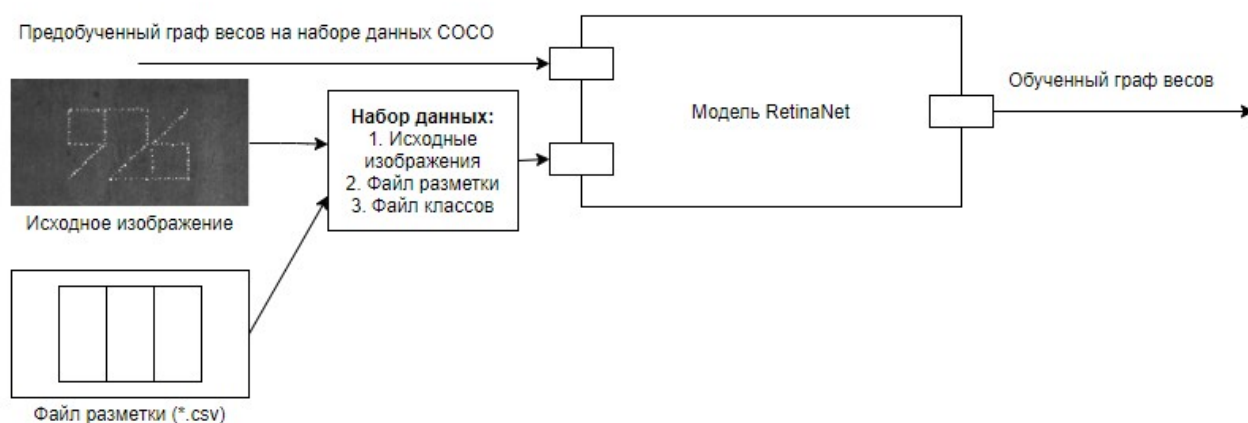


Рисунок 19 - Схема обучения нейросети

На данной схеме видно, что для обучения используется следующий набор исходных данных:

- набор изображений со сгенерированной маркировкой (обучающая выборка);
- файл разметки (или файл аннотаций, в формате csv) – это файл, который содержит в себе информацию по каждому символу обучающей выборки: название изображения, координаты расположения верхней левой и нижней правой точек цифры, а также класс, к которому принадлежит цифра (от 0 до 9);
- файл классов (в формате csv) – это файл, содержащий информацию о количестве классов (от 0 до 9).

После обучения графа весов СОСО вышеперечисленным набором исходных данных, на выходе получается обученный граф весов (модель нейросети, в формате h5).

Эмпирическим путем были выбраны следующие параметры обучения:

- 50 эпох,
- 500 итераций на эпоху,
- размер пакета данных – 10 изображений.

На рисунке 20 изображен график изменения параметров обучения нейросети, из которого следует, что выбранные параметры обучения являются оптимальными в данном случае.

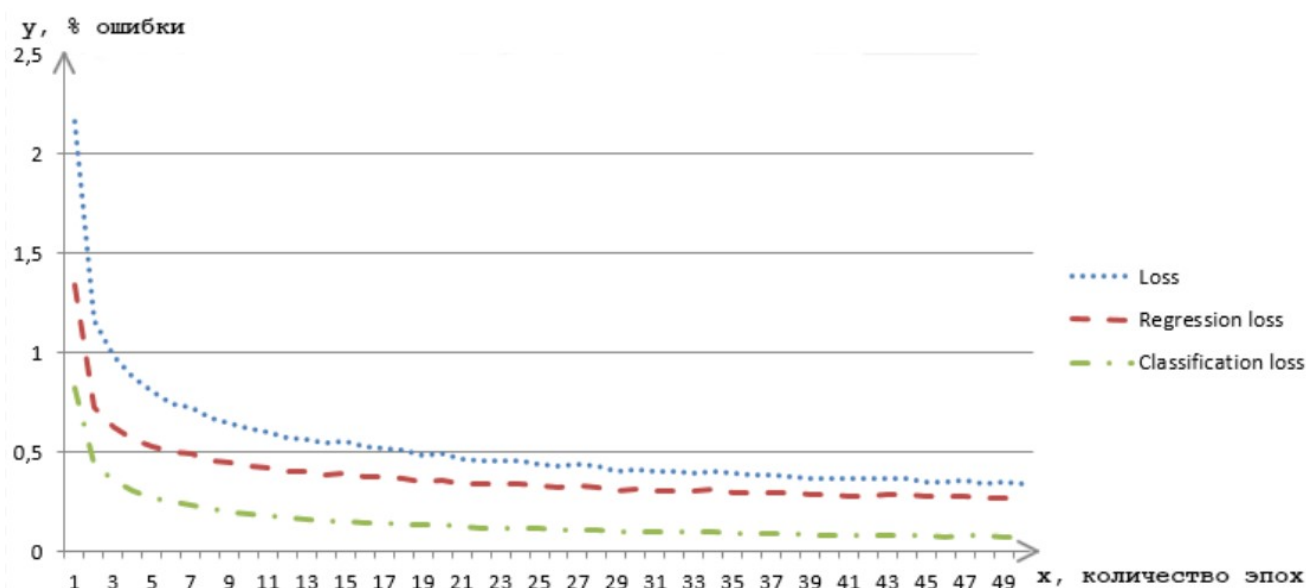


Рисунок 20 - График изменения параметров обучения

В результате ошибка классификации (classification loss) составила 0,0756.

Общая схема использования нейронной сети представлена на рисунке 21.

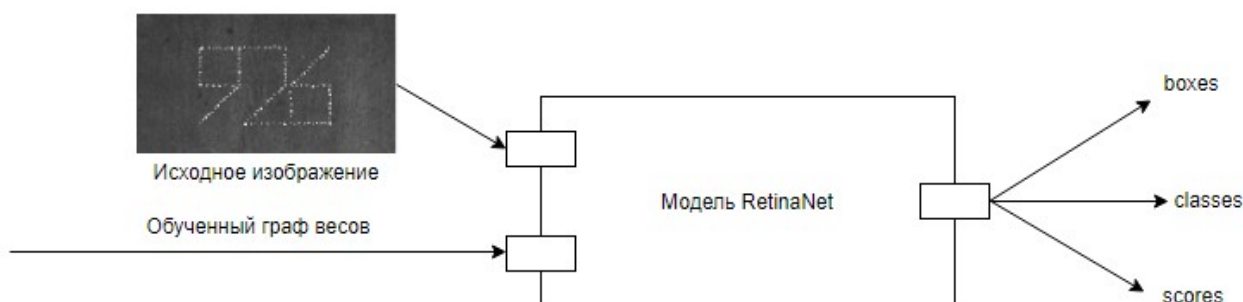


Рисунок 21 – Схема использования нейронной сети

Исходя из схемы видно, что на вход системе подаются изображение со сгенерированной маркировкой и обученный на предыдущем этапе граф весов. После обработки нейросетью, она выдает следующие данные: boxes, classes и scores. Boxes – это данные, которые содержат в себе координаты распознанной цифры в маркировке. Classes – это данные, которые содержат в себе класс, к которому принадлежит распознанная цифра (от 0 до 9). Показатель scores отражает вероятность распознавания цифры.

3.2 Тестирование разработанной системы

В конечном итоге, было произведено тестирование разработанной системы.

Из каждого набора данных было выбрано по 30 изображений. Наглядно примеры нескольких обработанных нейросетью изображений приведены на рисунках 22, 23 и 24.



Рисунок 22 - Пример изображения с плохо распознанной маркировкой



Рисунок 23 - Пример изображения частично распознанной маркировкой

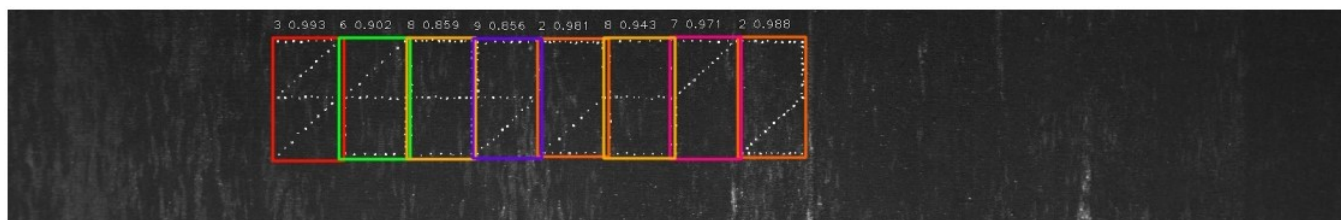


Рисунок 24 - Пример изображения с верно распознанной маркировкой

Результаты тестирования работы системы на непредобработанных изображениях представлены в таблице 1.

Таблица 1 – Результаты тестирования (непредобработанные изображения)

№ п/п	Название файла	Число на маркировке на изображении	Распознанное число на маркировке	Процент верно распознанных цифр
1	0000f269f.jpg	605880	605880	100%
2	000a4bcdd.jpg	827	827	100%
3	000f6bf48.jpg	08397935	08397935	100%
4	00a0b7730.jpg	29844921	2984421	100%
5	00a0f9949.jpg	14051618	14051618	100%
6	00ac8372f.jpg	118	118	100%
7	00af2671f.jpg	56016	56016	100%
8	00b7fb703.jpg	20006	2000-	80%
9	00b989e78.jpg	569	569	100%
10	00bbcd9af.jpg	759	759	100%
11	00bc01bfe.jpg	5416288	5416288	100%
12	00bf8497a.jpg	9267	9267	100%
13	00c24a74c.jpg	99679	999*679	83,33%
14	00c6060db.jpg	10683	10683	100%
15	00cdb56a0.jpg	938057	938057	100%
16	00d7ae946.jpg	7639881	73*39881	83,33%
17	00d72c660.jpg	691860	691860	100%
18	00dcb19f3.jpg	1689	1-89	75%
19	00ded7837.jpg	513834	513834	100%
20	00df5b714.jpg	68050	68050	100%
21	00e0398ad.jpg	47826388	4---4*388	50%
22	00ec97699.jpg	34710	34710	100%
23	00f3799a7.jpg	51844	51844	100%
24	00fe04c98.jpg	587	587	100%
25	00fe7f023.jpg	40734793	407347-3	87,5%
26	0a1cade03.jpg	684	684	100%
27	0a2dbbb6f.jpg	31311	31311	100%
28	0a3bbea4d.jpg	30302	30302	100%
29	0a4ad45a5.jpg	26746069	2674606-	87,5%
30	0a4ef8ee7.jpg	394671	35*4671	83,33%

Таким образом, общий процент распознавания цифр составил около 94,33%.

Следующий этап – тестирование работы системы с использованием предобработанных изображений. Для достоверности эксперимента решено было взять те же изображения, что использовались и в предыдущем тестировании (из набора данных предобработанных изображений).

Примеры предобработанных изображений с распознанными маркировками представлены на рисунках 25, 26 и 27.

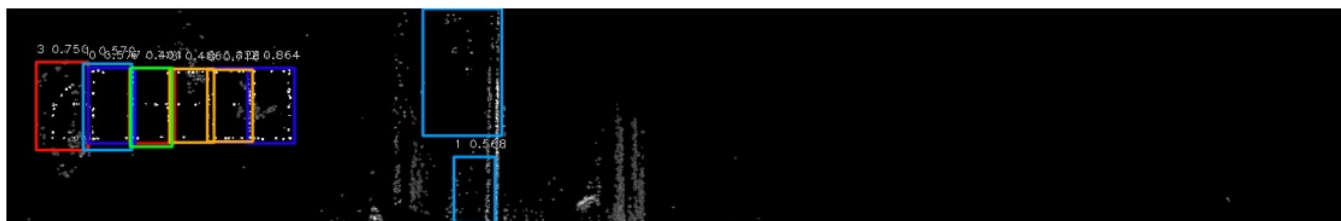


Рисунок 25 - Пример изображения с плохо распознанной маркировкой

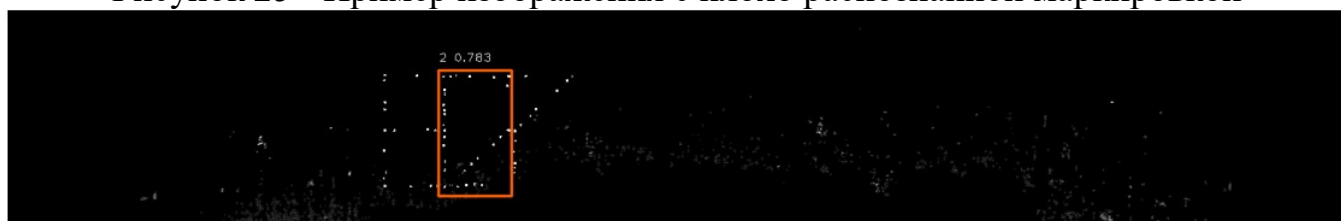


Рисунок 26 - Пример изображения с частично распознанной маркировкой



Рисунок 27 - Пример изображения с верно распознанной маркировкой

Как видно, предложенные алгоритмы процесса предобработки частично справились со своей работой. Изображения стали более контрастными, некоторая часть дефектов была бинаризована закрашена фоновым цветом, согласно алгоритму удаления световых бликов на изображениях. Но, как как за основу брались настоящие изображения черных металлов, которые имеют неоднородную текстуру, достичь качественного результата распознавания не удалось.

В таблице 2 представлены результаты тестирования работы разработанной системы на предобработанных изображениях.

Общий процент распознавания цифр на маркировках в случае с предобработанными изображениями составил 26%.

Таблица 2 – Результаты тестирования (предобработанные изображения)

№ п/п	Название файла	Число на маркировке на изображении	Распознанное число на маркировке	Процент верно распознанных цифр
1	0000f269f.jpg	605880	3*-4*-80*0	33,33%
2	000a4bcdd.jpg	827	-2-	33,33%
3	000f6bf48.jpg	08397935	0---76*35	50%
4	00a0b7730.jpg	29844921	-----	0%
5	00a0f9949.jpg	14051618	14---61-	50%
6	00ac8372f.jpg	118	1-8	66,67%
7	00af2671f.jpg	56016	-----	0%
8	00b7fb703.jpg	20006	--0--	20%
9	00b989e78.jpg	569	-1*1*	0%
10	00bbcd9af.jpg	759	7--	33,33%
11	00bc01bfe.jpg	5416288	-4-----	14,28%
12	00bf8497a.jpg	9267	----	0%
13	00c24a74c.jpg	99679	3*--7-	20%
14	00c6060db.jpg	10683	10--3	60%
15	00cdb56a0.jpg	938057	-----	0%
16	00d7ae946.jpg	7639881	7-36*8-1	57,14%
17	00d72c660.jpg	691860	--18-0	50%
18	00dcb19f3.jpg	1689	-1*--	0%
19	00ded7837.jpg	513834	--3-3-	33,33%
20	00df5b714.jpg	68050	----0	20%
21	00e0398ad.jpg	47826388	-----	0%
22	00ec97699.jpg	34710	----2*	0%
23	00f3799a7.jpg	51844	5----	20%
24	00fe04c98.jpg	587	55*7	66,67%
25	00fe7f023.jpg	40734793	1*0--4---	25%
26	0a1cade03.jpg	684	---	0%
27	0a2dbbb6f.jpg	31311	--3--	20%
28	0a3bbea4d.jpg	30302	--30-	40%
29	0a4ad45a5.jpg	26746069	-----	0%
30	0a4ef8ee7.jpg	394671	--4671	66,67%

Ошибки, полученные в результате распознавания, можно разделить на 2 вида: ошибки 1-го рода и ошибки 2-го рода. К ошибкам 1-го рода относятся случаи, когда цифра не была распознана (в таблицах выше нераспознанная цифра отмечена знаком «-»). Ошибки 2-го рода – это случаи, когда цифра была неверно распознана нейросетью (в таблицах выше неверно распознанные цифры отмечены символом «*»). Количество выявленных ошибок по каждому виду изображений представлено в таблице 3.

Таблица 3 – Ошибки, выявленные при распознавании

	Непредобработанные изображения	Предобработанные изображения
Количество ошибок 1-го рода	7	108
Количество ошибок 2-го рода	4	12
Общее количество ошибок	11	120

Исходя из данных, полученных в результате тестирования разработанной системы, можно сделать следующий вывод: разработанная система на 68,33% лучше работает с непредобработанными изображениями. Таким образом, при использовании разработанной системы для распознавания точечной маркировки в практических целях, предобработка изображений не имеет смысла, так как ее использование отрицательно влияет на качество распознавания [8].

Заключение

Таким образом, в рамках работы над выпускной квалификационной работой была разработана и протестирована система распознавания точечной маркировки на цифровых изображениях черных металлов.

В процессе работы над созданием системы были выполнены следующие поставленные задачи:

- был проведен анализ предметной области, проанализированы аналогичные алгоритмы распознавания, предъявлены требования к разрабатываемой системе;
- проанализированы и описаны алгоритмы, используемые при разработке системы;
- сгенерирован набор данных для обучения нейронной сети;
- был создан набор данных с предобработанными изображениями для тестирования эффективности работы системы;
- была реализована система распознавания;
- была протестирована разработанная система, а также сделаны следующие выводы о ее работе с непредобработанными и предобработанными изображениями:
 - а). из 30-ти непредобработанных изображений верно было распознано 94,33% цифр;
 - б). из 30-ти предобработанных изображений верно было распознано 26% цифр;
 - в). в практической деятельности лучшим решением будет использовать изображения черных металлов с маркировкой, которые не прошли предобработку алгоритмами по удалению дефектов изображения.

Задачи, поставленные на выпускную квалификационную работу, можно считать выполненными в полном объеме.

Список литературы

1. Object Detection with RetinaNet [электронный ресурс]. Режим доступа: <https://keras.io/examples/vision/retinanet/> (дата обращения: 08.05.2021).
2. Severstal: Steel Defect Detection [электронный ресурс]. Режим доступа: <https://www.kaggle.com/c/severstal-steel-defect-detection> (дата обращения: 08.05.2021).
3. TensorFlow 2 Object Detection API tutorial [электронный ресурс]. Режим доступа: <https://tensorflow-object-detection-api-tutorial.readthedocs.io/en/latest/> (дата обращения: 08.05.2021).
4. Архитектура нейронной сети RetinaNet [электронный ресурс]. Режим доступа: <https://habr.com/ru/post/510560/> (дата обращения: 08.05.2021).
5. Гонсалес, Р., Вудс, Р. Цифровая обработка изображений: монография. - Москва: Техносфера, 2005. - 1072 с.
6. Грузман, И.С. Цифровая обработка изображений в информационных системах: учеб. пособие. – Новосибирск: Изд-во НГТУ, 2000. – 168 с.
7. Обнаружение объекта [электронный ресурс]. Режим доступа: https://www.tensorflow.org/hub/tutorials/object_detection (дата обращения: 08.05.2021).
8. Приоров, А.Л. Цифровая обработка изображений: учеб. пособие / Ярославский государственный университет. – Ярославль: ЯрГУ, 2007. – 235 с.
9. Фисенко, В.Т., Фисенко, Т.Ю. Компьютерная обработка и распознавание изображений: учеб. пособие. – СПб.: СПбГУ ИТМО, 2008. – 192 с.