

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего
образования
«Тольяттинский государственный университет»

Институт машиностроения

(наименование института полностью)

Кафедра «Оборудование и технологии машиностроительного производства»

(наименование)

15.03.05 Конструкторско-технологическое обеспечение
машиностроительных производств

(код и наименование направления подготовки)

Технология машиностроения

(направленность (профиль))

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)**

на тему Разработка системы дефектоскопии и дефектометрии заготовок
при помощи системы машинного зрения

Обучающийся

Р.Р. Журавлев

(Инициалы Фамилия)

(личная подпись)

Руководитель

к.т.н., доцент А.В. Зотов

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Консультанты

к.э.н., доцент О.М. Сярова

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

доцент М.А. Кривова

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2025

Аннотация

Разработка системы дефектоскопии и дефектометрии при помощи системы машинного зрения. Выпускная квалификационная работа (ВКР). Тольятти. Тольяттинский государственный университет, 2025 год.

В бакалаврской работе представлена разработка контрольного стенда для дефектоскопии и дефектометрии детали «Головка блока цилиндров» с использованием систем машинного зрения и обучения в условиях массового производства.

Ключевые слова: деталь, заготовка, контроль, машинное зрение, код, нейронные сети, стенд контрольный, безопасность и экологичность проекта, экономическая эффективность.

Работа состоит из введения, семи разделов, заключения и списка используемых источников.

В первом разделе проводится полный сбор и анализ имеющихся на предприятии «АвтоВАЗ» данных об контролируемой заготовке: способ получения, конструктивные и технические параметры и возникающие на ней дефекты.

Во втором разделе выбирается и изучается представленный вид использующейся системы машинного зрения и требования к работе аппаратуры для неё, такие как процесс сбора и обработки изображения.

В третьем разделе проводятся результаты серии экспериментов по непосредственному контролю дефектов на представленной заготовке. Это позволит изучить все протекающие при этом процессы обработки и фильтрации данных.

В следующем разделе систематизирован и проанализирован алгоритм работы системы машинного обучения. Использование данных нейронных сетей составляет важнейшую часть работы, так как они, благодаря алгоритмам обучения по вводным данным, почти полностью позволяют определить количество и расположение дефектов заготовки.

В пятой главе полученные знания систематизированы и использованы в разработке и организации рабочего места контрольного стенда. Здесь разработана его конструкция вместе с габаритами, количеством используемых камер и другими характеристиками.

Шестая посвящена определению безопасности и экологичности разрабатываемой системы.

Последняя – приводятся расчёты экономической эффективности производства контрольного оборудования.

Заключение работы содержит выводы об окончании процесса и полученным в ходе эксперимента результатам.

Бакалаврская работа содержит пояснительную записку в размере 117 страниц, содержащую 14 таблиц, 20 рисунков, и графическую часть, содержащую 7 листов.

Abstract

The title of the graduation work is «Development of a system for detection and flaw sizing using a machine vision system». The graduation work consists of an introduction, 7 parts, 26 figures, 14 tables, a conclusion, and a list of 25 references including foreign sources. The aim of this graduation work is to automate the applied control process and reduce the influence of the human factor on it.

The object of the graduation work is system of detection and flaw sizing of work pieces at the factory. The subject of the graduation work is design of a control and measuring stand for detection and flaw sizing of a cast work pieces "Cylinder Head" for the enterprise JSC "AvtoVAZ" with the implementation of machine vision and machine learning systems. The key issue of the graduation work improves the quality of manufactured products and the competitiveness of cars on the markets. The graduation work may be divided into several logically connected parts which are literature review, results and their discussion and experimental part.

The first part describes a complete collection and analysis of the data available at the AvtoVAZ on the controlled work piece. The second part outlines the results the type of the used machine vision system and the requirements for the operation of the equipment for it. The third part consists of the results of a series of experiments on defects control on the presented work piece. The next part develops the algorithm of the machine learning system. The use of neural network data is the most important part of the work, since based on it, the number and location of defects in the work piece are completely determined. In the fifth part, it is used in the organization of the workplace of the control stand. The sixth chapter is devoted to the safety and environmental friendliness of the system. The last one provides calculations of the economic efficiency.

In conclusion we'd like to stress contains conclusions on the relevant analyzed issues. Nevertheless, more experimental data are required. The work is of interest for wide circle of readers interested in modern technologies in the field of mechanical engineering.

Содержание

Введение.....	7
1 Анализ исходных данных детали	9
1.1 Описание конструкции и технических параметров заготовки.....	9
1.2 Возможные дефекты заготовки	13
1.3 Контроль качества отливок.....	17
2 Анализ системы машинного зрения.....	20
2.1 Виды неразрушающего контроля.....	20
2.2 Технические характеристики применяемого оборудования	22
2.3 Обработка полученных изображений	24
2.4 Обработка и выявление дефектов для обучения	29
3 Проведение экспериментов.....	30
3.1 Общие сведения	30
3.2 Подготовка образца	31
3.3 Организация рабочего места экспериментатора.....	32
4 Внедрение системы машинного обучения	34
4.1 Общие сведения об нейронных сетях	34
4.2 Свёрточные нейронные сети.....	35
4.4 Разработка машинного кода.....	39
4.5 Использование пятиклассового метода	42
5 Организация рабочего места проведения контрольной операции.....	45
5.1 Разработка стенда контрольного	45
5.2 Базирование заготовки на рабочем месте.....	48
6 Безопасность и экологичность технического объекта	49
7 Экономическая эффективность работы	55
7.1 Капитальные вложения, трудоёмкость и длительность работ	55
7.2 Построение сетевого графика.....	58
7.3 Расчёт фонда заработной платы и суммарных затрат.....	61
Заключение	63
Список используемых источников.....	65
Приложение А Участок производственного корпуса 21	68
Приложение Б Калибровка.....	69

Приложение В Результаты машинного обучения 2-х классового исходного кода	72
Приложение Г Результаты машинного обучения 2-х классового кода на предобученной нейронной сети.....	77
Приложение Д Результаты работы машинного кода 5-классового	82
Приложение Е Спецификация	91

Введение

Производственный процесс автомобиля на любом машиностроительном предприятии включает в себя сборку нескольких тысяч разнообразных деталей в готовую продукцию.

На предприятии АО «АвтоВАЗ» можно произвести различные металлические детали автомобиля, для этого необходимо произвести заготовку путём штамповки или литья, провести механическую обработку с целью получения определённых конструктивных характеристик.

Также немаловажную роль играет контроль изделия на всех станциях производства - контроль перед и после механической обработки. Например, крупногабаритное и сложнопрофильное изделие с большой площадью поверхности и множеством отверстий в корпусе имеет достаточно продолжительный этап контроля.

Кроме того, применяемый на сегодняшний день визуальный осмотр не идеален, потому что он не позволяет исключать бракованные изделия. Для решения данной проблемы разрабатывается система дефектоскопии и дефектометрии на основе оптического метода контроля деталей с использованием системы машинного зрения.

Актуальность данной работы состоит в разработке контрольного стенда с использованием современных технологий в сфере аппаратного решения и программирования, таких как, машинное зрение и обучение. Это позволит повысить качество выпускаемой продукции и конкурентоспособность автомобилей на российском и международном рынке.

Объектом работы выступает система дефектоскопии и дефектометрии заготовок на предприятии.

Предметом данной работы является конструкция контрольно-измерительного стенда по дефектоскопии и дефектометрии литой заготовки «Головка цилиндров» для предприятия АО «АвтоВАЗ» с внедрением систем машинного зрения и машинного обучения.

Цель работы в ВКР состоит в автоматизации применяющегося контрольного процесса и снижении влияния человеческого фактора на него. Для этого необходимо выполнить в работе следующие задачи:

- проанализировать конструкцию и технические характеристики выбранной детали;
- определить появляющиеся в процессе заготовки дефекты детали, которые нужно найти на данном этапе контроля;
- выбрать и изучить метод неразрушающего контроля и технические параметры используемого при этом инструмента;
- провести серию экспериментов по эффективности разработанного алгоритма работы;
- изучить работу системы машинного обучения, включая, алгоритм работы, требуемые входные данные и так далее;
- на основе полученных данных провести организацию рабочего места для контроля заготовки с помощью системы машинного зрения;
- рассчитать экономическую составляющую работы;
- обеспечить экологическую безопасность системы.

При проведении исследования использовались общенаучные методы познания.

Практическая значимость заключается в автоматизации системы контроля, с помощью этого минимизируется человеческий фактор. Благодаря чему заметно уменьшается процент брака, попадающий на следующие этапы производство. Это позволяет минимизировать риск остановки производства и уменьшить дополнительные расходы.

Научная новизна исследования заключается в том, что наряду с системами машинного зрения, в процесс работы механизма включено использование систем нейронных сетей, сейчас более известных как искусственный интеллект.

1 Анализ исходных данных детали

1.1 Описание конструкции и технических параметров заготовки

В сборке автомобиля принимает участие большое количество самых разнообразных деталей, начиная от различных пластмассовых частей интерьера машины, заканчивая разнообразными металлическими деталями и узлами сборки.

На сегодняшний день наиболее популярными в мире являются автомобили с двигателем внутреннего сгорания. Настоящее «сердце» таких транспортных средств есть непосредственно двигатель, где энергия сгорания топлива превращается в механическую работу.

Автомобиль – определённо очень опасный вид транспорта, любая существенная проблема во время его движения может привести к аварии и жертвам среди населения. Именно поэтому все заинтересованы в том, чтобы не допустить попадание непригодного к работе двигателя. Для чего следует проводить ступенчатый, всеобъемлющий контроль изделий такого типа.

Сам двигатель внутреннего сгорания состоит из множества деталей. Внутри цилиндров происходит процесс сгорания топливно-воздушной смеси, что приводит в движение находящиеся здесь поршни. Они являются подвижной частью кривошипно-шатунного механизма, который нужен для передачи движения дальше на коленчатый вал автомобиля. Основной же неподвижной частью кривошипно-шатунной системы является блок цилиндров и головка блока цилиндров.

Головка блока цилиндров предназначена для запираания цилиндров и образования замкнутой камеры сгорания, герметизируя её с поршнями от окружающей среды. Кроме того, на головку крепятся другие детали как клапаны с пружинами, свечи зажигания, форсунки и некоторые другие. В зависимости от отдельно выбранной модели транспорта, параметры конструкции головок могут варьироваться в больших пределах.

В рассматриваемом случае была выбрана на исследование головка блока цилиндров 11182-1003015-70 для 8-ми клапанного двигателя автомобиля Lada Kalina с предприятия «АвтоВАЗ». Заготовка представлена на рисунке 1. Она является цельнолитой деталью, так как должна выдерживать большие нагрузки, возникающие в процессе работы двигателя.



Рисунок 1 – Заготовка «Головка блока цилиндров 11182-1003015-70»

Изделие относительно крупное, максимальные его габариты составляют: высота – 141,7 мм; ширина – 140,6 мм; длина до базовой поверхности – 415,5 мм. Масса готовой продукции есть 9,82 кг, а заготовки без механической обработки – не более 12,01 кг.

Заготовка получается путём литья алюминиевого сплава АК6М2 в предварительно подготовленный для этого кокиль. Кокиль - металлическая литейная форма для создания отливок, это сложная система куда кроме непосредственно формы входит литниковая система и прибыли. На рисунке 2 можно увидеть пример кокиля, применяемого на производстве АвтоВАЗа. Перед заливкой алюминиевого сплава кокиль нуждается в дополнительной обработке: очистки от старой краски всех формообразующих, окрашивание грунтовой и специальной краской литниковых каналов, прибылей и поверхностей кокиля, нагрев до 280-300 °С, смазка зон трения. Также каждые 2-3 рабочие недели необходимо проводить техническое обслуживание кокиля: демонтаж, мойку, пескоструйную обработку, контроль и сборку, чтобы избежать появления брака и выявить износ оборудования.



Рисунок 2 – Пример подготовленного и окрашенного кокиля

Используемый алюминиевый сплав АК6М2 включает в себя большое количество различных элементов. Основной компонент сплава – алюминий Al, также вводится 5,5-6,5% кремния Si, 1,8-2,3% меди Cu, 0,30-0,45% магния Mg, 0,1-0,2% титана Ti, остальных элементов, включая железо Fe, никеля Ni, цинка Zn и молибдена Mn в сумме не более 0,7%.

Включение такого количества элементов необходимо для придания сплаву определённых свойств. Магний добавляют, чтобы изменить твёрдость отливок после термической обработки. Если процент добавленного магния преуменьшен или преувеличен, то заготовки получаются бракованными.

Титан Ti добавляется в сплав для рафинирования. Рафинирование — это химическая обработка, которая позволяет металлу затвердеть с кристаллической структурой намного мельче, улучшая таким образом металлургические и механические характеристики отливок.

Для изменения кристаллической структуры кремния Si находящегося в алюминиевом сплаве, проводят процесс модификации путём добавления стронция Sr [8, с.16]. Далее сплав подвергается операциям дегазации с помощью обдувки чистым азотом N, и шлакоудалению внесением шлакоудаляющего вещества в виде соли зернообразной формы. Если их не провести качественно, то это приведёт к образованию большого числа газовых

раковин на вспухшей поверхности и включению оксидов алюминия в тело заготовки.

Перед заливкой в полость кокиля устанавливаются специальные стержни, они нужны для образования внутренних плоскостей и отверстий в головке. Такие стержни изготавливаются штамповкой из специального “французского” песка АФА 45 – смеси регенерирующего песка, смолы с добавлением канализатора. Изображения некоторых стержней представлены на рисунке 3. Важно проводить дополнительный контроль за стержнями, так как изменения в пропорциях смеси и появлении дефектов стержней, приводят к литью бракованных заготовок. После окончания операции литья и охлаждения сплава стержни с силой выбиваются специальной машиной.



Рисунок 3 – Формовочные песчаные стержни

Строго обязательно соблюдать температурные требования сплава АК6М2, металл должен быть температурой не менее 755 °С и не более 780 °С [9]. В противном случае заливка получит дефекты либо недолива (в первом случае), либо пережога (во втором).

Таким образом, видно, что на процесс литья влияет большое число самых различных параметров, начиная от температурных диапазонов, состава сплава, процесса очистки, заканчивая качеством песчаных стержней и чистотой формообразующих кокиля. Любое отклонение от норм может привести к появлению соответствующих дефектов, представленных ниже, в следующем подразделе работы.

1.2 Возможные дефекты заготовки

Существует два, поддерживающих друг друга, метода предотвращения попадания бракованных изделий дальше в производство. Первый из них состоит в профилактических работах, ответственные за это люди следят за соблюдением нормативов производства на всех участках, в таком случае минимизируется возможный брак литья. Второй путь заключается в постоянном контроле проходящей продукции, контроль выявляет как случайные дефекты производства, которые сложно заранее предугадать, так и систематические, вызванные несоблюдением профилактики, устареванием оборудования или не идеальностью нынешнего процесса.

В любом случае, контрольные мероприятия не должны допускать проход бракованных заготовок на этап механической обработки, ведь чем раньше определится дефекты, тем быстрее головка направится или на ремонт, в случае исправимого брака, или на переплавку при неисправимом.

Полезно как для первого, так и для второго пути проводить систематизацию и сбор данных за продолжительный период о возникающих дефектах, их причинах и зонах появления. На предприятии АвтоВАЗ такой процесс проходил неоднократно и теперь у нас есть рабочая и корректная их классификация.

Важно помнить, что поверхности головки не равнозначны и имеют серьёзные различия. Во-первых, поверхности могут быть внутренними, располагающимися в полости детали, а могут и внешними, снаружи головки блока цилиндров. Внутренние поверхности гораздо более труднодоступны для контроля, так как находятся в тени и, зачастую, не видимы средствами контроля и даже человеческим глазом.

К обязательным в проверке зонам относятся базовые поверхности головки: опорные базы для обрезки прибыли, опорные базы для дальнейшего контроля детали, базы механической обработке. Также не допускаются

дефекты на маркировке продукции, в камерах сгорания, зонах сальников и некоторых других.

Наиболее распространёнными дефектами литья деталей из алюминиевого сплава являются пережоги, недоливы, утяжки, стержневые заливки и неспаи; деформации и трещины; задиры, риски, зарезы; воздушные и песчаные раковины; сколы; засоры; рыхлота.

Пережоги и спаи являются критическими дефектами, зачастую они вызваны несоблюдением установленной на заводе температуры заливки, иногда кокиля. Если температура алюминия была выше максимальной критической, то это вызывает чрезмерный рост зёрен и обезуглероживание поверхности – пережогу. Если наоборот, температура ниже минимальной, то неправильное затвердевание потоков приводит к неспаю на рисунке 4.

Иногда неправильная температура и скорость распространения металла (которая тоже зависит от температуры) создают условия для появления утяжек - углублений с закругленными краями на поверхности отливки, образовавшегося вследствие усадки металла при затвердевании. Утяжки менее 3 мм сложно поддаются ремонту, более – почти никак.

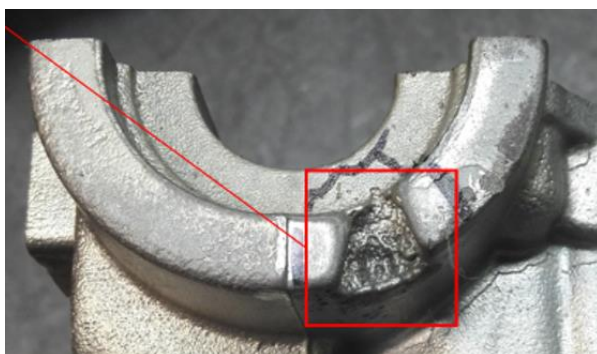


Рисунок 4 – Дефект неспай на плоскости под сальник вала распределительного

Если должным образом кокиль и литниковая система не проходила техническое обслуживание, тогда они могут дать трещину и привести к недоливу – вытеканию металла. Остатки шлака, старой краски, песка и частиц

алюминия накапливаются в кокиле или литнике, образуя в следующей заготовке засоры как на рисунке 5.

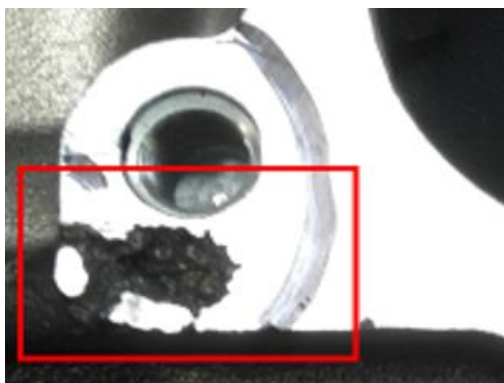


Рисунок 5 – Дефект засора на фланце впускных и выпускных газовых каналов

Тот же самый мусор может затруднять выведение лишних газов при помощи фильтрации из полости заполненной жидким металлом в форме. Это приводит к образованию воздушных раковин на поверхности (рисунок 6). На появление этих раковин влияет и дефекты смеси формовочных стержней, например, при высоком содержании как катализатора, так и смолы.



Рисунок 6 – Дефект воздушных раковин в камере сгорания

Вообще, недостаток смолы в смеси приводит к хрупкости стержней, в процессе литья они могут частично разрушиться, образовав при этом и стержневые заливы- образование металлических выступов в полостях разрушенных стержней, и песчаные раковины – пустоты в теле отливки,

заполненные формовочной смесью. Большое количество газа приводит к рыхлости поверхности, то есть появлению множества воздушных пузырей на вспученной поверхности.

В противоположность, повышенный процент смолы придаёт стержням излишнюю твердость, из-за чего могут возникнуть проблемы их выбивки, а значит к излишним деформациям и трещинам.

Деформация и трещины одни из наиболее распространённых дефектов у которых множество первопричин, начиная от проблем в структуре металла, заканчивая механическим воздействием внешних сил.

При выбивании формовочных стержней сразу же должны выполняться операции зачистки заготовки от остатков литниковой системы и прибылей, заусенцев [13]. Несоблюдение данного регламента не допускается и является жёстко наказуемым.

Базовые поверхности под механическую обработку не должны иметь раковин, заусенцев, следов от разъёма форм, стержней и выталкивателей. На остальные необработанные поверхности они, вместе со слоем оставшейся краски, смазки или трещин, не должны быть более 0,5 мм в общем случае. Также на базовых поверхностях не разрешены уступы, наросты, заусенцы. При этом, следы от масляных разводов и окисей на этом этапе контроля не возбраняются.

На необрабатываемых поверхностях отливок, отлитых в кокиль, допускаются единичные раковины, шлаковые включения, засоры и другие дефекты, при размерах диаметра не более 3 мм и глубины не более 1/3 толщины стенки, и не более 1,5 мм при условии, что с противоположной стороны в этом же сечении нет подобных дефектов.

В целом, можно сделать вывод, что количество контролируемых параметров для одной лишь головки цилиндров измеряется десятками. Для получения требуемой заготовки нужна слаженная работа соответствующих дирекций и отделов. Отступ от какого-либо нормативного документа или остановка производства ведут к образованию негодных партий.

1.3 Контроль качества отливок

До, во время литейного процесса и после него все участвующие компоненты строго обязательно проходят контроль. Контроль проходят в производственном цеху на контрольном стенде и, иногда, выборочно в лаборатории.

В лаборатории проходит проверку, перед использованием, поставленный на предприятие песок. Его проверяют на зернистость просеиванием и кислотность. Также стержни проверяются на наличие трещин и разрушений.

Контроль химического состава металла необходимо осуществлять в каждой раздаточной печи на наличие в сплаве излишнего газообразования или неверных пропорций. Так как проблема касается металлургии отливки, то необходимо сразу же остановить производство, партия отправляется на либо переплав, либо на проверку точности параметров (температура охлаждения, состава, выделение шлаков и так далее) металла или температуры кокиля.

Для того, чтобы понять условия контроля необходимо посмотреть плана производственного цеха, где проходит это мероприятие. Правда, в виду громоздкости этого генерального плана, нам лучше и достаточно изучить участок производственной площади в Приложении А.

Рассматриваемая площадь огорожена от основного металлургического производства стенкой в целях безопасности и организации труда. Важные для нас участки в рамках контроля изделия головка блока цилиндров выделены с помощью цветных прямоугольников. Остальные части непосредственно к рассматриваемой работе не имеют отношения, хотя немаловажны для работы полного процесса изготовления.

Зелёным выделена главная для нас часть – контрольные столы, где происходит непосредственный визуальный контроль заготовок. Как видно занимаются этим около 14 специальных рабочих-контролёров (бывает и меньше). В производстве все заготовки проходят контроль на специальном

контрольном стенде (рисунок 7). На нём находится основной этап визуального контроля поверхности изделия контролёром. В связи с наблюдаемым дефицитом кадров и большой их текучестью контролёры зачастую не обладают обширным опытом работы в данной сфере.

Красным показан метрозал со специальным отдельным контрольным оборудованием – машиной «ДЕРБИ». Она предназначена на АвтоВАЗе для автоматической разметки отливок. Она является инструментом, который проверяет в 3-х плоскостях определенные точки. Машина связана с компьютером посредством программирования, и автоматически читает размеры определенных пунктов и затем создает отчет контроля. В зависимости от уровня важности, принимается решение доработать партию или её исключить. Во всяком случае партия всегда находится отдельно от других с биркой БРАК.

Если заготовки имеет неисправимый брак, то она уходит на переплавку, а если исправимый – то на доработку, эта часть выделена фиолетовым. Здесь другие рабочие с помощью машин занимаются исправлением дефектов заготовок: сваркой, очисткой заусенцев и так далее.

В конечном этапе прошедшие или исправленные заготовки оказываются в розовой зоне, где очищаются, упаковываются и складываются для дальнейшей транспортировки в зону сборки двигателя автомобиля.

Важным способом проверки скрытых дефектов в материале отлитой заготовки головки цилиндров является рентгеноконтроль, в виду трудоёмкости и опасности для человека такой контроль проводится только в специализированных лабораториях вдали от основных производственных помещений с соблюдением строгих мер безопасности. В данном случае он осуществляется только на сечении головки блока цилиндров. Возможно смотреть на видео и затем регистрировать на ленту внутреннюю структуру детали, таким образом легко определяются, трещины или пористость заготовки [23].

Ранее на заводе также осуществляли анализ проникающей жидкостью. Для этого после мойки отливка в проникающей жидкости и через пятнадцать минут сушки, пистолетом на поверхность распыляли белую пыль. В случае, когда через семь минут появлялись красные пятна, отливка признавалась дефектной. Но уже как более двух лет, в связи с наложенными санкциями, метод не проводился, а оборудование находится на консервации.



Рисунок 7 – Рабочее место контролёра

Из всего вышеперечисленного видно, что на данный момент не существует единого и адекватного метода дефектоскопии и дефектометрии поверхности заготовки головки блока цилиндров. Существующие традиционные методы или крайне специализированы для определённых мест или же не дают полную гарантию выявления всей бракованной продукции.

2 Анализ системы машинного зрения

2.1 Виды неразрушающего контроля

Наряду с традиционными методами разрушающего контроля, например, краш-тесты, в машиностроении всё чаще применяют так называемые неразрушающие виды контроля. Основным их отличием (что также является их преимуществом) является проведением исследований на поверхности и в глубине материала детали без их разрушения.

Условно все методы неразрушающего контроля можно разделить на три крупных группы: методы, основанные на электромагнитном взаимодействии; методы, основанные на изучении отражённых волн; и изучении поверхности контролируемого объекта [5].

К первой группе можно отнести магнитный, электрический и вихревой виды контроля. Принцип работы магнитного заключается в изучении взаимодействия ферромагнитных материалов непосредственно в магнитном поле. Электрический метод – взаимодействие в электрическом поле. В вихретоковом происходит регистрация изменения электромагнитных полей, возникающих в теле заготовки, вихревых токов.

С некоторыми материалами можно позволить использовать пьезоэлектрический эффект – изменение форм и размеров контролируемого объекта при проведении через него электрического тока и наоборот.

Вторая группа включает обширный перечень рассматриваемых методов. Каждый метод имеет в своей основе принцип изучения возвратившихся после облучения отражённых волн и их рассеивании. В зависимости от вида и частоты этих волн даны названия каждого из них. Соответственно, для радиоволнового – электромагнитные волны с частотой от 30 кГц до 3 ТГц; акустического (ультразвукового) - звуковые волны выше 20000 Гц; радиографический – рентгеновские волны; виброакустический – определёнными вибрациями и тому подобное.

Эти методы позволяют провести соответствующих контроль на наличие дефектов детали не только на её поверхности, но и в самой структуре, также позволяя заглянуть в закрытые полости, глухие отверстия и сложные фасонные поверхности. Используемое оборудование может являться опасным для человека, поэтому рабочий персонал должен быть специально обучен, подготовлен и защищён от излучения.

Как было написано выше в работе, на предприятии АвтоВАЗ в отдельных случаях и зонах головки цилиндров применяют метод рентгеном. С его помощью контролируют отверстия только в сечении цилиндров.

Методы контроля поверхности изделия обладают отличными физическими принципами работы. Наиболее распространённым видом неразрушающего контроля является визуальный контроль. Он позволяет быстро и дёшево проводить контролирование заготовок без специального оборудования на всех этапах производственного процесса. Главным недостатком является влияние человеческого фактора и возможности человеческого зрения на качество конечного результата.

Иногда для нахождения дефектов на поверхности, её покрывают или промывают специальными растворами – пенетрантами. Метод даёт яркий и видимый эффект для контролёра, на АвтоВАЗе от него отказались в виду дороговизны составляющих реагентов.

На данный момент оптический метод является наиболее идеальным. Он относительно недорогой в применении обслуживании, позволяет интегрировать его со множеством дополнительных программ, что снимает ограничение человеческого глаза [6]. Метод безопасный и распространённый, но требует также познаний в основах работы его оборудования.

Подводя итог, изучив некоторые из методов нарушающего контроля, были найдены их слабые и сильные стороны. Далее выбран наиболее подходящий к заданию метод, что позволит автоматизировать современные системы дефектоскопии и дефектометрии заготовок.

2.2 Технические характеристики применяемого оборудования

При работе с системой машинного зрения основным критерием работоспособности алгоритма является правильный выбор соответствующего оборудования. В оптическом методе контроля главный фиксирующий элемент – датчики, представленный камерами. В зависимости от поставленной цели и планируемого бюджета закупок камеры могут быть линейными или матричными [12, с.22].

В данном случае выбор пал на набор матричных камер от компании BASLER. Они были выбраны, так как на сегодняшний день BASLER занимаются производством высококачественных камер и компонентов для систем машинного зрения в различных областях, включая автоматизацию производства. Они предлагают большой набор самой разнообразной продукции в высокотехнологичной сфере производства и имеют опыт работы с клиентами.

Камеры BASLER обладают хорошими качественными параметрами по относительно доступным ценам. Среди представленных на рынке моделей мы выбрали ace 1440-73gm из серии Basler ace из усовершенствованной линии U. Аппарат можно увидеть на рисунке 8. Сама камера разработана с уникальным набором функций PGI и выпуском новейшего поколения КМОП-сенсоров ознаменовали для серии ace прорыв с точки зрения встроенного программного обеспечения и используемой технологии сенсора.



Рисунок 8 – Камера acA1440-73gm

Среди важных характеристик данной камеры можно отметить, физические короб аппарата имеет габариты 29х29 мм, то есть камера сама по себе достаточно компактная. Средство имеет разрешение по горизонтали и вертикали 1440 px x 1080 px, разрешением 1.6 МР и используемым сенсором IMX273. Камера обладает достаточно высокой частотой кадров по сравнению с обычными, она равняется 73 fps.

Данная модель, в отличие от некоторых из своей серии не поддерживает цветную картинку, поэтому выходящее изображение монохромное, в чёрно-белом цвете. Инструмент поддерживает в работе интерфейс GigE.

Работа с камерой производится в соответствии с представленной на официальном сайте инструкцией. Получаемые на объектив камеры изображение передаётся по единому оптоволоконному кабелю, благодаря GigE с технологией PoE, Camera Link с технологией PoCL и USB 3.0. Само изображение передаётся на экран дополнительного компьютера к специальной плате, на которой установлена соответствующее программное обеспечение. Примерная схема взаимодействия разных средств при использовании полного комплекса систем BASLER представлена на рисунке 9, архитектура не отражает реальных размеров, но даёт понять принцип работы элементов.

Несмотря на имеющиеся у камеры все гибкие возможности ввода-вывода, а также минимальной задержке и дрожании, для требуемой точности получающегося изображения камеру следует располагать на штативе или какой-либо ещё плоской поверхности без дополнительных вибраций [1].

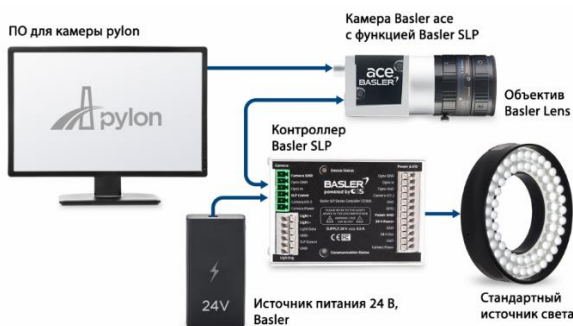


Рисунок 9 – Архитектура системы машинного зрения

На самом деле, освещение является одним из ключевых факторов в удачном процессе фотографирования. Свет должен быть не слишком ярким, чтобы не давать излишний отблеск на металлической поверхности, и не слишком тусклым, чтобы получать адекватные изображения поверхности. Для чего необходимо использовать соответствующие приспособления, виде специальных ламп.

Таким образом, можно отметить, что камера асе 1440-73gm достаточно удобна для решения представленной проблемы, также камера подходит в связи с отличным соотношением её цены и технических характеристик.

2.3 Обработка полученных изображений

При получение любого изображения рассматриваемого изделия с помощью камеры в нём всегда возникают определённые погрешности. Это вызвано тем, что широкоформатная камера приводит к изменению пограничных размеров получаемого изображения [10].

Таким образом, для адекватного определения количества, координат, площади располагаемых дефектов и измерения их глубины при дефектометрии, картина должна проходить дополнительную постобработку. Для чего нужно изучить определённые специфические методы этой обработке.

Во-первых, в самом начале проведения процесса дефектоскопии и дефектометрии контролёр должен выполнить предварительную калибровку камеры [11]. Для массового производства это выполняется в соответствии с составленной инструкцией. В нашей работе в лабораторных условиях первоначальная калибровка может происходить по калибровочной доске (ярким примером служит плоскость обыкновенных шахмат).

Линии калибровочных клеток искажаются до кривых в изображении камеры из-за оптических эффектов любой системы, особенно проявляющих себя с ростом расстояний между объективом камеры и плоскостью детали. Для того, чтобы приблизить их к реальности, вернуть из кривых в прямые,

нужно выполнить калибрование (calibration), введя в программу специальный коэффициент. Этот коэффициент зависит прежде всего от внутренних параметров камеры, включая фокусное расстояние, разрешение крана и другие. Подробно этот процесс будет показан в разделе проведения непосредственного эксперимента [21].

Калибрация может быть проведена в различных программах. На самом деле они являются высокоуровневыми языками программирования и средами для численных вычислений, визуализации и программирования. Они широко используются в научных исследованиях, инженерии и математике благодаря своим мощным инструментам для обработки данных, анализа и моделирования, что идеально подходит для целей научной работы. Программа предлагает обширную библиотеку функций для выполнения математических операций, работы с матрицами, статистики, оптимизации и симуляции динамических систем.

Программа поддерживает создание пользовательских графиков и визуализаций, а также интеграцию с другими языками программирования и внешними устройствами, что делает её незаменимым инструментом для исследователей и инженеров в областях обработки и анализа изображений.

Особенно в калибрации нас интересует соответствующий раздел Camera Calibration, в вводных данных для его работы требуется точный размер сторон квадратиков доски (померили с помощью инструмента штангенциркуля и получили значение 12 мм) и множество изображений калибровочной доски в различных положениях пространства, но на примерном стандартном расстоянии от объектива до плоскости исследуемых изделий.

Прогоняя полученные изображения через Camera Calibration, получаем что программа некоторые фотографии сумела откалибровать как на рисунке 10, то есть определить эту сетчатую структуру (некоторые следующие изображения, на которые ссылается дальнейший текст, располагаются в Приложении Б).

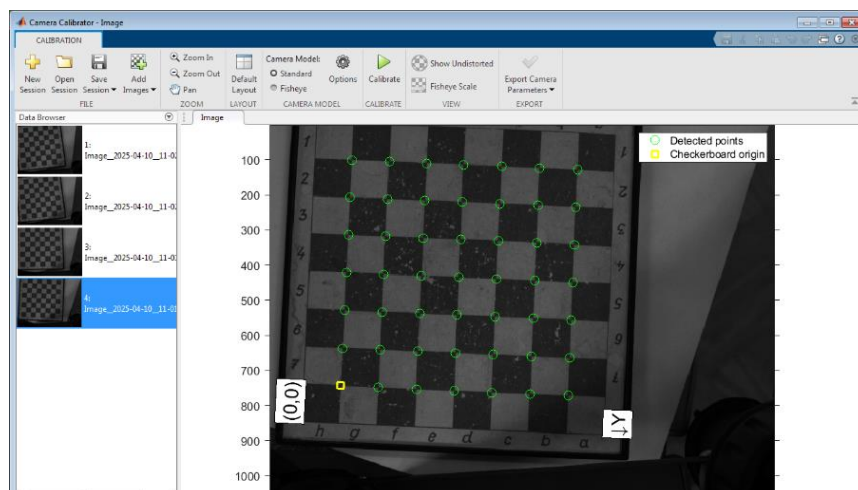


Рисунок 10 –Откалиброванное изображение доски

С помощью нескольких таких откалиброванных изображений программа способна определить расстояние, наклон, среднее отклонение относительно истинного положения и размеров калибровочной доски. В нашем случае на основании опытов размер этого среднего отклонения составил 0,56 пикселя (то есть менее одного пикселя).

Зная эти параметры можно спроецировать истинное положение дел, это является наглядным показателем с чем можно сравнивать искомую фотографию. На самом деле при погрешности в 0,56 пикселей отличить с ходу человеческим глазом разницу между искомым и истинным трудно. Это упрощается если вычесть из первого второе изображение, благо это производится в том же модуле.

Когда искомое и истинное изображение совпадают то этот пиксель окрашивается в чёрный. Как видно в конечном результате вычитания значимая часть всех пикселей полностью совпала, отличны лишь границы. Отклонения не влияют на дефекты, значит с уверенностью можно сказать что оптические отклонения на результат работы машинного зрения и обучения не возымеют действия.

Далее картинка подвергается процессу фильтрации, это помогает лучше выявлять и определять изучаемые параметры поверхности головки. Например, на сегодня существует множество методов фильтрации в сфере обработки

полученных фотографий и видео в том числе, в экспериментальной части мы рассмотрим следующие: изменение резкости, изменение контрастности изображения, выделение границ, бинаризация. Все это функции так же можно исполнить в программе.

Для улучшения видимости отдельных элементов на изображении можно через параметры его чёткости при определённом разрешении добиться нужного параметра резкости. Резкость описывает различимость деталей на фотографии, и она может использоваться как важный творческий инструмент для выделения определённых текстур.

Резкость применяется наряду с другим параметров обработки как контрастность. Контрастность - способность фотографического материала или оптической системы воспроизводить разницу границ разного цвета. Её можно, как и повышать для лучшего чтения границ, так и понижать (размытие Blur) для уменьшения избыточного шума в изображении, особенно полезно для нейронных сетей - акцентирования для них дефекта.

Достаточно часто в случае сбора материала для обучения системы искусственного интеллекта полезно пользоваться инструментом выделения границ. Выделение границ - термин в теории обработки изображения и компьютерного зрения, частично из области поиска объектов и выделения объектов, основывается на алгоритмах, которые выделяют точки цифрового изображения, в которых резко изменяется яркость или есть другие виды неоднородностей.

В пограничных случаях для лучшего освещения ситуации поверхности заготовки принят метод бинаризации изображения. Бинаризация заключается в объединении информации с нескольких пикселей таким образом, чтобы в среднем она была получена одним большим пикселем [22]. В монохромной картине тёмные участки становятся только чёрными, а светлые – белыми, как на рисунке 11.

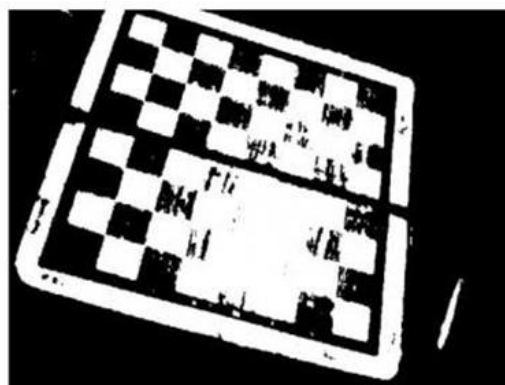


Рисунок 11 - Пример бинаризации изображения шахматной доски

В любом случае последующая обработка полученного изображения является творческим аспектом работы человека в экспериментальной части. Благодаря ему картина поверхности легко распознаётся и на человеческий глаз, и на систему машинного зрения со встроенным алгоритмом машинного обучения.

На самом деле использование всех вышеперечисленных методов относится только к лабораторным исследованиям. В дальнейшем обычные матричные камеры Basler должны быть заменены на более современные телецентрические и би-телецентрические камеры.

Основным плюсом их объективов является улучшенная глубина резкости, благодаря умной системы такие объективы не чувствительны к изменению расстояния от объектива до поверхности изделия, то есть получают изображения высокой точности и резкости в определённых диапазонах расстояний.

В условиях массового производства АвтоВАЗа система не должна подвергаться сильному изменению. Таким образом требуется настроить впервые камеру на головку блока цилиндров и пользоваться ей до следующего изменения номенклатуры изделия без проблем. Нужно лишь раз перед сменой проходить проверку настроенности оборудования специализированным человеком по специально заготовленным местам детали или другим шаблонам.

2.4 Обработка и выявление дефектов для обучения

Для обучения нейронной сети необходимы тысячи или даже миллионы разнообразных изображений рассматриваемого объекта. Изображения должны отвечать требованиям разнообразия, а также желательно не иметь излишнего видимого шума. Получить такое количество изображений достаточно трудно в нынешних условиях, поэтому наилучшим способом является искусственное увеличение числа изображений с помощью алгоритмов.

Все эти манипуляции комфортно производить в программе с помощью циклов, полученный код предоставлен в Приложении Б. Кратко – изображение получается путём копирования исходного с поворотом на 15° по часовой стрелке. В результате вместо одного исходного изображения дефекта получается 24, то есть из 87 сделанных вживую получается $87 \cdot 24 = 2088$ – что повышает степень валидации нейронной сети. Для бездефектных фотографий это значение равняется $25 \cdot 24 = 600$ штук.

Для увеличения точности работы нейронной сети требуется уменьшить визуальный шум на фотографии. Для этого, логичнее всего, непосредственно выделить сам дефект на фотографии (самостоятельно локализовать). Это можно провести и в обыкновенной пользовательской программе для работы с изображениями Paint.

Таким образом проведя подобный алгоритм естествоиспытатель получит качественный и многочисленный экспериментальный материал, который используется в дальнейшем рабочем цикле обучения и разработки.

3 Проведение экспериментов

3.1 Общие сведения

Одной из важных составляющих любой научной работы является экспериментальная часть. Главная её задача состоит в дефектоскопии поверхности и проверке достоверности данных.

Данный раздел состоит из трёх крупных этапов, следуя этой системе, мы сможем точно определить соответствующую информацию из эксперимента. Этими этапами являются: подготовка, сбор и обработка данных.

В подготовительном этапе проводим планирование эксперимента. Планирование заключается в подготовке образца или приблизительной модели (алюминиевой детали с относительно похожими конструктивными элементами и химическим составом) и удобного рабочего места.

Рабочее место включает в себя, как и испытуемый материал, так и специальное оборудование для лабораторной работы. Оно должно отвечать не только потребностям научной деятельности, но и правилам безопасности (особенно во время обработки и нанесении искусственных дефектов на поверхности изделия).

Второй этап включает непосредственный сбор данных. Здесь с помощью камеры получают изображения поверхности, нужно получить как поверхности с дефектами, желательно с нескольких сторон, так и идеальные места поверхности. Нужно получить как можно больше дефектов для лучшего обучения нейронной сети в будущем.

В последнем подразделе происходит процесс обработки полученных изображений. Для этого необходимо воспользоваться информацией из вышестоящего подраздела 2.3 «Обработка полученных изображений».

Из всего вышестоящего, основываясь на этих теоретических сведениях, можно приступать к практической составляющей работе.

3.2 Подготовка образца

Конечно, наиболее рационально заниматься экспериментальной частью над настоящим объектом дальнейшей дефектоскопии – Головкой блока цилиндров. Но так как в виду большого размера, высокой стоимости, ограниченных сроков и запрета, получить изделие на предприятии «АвтоВАЗ» оказалось невыполнимой миссией.

Для дальнейших экспериментов было решено вместо Головки блока цилиндров взять похожий аналог – советскую бытовую мясорубку «КАТЭК». На рисунке 12 представлен прибор в разобранном состоянии со всеми оставшимися деталями.



Рисунок 12 – Детали мясорубки «КАТЭК»

Данная мясорубка в СССР относилась к изделиям культурно-бытового назначения и была изготовлена по ГОСТу 4025-83 «Мясорубки бытовые». Корпус агрегата мог быть изготовлен из следующих материалов: сталь, чугун

и алюминий, в нашем примере корпус алюминиевый, из алюминия также изготовлены отдельные детали такие как ручка (с деревянной вставкой), шнек и другие.

В соответствии с ГОСТом 4025-83 мясорубка была подвергнута контролю во избежание дефектов на её поверхности включая: раковины, шлаковые включения и трещины. По предварительному визуальному контролю поверхности мясорубки, несмотря на продолжительный период её использование по прямому назначению, явных вышеперечисленных дефектов не имеют. Примем их за полностью бездефектные.

Таким образом выбранный аналог совершенно точно подходит для продолжения экспериментальной части, погрешности в проведении минимальны, а преимущества неоспоримы.

3.3 Организация рабочего места экспериментатора

Как и говорилось ранее, залогом получения удовлетворительных результатов работы специального оборудования является грамотное обеспечение рабочего места экспериментатора.

Для полноценной работы нужно произвести следующий алгоритм организации. Он включает в себя следующие основные элементы: рабочий стол, где будет располагаться алюминиевое изделие, штатив на котором с помощью специальной оснастки закреплена матричная камера Basler, через подключенный к камере кабель (на самом деле корректно называть именно технологию Gigabit Ethernet (GigE) как способ передачи Ethernet-кадров) получаемое изображение передаётся в программу Pylon.

На рисунке 13 показана компоновка технологического оборудования для съёмки фотокадров деталей ручки, корпуса и уголка. Как можно заметить камера на штативе закреплена вертикально вниз по направлению к столу. Специальная дополнительная лампа не понадобилась, так как норма освещённости в кабинете соответствует нормативам для дефектоскопии.

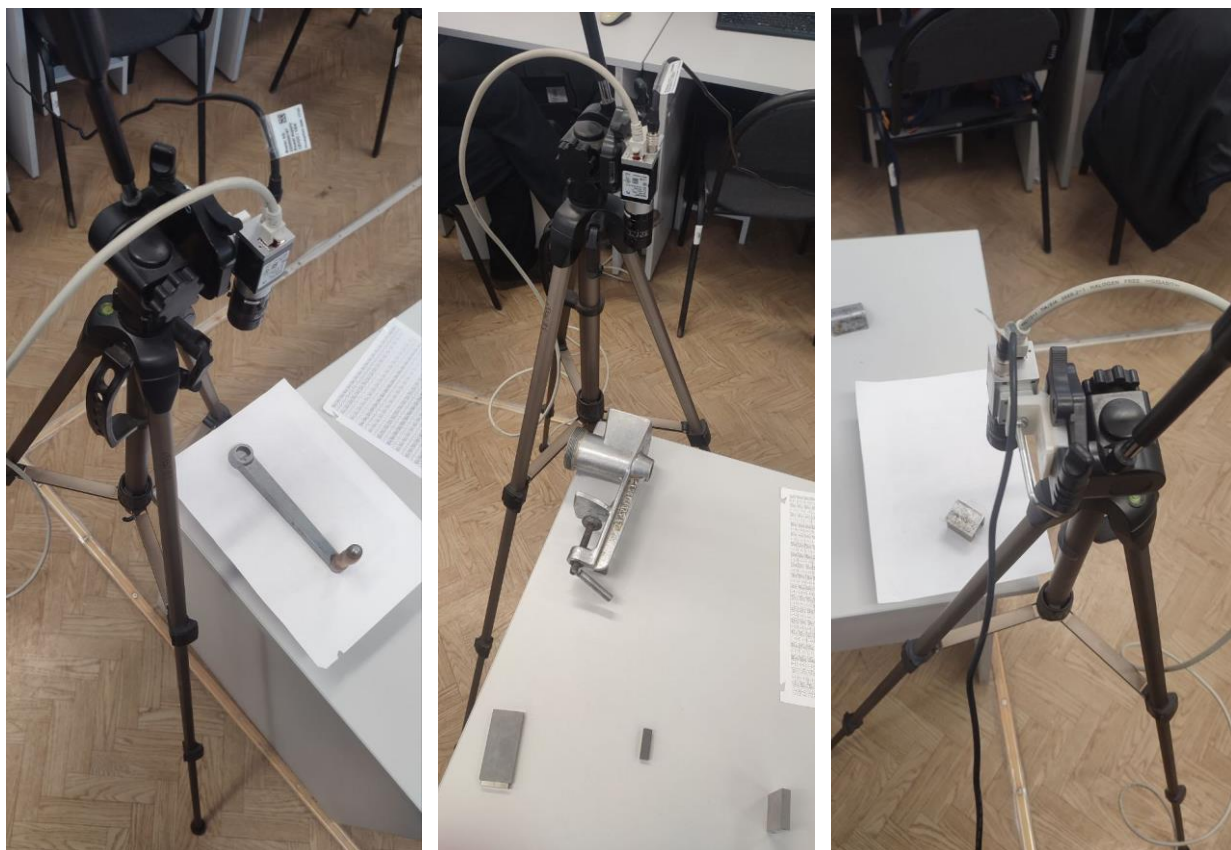


Рисунок 13 – Организация места проведения эксперимента

Предварительно, как и любая камера, Basler требует дополнительной фокусировки на изображении, с помощью круговых переключателей, она даёт возможность изменять значения освещённости, резкости и фокусного расстояния. В работе примем расстояние от объектива камеры до рассматриваемой поверхности изделия равным 300 мм. Теперь любая поверхность, которая подвергается фотосъёмке должна быть расположена на этом соответствующем расстоянии, для чего воспользуемся дополнительно концевыми мерами.

4 Внедрение системы машинного обучения

4.1 Общие сведения об нейронных сетях

Важнейшей частью в проектировании контрольного процесса является не столько получение соответствующего информации о поверхности заготовки, сколько сама обработка этих данных. С одной стороны вопроса, благодаря правильной фильтрации изображений, даже обычный человеческий глаз способен определить имеющиеся погрешности и их координаты, но это принципиально не решает проблему автоматизации контрольного процесса с целью уменьшения операционного времени технологического процесса.

Таким образом, если традиционные методы решения проблемы не имеют должного эффекта, то следует обратиться к современной науке в области программирования. Одной из самых быстро развивающихся сфер жизни и работе человека являются нейронные сети.

Нейронная сеть - математическая модель, а также её программное или аппаратное воплощение, построенная на принципах разложения, обработке, передаче и синтезе большого числа простых процессоров (искусственных нейронов) [14]. Появление нейронных сетей случилось благодаря объединению научных знаний в таких научных дисциплинах как: биология, математика и программирование.

Важно понимать, что нейронные сети не программируются в привычном смысле этого слова, они обучаются на входных данных [4]. Обучения является самым главным преимуществом нейронных сетей перед традиционными алгоритмами. Технически обучение заключается в нахождении коэффициентов связей между различными нейронами с помощью специальных, оптимизационных алгоритмов. В процессе обучения нейронная сеть способна выявлять сложные зависимости между входными и выходными данными, а также выполнять обобщение. Это значит, что в случае успешного обучения сеть сможет вернуть верный результат на основании данных,

которые отсутствовали в обучающей выборке, а также неполных, частично искажённых и неточных данных.

В однослойных нейронных сетях существует только два типа нейронов – входные и выходные. В многослойных появляются ещё один важный класс скрытых нейронов, в которых и происходит основной процесс обработки полученных данных. Решения обладают большими возможностями, с ростом количества этих слоёв также повышаются мощности этой нейронной сети.

После разработки алгоритмов обучения, получаемые модели стали использовать в практических целях: в задачах прогнозирования, для распознавания образов, в задачах управления и других. Именно поэтому использование нейросетевых механизмов позволит нам решить вопросы автоматизации в работе.

4.2 Свёрточные нейронные сети

Если работа по обработке большого массива данных чисел и значений с помощью нейросетевых механизмов велась достаточно давно и вполне удачно, то положительные аспекты работы с изображениями имели успех относительно недавно, привлекая внимание лишь с 2012 года.

Дело в том, что компьютер видит любое изображение как массив пикселей. По цветовой система каналов RGB (красного, зелёного и синего цвета) каждый пиксели имеет присвоенное значение интенсивности от 0 до 255. Дальше с помощью специальных механизмов необходимо представить эти данные в удобную и корректную форму, которую может считать система искусственного интеллекта. Для этого существует несколько способов.

Первый наиболее простой способ MLP. При нём рассматриваемая картинка «разворачивается» в один длинный вектор, а затем проходит через обычную многослойную сеть. Такой способ нежелателен так как имеет несколько важных недостатков: вектор имеет слишком большую длину и много параметров, также происходит потеря самой структуры изображения.

На сегодняшний момент наилучшими инструментами в области нейросетевого моделирования для обработки изображений и видео являются свёрточные нейронные сети (по-английски Convolution Neural Network (CNN)). Такие нейросети хорошо умеют распознавать и классифицировать объекты на картинке. Например, найти, классифицировать, и выделить на изображении дефекты поверхности заготовки [25, с.272].

Свёрточные нейронные сети состоят из нескольких слоёв. Примерная структура этих слоёв, их последовательность представлены на соответствующем рисунке 14. Чем больше слоёв, тем мощнее архитектура и лучше в обучении. Первым и самым важным слоем в этом типе является свёрточный (Convolution). Дело в том, что CNN представляет изображение в особом представлении — в виде трёхмерных массивов чисел или массивов матриц. В математике это называется тензорами.

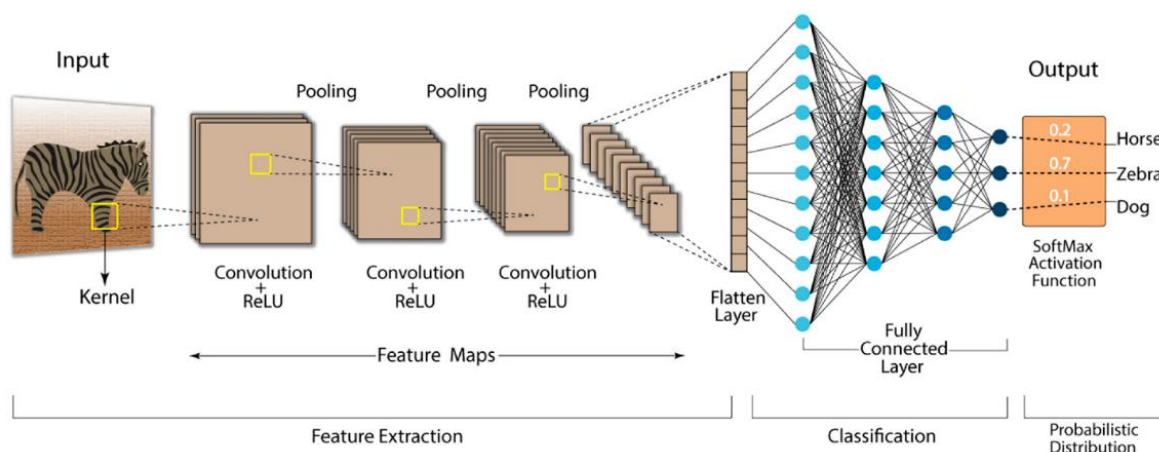


Рисунок 14 – Слои и алгоритм работы свёрточной нейронной сети

Основной принцип работы свёртки – на изображение последовательно, с определённым шагом, накладываются фильтры (их иногда называют, ядра свёртки). Каждый фильтр представляет собой отдельную матрицу, в количество этих слоёв равняется числу выделяемых признаков.

Фильтр проходит по каждому пикселю изображения, перемножает и суммирует числа своей матрицы и матрицы картинки, затем на выходе

получается новая матрица, потом к каждому значению матрицы прибавляют шаг, на который перемещался фильтр по изображению.

Все каналы, или матрицы после последнего шага фильтра (то есть обработке полного изображения) объединяются в один тензор. В итоге получается то же изображение, только другого размера и с другим числом каналов [2].

Таким образом фильтр формирует свою карту активации (данные элементы не совсем корректно называть картами активации, так как она является выходом со свёрточного слоя до функции активации, после активации - это карта признаков, но примем данное допущение, чтобы отличать тензоры свёртки), если получившиеся в карте активации значение большое, то это означает, что соответствующей ей фильтр в соответствующей ей зоне тензора нашёл то, на что был настроен, а если значение наоборот маленькое, то алгоритм не нашёл нужной ему информации.

Карта активации состоит из значений, полученных при прохождении фильтра только определённой области входного тензора, смещаясь, фильтр получает следующее значение (смотри рисунок 15). Каждое значение является технически нейроном свёрточного слоя, и зависит только от этой видимой области тензора – receptive field (область видимости) [3].

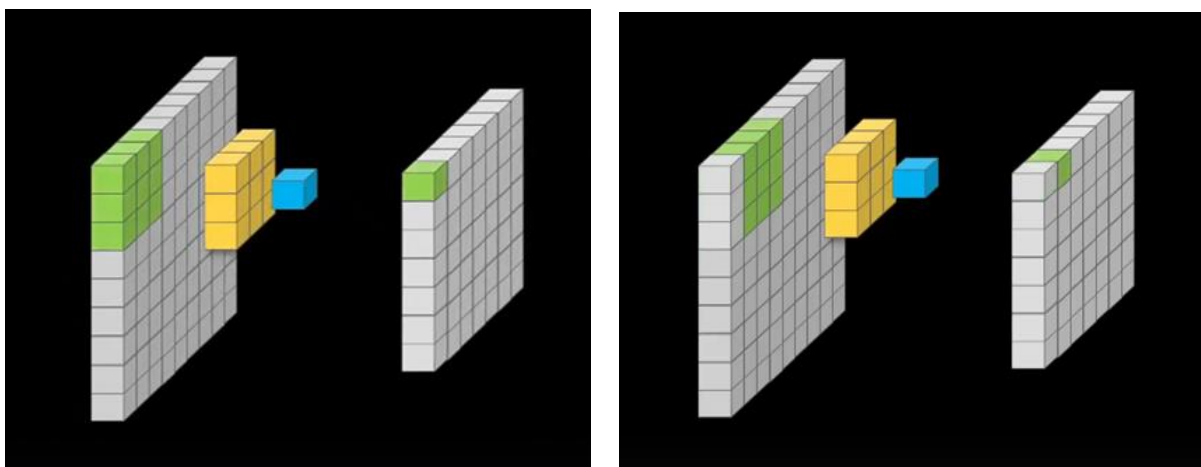


Рисунок 15 – Формирование карты активации из входного тензора

На самом деле во время этапа свёртки нейросеть проводит удаление лишней поступающей информации (различные шумы изображения) и оставляет полезную, которая может помочь проанализировать изображение. Каждый фильтр свёртки ищет свою информацию (создаётся для каждого признака), например, на самых ранних слоях ищутся линии, формы или ровные области, далее – фигуры, затем формы и тому подобное. Нейронная сеть будет сама подбирать их во время распознавания и классификации на каждом свёрточном слое.

Благодаря этим операциям, нейросеть получает множество нейронов для последующей классификации по каждому признаку с определённого receptive field как на рисунке 16. Это значит, что потери информации будут достаточно минимальны, для корректной работы.

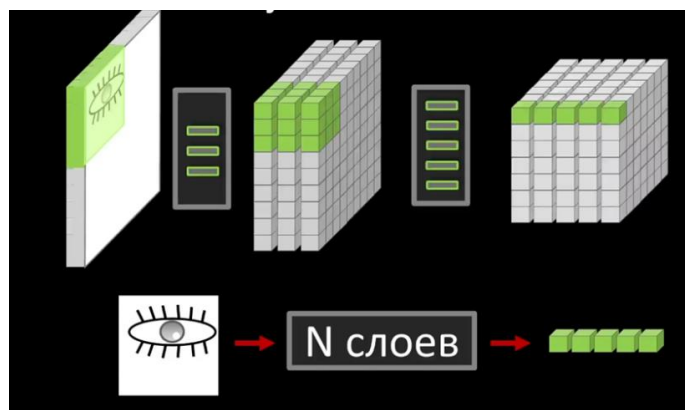


Рисунок 16 –Получение значений тензоров с информацией по определённой области видимости

На этом обработка не заканчивается, после свёрточного слоя идёт слой пулинга (Pooling). Из признаков, свёрточного слой, выбираются лишь самые важные, а несущественные удаляет [7]. На самом деле к результату, который получился во время пулинга, можно и нужно снова применить свёрточный слой, последовательно проводя несколько таких циклов, выстраивая при этом иерархию признаков: от самых примитивных, например мелких пор, до более сложных, например крупных трещин, спаев или рыхлости поверхности.

Во время циклов свёртки и пулинга полученные матрицы информации нейросеть определяет всё больше и больше деталей, заканчиваются пространственные признаки, связанные слоями, коллапсом всех измерений, кроме измерения партии, в одномерный массив. Такое состояние называется Flatten layer.

Далее этот одномерный массив проходит через классические алгоритмы нейросетей - Fully connected layer. В этом слое (слоях) в котором каждый следующий нейрон связан со всеми выходами предыдущего слоя. В результате присутствуют все возможные соединения между слоями, то есть каждый вход входного вектора влияет на каждый выход выходного вектора, из чего следует, что все простые параметры данных комбинируются в выход [15], [24].

В выходе, на основании фильтрации, выделения и комбинирования данных, нейронная сеть выдаёт несколько результатов. Каждый из них оценивается с определённым процентом достоверности. Чем мощнее и лучше обучена нейросеть, тем точнее результаты и выше их достоверность.

Именно для этого в предыдущем разделе мы подробно собирали и обрабатывали полученные фотографии дефектных и нормальных поверхностей.

4.4 Разработка машинного кода

Работа любой нейронной сети основывается прежде всего на знаниях и принципах программирования, а они — на математической базе. Среди прочего, для самостоятельного написания машинного кода требуется владение навыками линейной алгебры, математического анализа, работы с матрицами, и массивами и другими.

Создать нейронную сеть с нуля представляется достаточно сложным, так как для корректного обучения программы необходимо от нескольких сотен тысяч до миллионов картинок на основе тысяч классов. В результате единственным возможным решением является выбор уже готовой сети с

определённым уровнем знаний. Благодаря чему удаётся оптимизировать затраты на дообучение и переобучение нейронных сетей, всего лишь создав несколько новых классов дефектных и не дефектных поверхностей изделия.

Среди таких фреймворков известными является среда Pytorch для обучения VGG-19 или Google Colab. Каждый из них обладает своими преимуществами и недостатками, своими уникальными возможностями и требуемым первоначальным навыком программирования на языке Python.

Использование таких инструментов позволяет написать собственный Data set программы обучения машинного зрения. Вообще термин Data set (или набор данных) при написании нейронной сети — это коллекция данных, используемая для обучения, валидации и тестирования модели. Наборы данных играют ключевую роль в процессе разработки и оценки нейронных сетей.

Структурно Data set можно поделить в рамках двух основных наборов: обучающего и валидационного. Обучающая часть (Training set) состоит из нескольких иерархических структур по выгрузке и обработке предоставленной информации, как на рисунке 17.

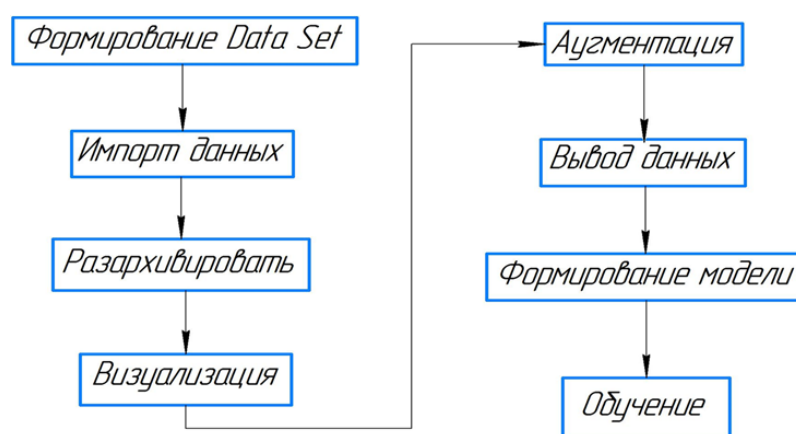


Рисунок 17 – Структура обучающего набора

Само обучение в этом наборе данные происходит на входе нейронной сети, где она настраивает коэффициенты веса каждого входного данного на

основе верных предположений и ошибок, которые она делает. На выходе сеть получает значение обучающей ошибки (`train_loss`), которая показывает насколько точен полученный именно сейчас её результат.

В отличие от обучения (`Validation set`) используется для настройки гиперпараметров модели и предотвращения переобучения. Он помогает оценить, как модель будет работать на новых, неизвестных для нейросети данных. Можно сказать, что валидация (с помощью `val_loss`) является оценкой деятельности всего обучения нейронной сети, тем выше значение валидации, тем ниже уровень ошибки нейросети.

Кроме значений `loss` и Assurance о достоверности и точности результата работы сети говорят следующие 3 значения: PRE (`precision`), REC (`recall`) и F-мера (или `F-score`).

PRE показывает долю верно-положительных объектов среди всех объектов, предсказанных как положительные. Например, при классификации трещин на поверхности — это доля действительно треснувших заготовок среди всех предсказанных как трещины. REC же показывает долю верно-положительных объектов среди всех объектов, в действительности принадлежащих положительному классу. В том же примере `recall` важен, если нужно обнаружить все трещины поверхности, пусть и с некоторой долей ложных срабатываний. Этих двух параметров недостаточно, так как возможны случаи, когда при высоком значении одного из них, второе стремилось к нулю, для чего была введена F-мера. Она является средним гармоническим между `precision` и `recall`. F-мера стремится к нулю, если компоненты стремятся к нулю, и тем выше, чем PRE и REC больше.

Создание новых нейронных сетей начинается с низких значений валидации и высокой ошибки, но с увеличением количества новых данных, она проходит переобучение и соответственно изменяет уровни валидации и ошибки в противоположную сторону.

В данной ситуации, в связи с написанием не полноценной свёрточной сети, а лишь структурой `Data set`, получаемые в конце валидационные

значения должны быть велики. В идеальном случае, параметры PRE, REC и желательно F-меры должны быть не менее 0,7 – это означает, что сеть грамотно определяет верный результат согласно внутренней логики, а не случайности.

В Приложении В показан результат работы аутентично написанного машинного кода. После проведения всех необходимых модулей, загрузки материала и его вывода конечный результат работы нейронной сети, его F-score составил 0,94 для дефектных заготовок и 0,8 для бездефектных (на Батче-4 и аналогично на Батче-16).

Это отличный результат, который полностью удовлетворяет поставленной задаче, и с которым далее можно проводить дополнительные манипуляции по оптимизации процесса. Разность полученных значений для бездефектных и дефектных заготовок можно объяснить неоднородностью количества данных, при увеличении обучающего материала она сойдёт на нет.

Для сравнения, немного изменив код, было проведено повторное исследование уже на основе готовой нейронной сети VGG-19. Результат представлен в Приложении Г. В результате значения параметров достоверности почти аналогичны, это говорит о высоком уровне работоспособности нашего Data Set.

В результате готовый код полностью соответствует предварительному плану по распознаванию дефектных и бездефектных поверхностей. Для реализации более продвинутых задач и целей необходимо провести модернизацию машинного кода и обучения нейронной сети на оригинальной детали.

4.5 Использование пяти классового метода

Если в предыдущих случаях план допускал определение только дефектной и не дефектной поверхности, то логическая эволюция подразумевает уже дифференциацию дефектов на определённые виды.

Для достижения поставленной задачи имеющихся на данный момент двух классов недостаточно. Таким образом, в отличие от бинарной классификации, где существует только два класса, создают мультиклассовую классификацию количество классов в которой может быть более.

Мультикласс в машинном обучении относится к задаче классификации, где модель должна предсказать одну из нескольких возможных категорий или классов для каждого входного примера. Увеличении количества дополнительных классов приводит к значительной трудозатратности написания и обучения нейронной сети.

В Приложении Д показан машинный код решения пятиклассовой задачи. В отличии от прошлых решений здесь код можно чётко разделить на две составляющие: обучение на основе предобученной сети VGG-19 и внедрения пяти классов дифференциации результата.

Несмотря на то, что сделанный DataSet на параметрах VGG-19 показал себя менее точно, чем первоначальное обучение, но использование заранее подготовленной нейронной сети позволяет в разы упростить решение задачи, особенно в части работы с оптимизацией, переобучение и дообучением сети.

Для повышения конечного результата точности нейронной сети до 98%-100% требуется определить в дальнейшем корень проблемы. Среди возможных действий можно, например, сократить число последовательностей свёрточных слоёв и пулинга (нейросеть может проводить излишнее переобучение на незначительных параметрах) или настроить коды остановки обучения (для того же) и так далее. Как итог, конечный результат способен улучшиться выше значений первого, оригинального обучения.

Если в бинарной задаче выделялось два класса: дефект-0 и не дефект-1 (а в VGG-19 1 и 0 менялись местами), то в пяти классовой задаче их больше. При обучении модели важно учитывать баланс классов, так как несбалансированные данные могут привести к смещению в предсказаниях. Была проведена дифференциация, где каждому классу была присвоена своя цифра, начиная от 0 и до 4:

- a. 0 – no defects (бездефектные поверхности);
- b. 1- attrition (потёртости и излишние шероховатости);
- c. 2 – pores (поры и раковины);
- d. 3 – scratch (царапины, одинокие и несколько);
- e. 4 – label (дефекты на маркировке).

С повышением уровня сложности требуется все больше и больше методы контроля за конечным результатом, программированием предлагает большой спектр разнообразных алгоритмов проверки. В рассматриваемом случае выбирается два: ранее известная f-мера и micro усреднение – она применяется, когда важнее общий результат.

После обучения, полученные результаты работы нейронной сети не являются удовлетворительными, особенно по f-мере. Нейронная сеть не “видит” классы 0 (бездефектный) и 2 (поры и раковины), зато чрезмерно отправляет их в класс 1 (потёртости).

Но определённо, с продолжением развития машинного кода и обучением на новых данных, можно говорить о реализации следующей ступень эволюции.

Из всего этого можно сделать вывод, что мультикласс в задачах машинного зрения упрощает процесс обработки и анализа изображений, так как модель может делать более точные предсказания, учитывая все возможные классы, что особенно важно в сложных сценариях, таких как распознавание объектов.

5 Организация рабочего места проведения контрольной операции

5.1 Разработка стенда контрольного

Конечная задача работы состоит в разработке конструкции и организации рабочего места. Рабочее место - это часть производственной площади цеха или участка на предприятии, которая закрепляется за определенными работниками в случае посменной работы.

Автоматизация рабочего места состоит в создании системы мероприятий по оснащению рабочего места передовыми научными средствами и предметами труда и их размещение в целях минимизации человеческого труда [16].

Рациональная организация рабочего места должна создавать условия для уменьшения влияния человеческого труда на готовый продукт, уменьшение числа работников при том же уровне производственных мощностей, улучшение условий труда по средствам создания благоприятного положения тела, рук и головы контролёра; способствовать рациональному размещению средств оснастки; обеспечивать простоту и надежность в обращении со средствами и предметами труда и уменьшать монотонность самого труда.

Автоматизированное рабочее место контроля методом оптического зрения должно быть по возможности отдельно от производственного оборудования, чтобы минимизировать влияние внешнего окружения на получение результата, особенно с учётом применения высокочувствительного оборудования, по типу телескопических камер.

Оно обязано быть оснащено исправными средствами контроля и иметь достаточное освещение, как для работы человека по правилам безопасности труда, так и для адекватного результата системы машинного зрения. По российским нормам СанПиН 2.2.1/2.1.1.1278-03 освещённость лабораторий

научно-технического комплекса (кроме медицинских учреждений) должно составлять не менее 400 лк при общем освещении.

Любое проектирование начинается с разработки концепта механизма. Он должен соответствовать нужным параметрам:

- степени автоматизации механизма;
- включения компонентов сборки и использования оборудования;
- общему расположению элементов системы.

В результате был создан эскиз предварительной компоновки сборки участка автоматизированного рабочего механизма, показанный на рисунке 18.

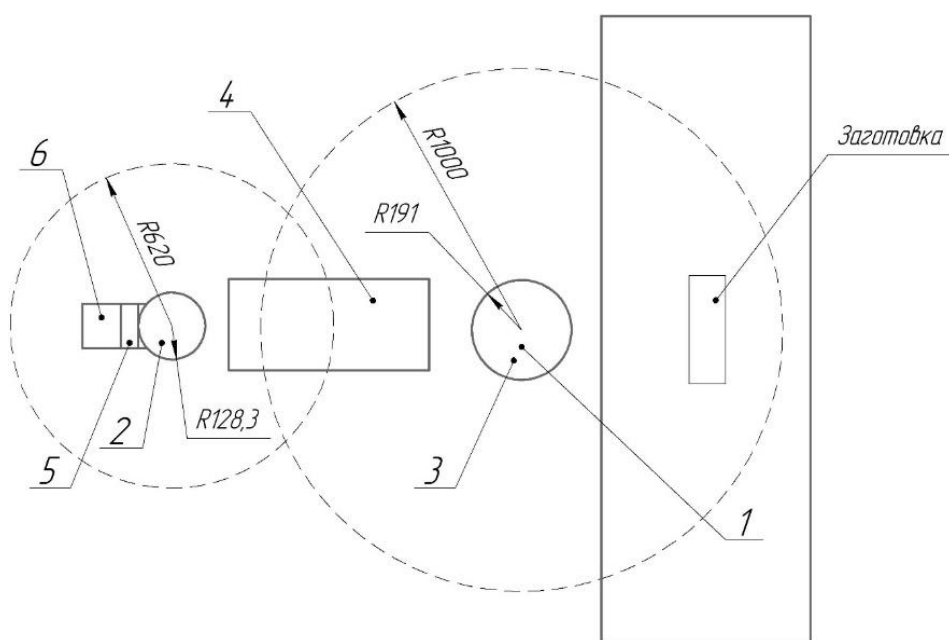


Рисунок 18 – Компоновка участка

Высокий уровень автоматизации достигается использованием двух роботов 1 и 2. Робот 1 захватывает и транспортирует заготовку с транспортной линии 3 на поворотную плиту 4. А робот 2 с системой подсветки 5 удерживает камеру 6 и переносит её объектив в заданные зоны контроля заготовки.

Важнейшую роль в проектировании данного стенда является адекватный выбор роботов - манипуляторов. Наиболее подходящими являются коллаборативные роботы DOBOT от компании ООО «Тесвел».

Коллаборативный робот (кобот) — это современный промышленный робот со встроенной функциональностью, которая обеспечивает возможность совместной работы в рабочем пространстве одновременно с человеком в рамках одной производственной операции.

В отличие от традиционных промышленных роботов, такой манипулятор более доступен, подвижен, отличается компактными размерами, обладает повышенной производительностью, надежен и прост в применении. Эти устройства позволяют оперативно заменить квалифицированных специалистов, когда требуется ускорить производственные процессы, но для этого на предприятии недостаточно опытных сотрудников.

Робот 1 – это коллаборативный робот DOBOT CR16, так как он способен перемещать изделия до 16 кг (масса заготовки головки блока цилиндров – не более 12 кг), а максимальный радиус охвата – 1000 мм. Робот 2 является коллаборативным роботом DOBOT CR3, при его характеристике: максимально грузоподъёмность – 3 кг, радиус действия – 620 мм.

При работе камер машинного зрения, для получения качественных изображений поверхности требуется специализированная подсветка, а также захватывающие устройства. Тесвел вместе со своими партнёрами предлагают для этого готовые решения: захват с широким ходом для коботов OnRobot RG2 для захвата камеры; трёхпальцевый захват для коботов OnRobot 3FG15 для захвата заготовки, и камера OnRobot Datasheet Eyes. Технические характеристики к оборудованию представлены в источниках: https://osnastik-robots.ru/f/spietsifikatsiia_rg2_v11_ru.pdf, https://osnastik-robots.ru/f/spietsifikatsiia_3fg15_v12_ru.pdf, https://onrobot.com/sites/default/files/documents/Datasheet_Eyes_EN_0.pdf.

Из всего, выше сказанного, можно создать полноценный сборочный чертёж рабочего места, при этом предпочтение отдаётся стандартизированным, уже существующим на рынке, зарекомендовавшим себя решениям. Прикреплённую к чертежу техническую документацию (спецификацию) прикладывается в Приложении Е.

5.2 Базирование заготовки на рабочем месте

Базирование в проведении контрольной операции — это важный этап, который обеспечивает эффективность и точность проверки. Во время контроля заготовка должна быть строго запозиционирована как на конвейере, так и на контрольной плите [17].

Предварительно примем, что на линии конвейера заготовка запозиционирована на обычной плите. Это необходимо, так как требуется соблюдение точности расположения изделия, для последующего его захвата роботом-манипулятором.

Деталь обязана быть крепко закреплена на поворотной плите, то есть не иметь свободных координат для перемещения. Это должно уменьшить ошибку при контроле, так как обеспечивает постоянство расстояния между поверхностью головки блока цилиндров и объективом камеры, закреплённой на коллаборативном роботе. В результате, система машинного обучения точнее определяет достоверность полученных изображений, потому что они стремятся к идентичности, а роботы-манипуляторы строго следуют программе перемещений, без эксцессов.

В базировании на поворотной плите применяется, в условиях массового типа производства, специальная оснастка. На предприятии АвтоВАЗа определённо существует такая, но, на данный момент, в виду конфиденциальности, не присутствует данная техническая документация.

В отсутствии полноценной документации: чертежа детали и заготовки, инструментальной оснастки и других, не представляется возможным полностью разработать и/или внедрить средства базирования. Во-первых, разработанная предварительная оснастка на самом деле может не соответствовать по техническим и технологическим параметрам. Во-вторых, трата времени на разработку возможно уже существующего приспособления экономически нецелесообразно. Это значит, что нужно проработать вопрос дополнительно для получения последующей информации.

6 Безопасность и экологичность технического объекта

Современное производство должно быть не только эффективным, но также и безопасным для жизнедеятельности человека. Во избежание производственных травм и чрезвычайных ситуаций при разработке любого устройства необходимо провести комплекс работ по обеспечению его безопасности и экологичности.

В таблице 1 показан паспорт объекта, с помощью которого можно определить опасности, возникающие в процессе эксплуатации представленного оборудования.

Таблица 1 – Паспорт объекта

Объект	Технологическая операция	Наименование должности работника	Оборудование, техническое устройство, приспособление	Материалы и вещества
Стенд контрольный	Контрольно-измерительная	Рабочий-контролёр	Стол контрольный, камера телескопическая, рабочий ПК, роботы-манипуляторы	Металл, пластик, стекло

В таблице 2 приведены результаты идентификации производственно-технологических и эксплуатационных рисков, к которым относят вредные и опасные производственные факторы.

Таблица 2 - Определение рисков

Технологическая операция	Опасный и вредный производственный фактор (ОВПФ)	Источник ОВПФ
Контрольно-измерительная	Высокий уровень напряжения в электросети	Электрический ток
	Повышенный уровень инфракрасного излучения, рентгеновского излучения и ультрафиолетового излучения	Экран монитора, фотокамера

Продолжение таблицы 2

Технологическая операция	Опасный и вредный производственный фактор (ОВПФ)	Источник ОВПФ
Контрольно-измерительная	Неравномерное распределение яркости в поле зрения рабочего	Экран монитора, освещение
	Скопление пыли и аллергенов на компьютерной технике и оборудовании	Рабочее место, системный блок компьютера, элементы механизма
	Физические: движущиеся машины и механизмы, их незащищенные подвижные части, острые кромки, заусенцы и шероховатость на поверхности заготовок и оборудования	Элементы механизма, роботы-манипуляторы

Каждый фактор может привести в последствии к производственной травме, среди которых можно выделить удар током, порезы царапины, удары и другие, которые приводят к потере здоровья рабочих на предприятии.

Для защиты работников от негативных факторов необходимо использовать на предприятии средств индивидуальной защиты (СИЗ) и проводить мероприятия по снижению уровня опасных факторов. Все они также приведены в соответствующей таблице 3.

Таблица 3 - Мероприятия по снижению уровня ОВПФ

ОВПФ	Возможные последствия влияния негативных факторов	Технические средства, организационные методы	Средства защиты (СИЗ)
Движущиеся машины и механизмы	Ранения, царапины, порезы, удары	Применение ограждений, проведение инструктажа	Костюм для защиты от производственного и механических воздействий, перчатки с полимерным покрытием
Высокий уровень напряжения в электросети	Удар током	Инструктаж по технике безопасности с интервалом не менее чем раз в 6 месяцев.	Периодическое проведение инструктажа по технике безопасности. Осмотр ПК на наличие заломов проводов, исправности всех разъемов, повреждение изоляции.

Продолжение таблицы 3

ОВПФ	Возможные последствия влияния негативных факторов	Технические средства, организационные методы	Средства защиты (СИЗ)
Повышенный уровень инфракрасного излучения, рентгеновского излучения и ультрафиолетового излучения	Головные боли, неприятные ощущения в глазах (сухость, слезоточивость) Ухудшение зрения.	Инструктаж по технике безопасности с интервалом не менее чем раз в 6 месяцев.	Монитор должен находиться на расстоянии не ближе 500 мм. Монитор должен быть оснащен защитным фильтром и иметь возможность регулировки положения
Неравномерное распределение яркости в поле зрения рабочего	Потеря остроты зрения	Инструктаж по технике безопасности с интервалом не менее чем раз в 6 месяцев.	Использование световых источников в указанных санитарными нормами диапазонах.
Скопление пыли и аллергенов на технике	Заболевания органов дыхания, аллергия.	Инструктаж по технике безопасности с интервалом не менее чем раз в 6 месяцев	Периодическая уборка помещения, продувка системного блока от пыли. Проветривание помещения.

Класс пожара А - пожары, связанные с горением твердых горючих веществ и конструкционных материалов. В следующей таблице 4 оформлены опасные факторы пожара.

Таблица 4 - Классы и опасные факторы пожара

Участок	Оборудование	Класс пожара	Опасные факторы пожара	Сопутствующие проявления факторов пожара
Контрольно-измерительный	Компьютер, фотокамера	А	Высокая температура, задымленность, выделение ядовитых и канцерогенных паров при сгорании легкоплавких веществ: пластмассы и резины	Части объектов, разрушившихся при возгорании, распространение тока при повреждении проводки, вредные выбросы в атмосферу, отравление окружающей среды продуктами горения

Исходя из данного класса пожара, необходимо применять средства пожарной безопасности по таблицам 5-7.

Таблица 5 - Технические средства обеспечения пожарной безопасности

Средства пожаротушения				Оборудование
первичные	мобильные	стационарные	автоматики	
Ящик с песком, пожарный гидрант, огнетушители	Пожарные автомобили	Пенная система тушения	Технические средства по оповещению и управлению эвакуацией	Напорные пожарные рукава

Средства индивидуальной защиты (СИЗ) от пожара на предприятии необходимы для обеспечения безопасности работников и минимизации рисков при возникновении чрезвычайных ситуаций. Они помогают защитить сотрудников от воздействия высоких температур, токсичных газов и дымов, а также предотвращают термические ожоги и другие травмы.

Таблица 6 - Средства защиты и пожаротушения

СИЗ	Инструмент	Сигнализация
Веревки пожарные карабины пожарные противогазы, респираторы	Лопаты, багры, ломы и топоры ЩП-Б	Автоматические извещатели

Таблица 7 - Средства обеспечения пожарной безопасности

Процесс, оборудование	Организационно - технические меры	Нормативные требования
Технология изготовления вала	«Общее руководство и контроль за состоянием пожарной безопасности на предприятии. Организация пожарной охраны, проведение мероприятий по обучению сотрудников действиям во время пожара» [20]	«Наличие пожарной сигнализации, Наличие автоматической системы пожаротушения, первичные средств пожаротушения, проведение инструктажей» [20]

Важно следить не только за безопасностью рабочих на предприятии, но и соблюдать нормативные стандарты, во избежание чрезмерного антропогенного загрязнения окружающей среды процессом.

Таблица 8 - Идентификация негативных экологических факторов технического объекта

Наименование технического объекта, производственно технологическо го процесса	Структурные составляющие технического объекта, производственно технологическо го процесса	Воздействие технического о бъекта на атмосферу	Воздействие технического о бъекта на гидросферу	Воздействие технического о бъекта на литосферу
Стенд контрольный	Контрольно- измерительная операция	Выделение излишнего тепла при прохождении электрического тока	-	-

Во время эксплуатации контрольно-измерительный стенд не оказывает негативного воздействия на атмосферу, литосферу и гидросферу так как не имеет любых побочных продуктов работы.

Таким образом можно сделать вывод, что в разделе «Безопасность и экологичность технического объекта» приведена характеристика производственно-технологического процесса, перечислены технологические операции, должности работников, используемое производственно-техническое и инженерно-техническое оборудование и так далее.

Проведён соответствующий анализ профессиональных рисков по проекту. Описаны организационно-технические мероприятия, содержащие технические устройства снижения профессиональных рисков, и подобраны необходимые средства индивидуальной защиты работников.

Изучены и использованы все мероприятия по улучшению пожарной безопасности на объекте. Осуществлена идентификация класса и опасных факторов пожара, внедрены согласно стандартам меры борьбы с пожарами,

включая выбор СИЗ для работников, средств пожаротушения, средств оповещения и анализ мероприятий по обеспечению пожарной безопасности на техническом объекте.

Произведена идентификация экологических факторов, и также степень антропогенного влияния рассматриваемого оборудования на окружающую среду.

В заключении дипломной работы можно отметить, что обеспечение технологической и экологической безопасности является неотъемлемой частью устойчивого развития современного общества. Эффективное управление технологическими процессами, включая внедрение инновационных решений и соблюдение стандартов безопасности, способствует минимизации негативного воздействия на окружающую среду.

Важно интегрировать экологические аспекты в промышленную деятельность, что позволит не только защитить природные ресурсы, но и повысить конкурентоспособность предприятий. Таким образом, комплексный подход к вопросам безопасности, включающий как технологические, так и экологические меры, является ключом к созданию безопасного и устойчивого будущего.

7 Экономическая эффективность работы

7.1 Капитальные вложения, трудоёмкость и длительность работ

В обществе внедрение любого новшества должно иметь заметные положительные экономические эффекты для производства, особенно массового. Единственным правильным способом увеличить капитал для компаний – повысить уровень автоматизации для производства, чтобы уменьшить роль труда в общей стоимости.

Во-первых, требуется рассчитать сумму всех денежных средств, затраченных на приобретение материалов и деталей оборудования по таблице 9.

Таблица 9 - Капитальные вложения в устройство

Наименование	Цена за ед., руб.	Кол-во, шт.	Общая стоимость, руб.
Коллаборативный робот DOBOT CR16	2 500 000	1	2 500 000
Коллаборативный робот DOBOT CR3	1 210 000	1	1 210 000
Манипулятор OnRobot 3FG15	518 300	1	518 300
Камера	120 000	1	120 000
Стол круглый поворотный с механическим приводом	63 000	1	63 000
Система освещения	100 000	1	100 000
Транспортная линия (заказ)	650 000	1	650 000
Персональный компьютер	145 000	1	145 000
Кабель-канал	10 000	1	10 000
Оснастка специальная	56 000	1	56 000
Итого	5 316 000		

Далее необходимо определить трудоёмкость работ, для начала необходимо сначала выделить этапы, которые входили в работу по проектированию в таблицу 10.

Таблица 10 - Приводимый перечень работ

Стадии проведения исследований	Этапы научно исследовательской работы	Содержание проводимых работ
Подготовительная стадия	Выполнение комплекса подготовительных работ	1. Разработка технического задания на ВКР
		2. Изучение исходных данных
		3. Исследование имеющихся аналогов устройства. Поиск литературы
Теоретические работы	Разработка теоретической части исследования	4. Выбор подходящего метода дефектоскопии
		5. Исследование систем машинного зрения с помощью фото/видеоустройств
		6. Изучение теории машинного, нейросетевого обучения
Практические работы	Разработка и прототипирование	7. Написание кода для обучения нейронной сети
		8. Разработка компоновки стенда контрольного
		9. Разработка конструкторской документации
Заключительная стадия	Выводы по выполненной работе	10. Анализ работы прототипа
		11. Обработка полученных результатов, определение и внесение доработок

Итого трудоёмкость может быть рассчитана по формуле (1):

$$T = f \cdot D_{\text{ц}} \cdot P \quad (1)$$

где $D_{\text{ц}}$ - длительность цикла выполнения научно-исследовательской работы, дни;

P - количество участников, шт;

$f = 1$ – приведённый коэффициент.

Ожидаемая трудоёмкость рассчитывается по формуле (2):

$$T_{ож} = \frac{3 \cdot T_{min} + 2 \cdot T_{max}}{5} \quad (2)$$

где T_{min} - оптимистическая оценка трудоёмкости, дни;

T_{max} - пессимистическая оценка трудоёмкости, дни.

Расчётные значения трудоёмкости выполнения, численности участников проекта и длительность выполнения работ указаны в таблице 11.

Таблица 11 – Расчёт трудоёмкости и времени цикла выполнения работ

№ этапов	№ работ	Трудоёмкость выполнения работ, дней			Численность работников			Длительность выполнения работ и этапов с учётом численности работников, дней
		T_{min}	T_{max}	$T_{ож}$	Руков. проекта	Студент	Итого	
1	2	3	4	5	6	7	8	9
1	1	5	10	7	1		1	7
	2	28	33	30		1	1	30
	3	5	12	8	1	1	2	16
Итого				53				53
2	4	18	23	20	1	1	2	40
	5	5	12	8	1	1	2	16
	6	10	15	12	1	1	2	24
Итого				32				80
3	7	30	50	38	1	1	2	76
	8	24	36	29		1	2	58
	9	5	10	7	1	1	2	14
Итого				76				148
4	10	12	22	18	1	1	2	36
	11	7	9	8	1	1	2	16
Итого				26				52
Всего				187				333

Таким образом, рассчитываемое время продолжительности проекта составляет 333 дней.

7.2 Построение сетевого графика

Зачастую., для более наглядного представления экономических данных проекта, производят построение так называемого сетевого планирования.

Сетевое планирование - метод научного планирования и управления производственными процессами, выполняющими большие объемы работ. В рамках экономического планирования любого производства. Сетевое планирование имеет ряд преимуществ:

- наглядность технологической последовательности работ;
- составление оперативных и текущих планов
- прогнозирование сложных процессов;
- повышение эффективности.

Окончанием сетевого планирования служит построенный в конце сетевой график. График без времени показан на рисунке 19, созданный на базе таблицы 12.

Таблица 12 - Перечень работ в составе сетевого графика

№ п/п	Наименование работ	Код работ	Продолжительность, ч
1	Разработка технического задания на ВКР	1-2	30
2	Изучение исходных данных	1-3	7
3	Исследование имеющихся аналогов устройства. Поиск литературы	2-4	16
4	Выбор подходящего метода дефектоскопии	3-4	40
5	Исследование систем машинного зрения с помощью фото/видеоустройств	4-5	16
6	Изучение теории машинного, нейросетевого обучения	4-6	24
7	Написание кода для обучения нейронной сети	6-7	30
8	Внедрение ПО в систему	7-8	76
9	Разработка компоновки стенда контрольного	5-8	58
10	Разработка конструкторской документации	8-9	14
11	Анализ работы прототипа	9-10	36
12	Обработка полученных результатов, определение и внесение доработок	10-11	16

На основании чего строится график, где события изображают кружком с их номером, а работы - ориентированными стрелками, с указанием времени работы.

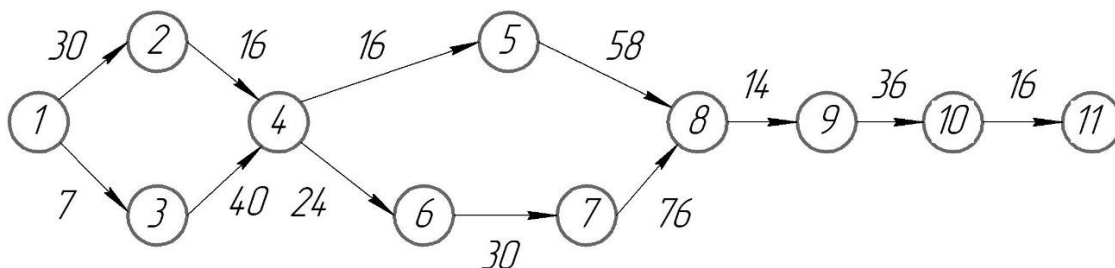


Рисунок 19 - Сетевой график

Для расчёта параметров на сетевом графике непосредственно указывают следующие параметры записи в разделённом на 4 части кружочке [19]:

- верхний - предназначен для записи номера события - i ;
- правый - для записи раннего срока свершения события - T_i^p ;
- левый - для записи позднего срока свершения события - T_i^n ;
- нижний - для записи резерва времени события - R_i .

Такой график показан на рисунке 20.

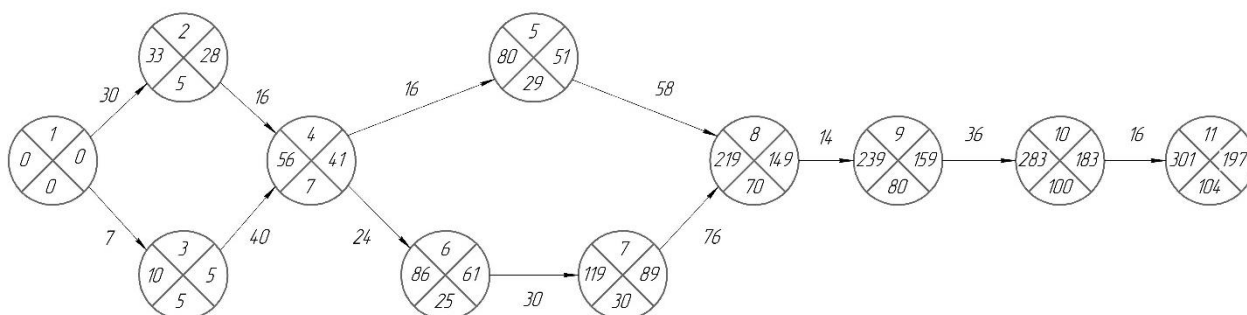


Рисунок 20 - Сетевой график разработки стенда контрольного

При дотошном расчёте длительности, вычисляются параметры, занесённые в таблицу 13, по следующим формулам (3) – (7):

1) Ранний срок начала работ (t_{ij}^{pH}) равен раннему сроку наступления события, из которого исходит данная работа:

$$t_{ij}^{pH} = T_i^p \quad (3)$$

2) Ранний срок окончания работы (t_{ij}^{po}) определяется путем прибавления к раннему сроку начала работы продолжительности самой работы:

$$t_{ij}^{po} = t_{ij}^{pH} + t_{ij} \quad (4)$$

где t_{ij} - продолжительность работы ($i - j$).

3) Поздний срок окончания работы (t_{ij}^{no}) равен позднему сроку наступления последующего события:

$$t_{ij}^{no} = T_i^n \quad (5)$$

4) Поздний срок начала работы находится путем вычитания из позднего срока наступления последующего события продолжительности работы:

$$t_{ij}^{pH} = T_i^n - t_{ij} \quad (6)$$

5) Полный резерв времени работы показывает время, на которое можно перенести начало данной работы (или увеличить её продолжительность), не изменяя при этом длины критического пути

$$r_{ij}^{CB} = T_i^p - T_i^n - t_{ij} \quad (7)$$

Таблица 13 - Таблица расчета временных характеристик сетевого графика

№ п/п	Код работы	Продолжите льность работ	Начало работ		Окончание работ		Запас времени полный
			раннее	позднее	раннее	позднее	
1	2	3	4	5	6	7	8
1	1-2	30	0	3	30	33	3
2	1-3	7	0	3	7	10	3
3	2-4	16	28	40	44	56	12
4	3-4	40	5	16	45	56	11
5	4-5	16	41	64	57	80	23
6	4-6	24	41	62	65	86	21
7	6-7	30	61	89	91	119	28
8	7-8	76	89	143	165	219	54
9	5-8	58	51	161	109	219	110
10	8-9	14	149	225	163	239	76
11	9-10	36	159	247	173	283	110
12	10-11	16	183	285	199	301	102

В результате, полученные значения способствуют определению сроков работы, логистических путей проекта, помогут в определении суммы заработной платы участников и, в целом, способствуют лучшему экономическому пониманию ситуации для предприятия [18].

7.3 Расчёт фонда заработной платы и суммарных затрат

Кроме непосредственного первоначального капитала, в состав расходов на изготовление проекта входят и, например, заработная плата участвующих лиц, то есть приобретение труда работников-пролетариев.

Состав исполнителей по каждой теме проекта определяется штатным расписанием, в котором одновременно рассматривается и фонд заработной платы исполнителей по таблице 14.

Для участников, работающих по временной зарплата, эта сумма выдаётся как произведение величины средней дневной ставки на количество отработанных дней.

Таблица 14 - Штатное расписание исполнителей и фонда заработной платы

Наименование должности	Численность, чел.	Месячный оклад, руб.	Средняя дневная ставка, руб.	Количество отработанных дней	Сумма заработной платы, руб.
Научный руководитель	1	45 000	2 045	213	435 585
Инженер (студент)	1	25 000	1 136	294	333 984
Итого					769 569

Для обеспечения непосредственной сборки, требуется пригласить специалиста – рабочего-сборщика. Примем предварительно сумму его зарплаты, с учётом сложности и объёмности работ, равной 50 000 рублей.

Итого, общие выплаты заработной платы сотрудникам проекта за весь срок реализации составляют 819 569 рублей, а с учётом отчислений в страховые фонды в размере 30% от заработной платы, они равны 1 065 440 руб.

Сумма складывается из двух составляющих: общей суммы капитальных вложений (сумма равна 5 316 000 рублей) на закупку компонентов системы и суммы зарплат всех участников проекта (1 065 440 рублей). Таким образом, конечная сумма денег, затребованная для полноценной реализации проекта, составляет 6 381 440 рублей.

В результате анализа создания контрольного стенда для массового предприятия было установлено, что, несмотря на значительные первоначальные инвестиции, внедрение такого оборудования оправдано благодаря повышению качества продукции и снижению уровня брака. Расчеты показали, что общая стоимость включает затраты на материалы, оборудование и монтаж, при этом правильная оценка этих расходов способствует эффективному управлению бюджетом проекта.

Заключение

В разделах работы решались поставленные в введении задачи по цели. Ключевыми моментами являлись глубокий анализ предоставленных данных, использование современных научных методов и средств научно-технического прогресса. На её основании, можно сделать соответствующие выводы о степени выполнения конечной цели.

В ходе данной работы была проведена всесторонняя оценка методов дефектоскопии и дефектометрии заготовок с использованием системы машинного зрения. Мы рассмотрели принципы работы современных технологий, позволяющих эффективно выявлять и классифицировать дефекты на различных этапах производства. В результате анализа было установлено, что системы машинного зрения обеспечивают высокую точность и скорость обработки данных, что значительно повышает качество контроля заготовок.

В результате, в первом разделе мы проанализировали на предприятие АвтоВАЗ техническую документацию о конструкции и технических характеристиках детали «Головка блока цилиндров». Узнали о способе производства и методике получения заготовки.

На основании этой информации решили также следующую задачу – определили появляющиеся в процессе заготовки дефекты поверхности. Как выяснилось, наиболее частыми дефектами являются трещины, пор, неспаи и другие, а также причины их появления и частые зоны появления. Причиной появления частого брака является отход от заданной технологии, несоблюдение нормативной документации производства при литье.

В следующем разделе были изучены использующиеся на сегодняшний день методы неразрушающего контроля, среди них, как самый удобный для решения задачи, был выбран оптический метод контроля. Ещё в данном разделе, мы выбрали соответствующее оборудование при этом методе – камеры компании BASLER.

С помощью соответствующего оборудования проведены серии экспериментов, заключающихся в сборе фотографий дефектной и нормальной поверхности алюминиевой заготовки. Далее они были определёнными методами обработаны для лучшей визуализации информации.

Вся эта информация собиралась с целью последующего обучения систем машинного обучения – нейронных сетей. Общие принципы всех нейросетей и частные свёрточного типа рассматривались в четвёртом разделе работы. Также здесь была выбрана нейронная сеть, которая была использована непосредственно. Из всего этого делался вывод о целесообразности использования данного метода дефектоскопии и дефектометрии.

В пятом разделе проводился комплексный синтез информации предыдущих разделов. Это позволило получить соответствующие параметры для проведения и организации разработки рабочего места для данного контрольного мероприятия на заводе. Автоматизированное рабочее место с использованием машинного зрения и обучения удовлетворяет всем потребностям производства АвтоВАЗ.

Далее проводятся расчёты полученного контрольного стенда на предмет его экономического преимущества перед существующим процессом, а также его экологическую безопасность. И тот и тот параметр оправдали себя при расчёте, значит можно говорить о корректности работы.

В заключение, можно отметить, что внедрение систем машинного зрения в дефектоскопию и дефектометрию открывает новые горизонты для повышения эффективности производства и улучшения качества продукции. Перспективы дальнейших исследований в этой области могут включать разработку более совершенных алгоритмов обработки изображений, а также расширение применения машинного обучения для повышения точности диагностики. Таким образом, использование технологий машинного зрения в дефектоскопии представляет собой важный шаг к оптимизации производственных процессов и достижению высоких стандартов качества.

Список используемых источников

1. Антипов И.Т. Развитие фотограмметрии в России // ГЕО-Сибирь-2010. VI Междунар. науч. конгр. : сб. материалов пленарного заседания. (Новосибирск, 19–29 апреля 2010 г.). Новосибирск: СГГА, 2010. 132 с.
2. Белошапкин М.А., Гук Н.А. Калибровка цифровых камер по линейному тест-объекту // ГЕО-Сибирь-2006. Междунар. науч. конгр.: сб. материалов в 6 т. (Новосибирск, 24–28 апреля 2006 г.). Новосибирск: СГГА, 2006. Т. 3, ч. 1. – С. 121.
3. Галушкин А.И. Нейронные сети / Большая российская энциклопедия в 35 т.; под. ред. Ю. С. Осипов, 2022. 118 с.
4. Горина Л.Н. Промышленная безопасность и производственный контроль [Электронный ресурс]: электрон. учеб. пособие / Л. Н. Горина, М. И. Фесина, Т. Ю. Фрезе; ТГУ; Ин-т машиностроения; каф. "Управление пром. и экол. безопасностью". URL: <https://dspace.tltsu.ru/handle/123456789/36> (дата обращения 18.03.2025).
5. ГОСТ 18353-79. Контроль неразрушающий. Классификация видов и методов. Общая часть: утв. Постановлением № 4245 от 01.07.1980.
6. ГОСТ Р 53696-2009. Контроль неразрушающий. Методы оптические. Термины и определения. М.: Стандартиформ, 2010. 7 с.
7. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео // Диалог-МИФИ. 2003. ISBN 5-86404-170-X.
8. Даюнова В.А., Трапезников А.В., Леонов А.А., Коренева Е.А. Модифицирование литейных алюминиевых сплавов (обзор) [Электронный ресурс]: Электронный научно-технический журнал «Труды ВИАМ». 2020. №4. URL: <http://www.viam-works.ru> (дата обращения: 03.12.2024).
9. Ессин А.С., Ессин С.С. Особенности фотограмметрической обработки материалов цифровой аэрофотосъемки с БПЛА // ГЕО-Сибирь-2010. VI

Междунар. науч. конгр.: сб. материалов в 6 т. (Новосибирск, 19–29 апреля 2010 г.). – Новосибирск: СГГА, 2010. Т. 4, ч. 1. С. 80–82.

10. Ессин А.С., Ессин С.С. Технология фотограмметрической обработки аэрофотоснимков, полученных с БПЛА, в целях создания ортофотопланов // ГЕО-Сибирь 2009. V Междунар. науч. конгр.: сб. материалов в 6 т. (Новосибирск, 20–24 апреля 2009 г.). Новосибирск: СГГА, 2009. Т. 4, ч. 1. С. 72–75.

11. Зубарев Ю.М. Расчет и проектирование приспособлений в машиностроении: учебник; Санкт-Петербург: Лань, 2015. 320 с.

12. Иванов В.П. Оборудование и оснастка промышленного предприятия: учеб. пособие / В. П. Иванов, А. В. Крыленко; Минск; Новое знание; Москва: ИНФРА-М, 2016. 235 с.

13. Кирсанов Э.Ю. Цифровые нейрокомпьютеры: Архитектура и схемотехника; под ред. А.И.Галушкина. Казань: Казанский Госуниверситет, 1995. 131 с.

14. Ключев В.В., Соснин Ф.Г. Теория и практика радиационного контроля: учеб. пособие для вузов. М.: Машиностроение. 1998. 170 с.

15. Краснопевцева И. В. Экономика и управление машиностроительным производством: электрон. учеб.-метод. пособие / ТГУ ; Ин-т финансов, экономики и управления ; каф. "Торговое дело и управление производством". - Тольятти: ТГУ, 2014. 183 с. URL <https://dspace.tltsu.ru/handle/123456789/13?mode=full> (дата обращения 20.04.2025).

16. Никитин К.В., И.Ю.Тимошкин. Влияние параметров обработки расплава на структуру и свойства литого алюминиевого припоя системы Al-Cu-Si // Литейщик России. 2009. № 6. С. 19.

17. Оборин Л.А., Чернов Н.М. Технологические основы процессов изготовления литых деталей для специального машиностроения. 2016. С. 348.

18. Павлов П.В., Горюнов А.Е. Аппаратно-программный комплекс неразрушающего контроля авиационных деталей (обзор) // Электронный

научно-технический журнал «Труды ВИАМ». № 80. 2023. URL: <https://cyberleninka.ru/article/n/apparatno-programmnyy-kompleks-nerazrushayuschego-kontrolya-aviatsionnyh-detaley/viewer> (дата обращения: 03.12.2024).

19. Сикорский, О.С. Обзор свёрточных нейронных сетей для задачи классификации изображений // Новые информационные технологии в автоматизированных системах – Москва, 2017. № 20. С. 37–42.

20. Федоров П. М. Охрана труда: практ. Пособие: РИОР: ИНФРА-М. 2-е изд. – Москва, 2017. 137 с.

21. Christian Szegedy, Sergey Loffe, Vincent Vanhoucke, Alex Alemi. 2016. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. arXiv preprint arXiv: 1602.07261.

22. Keiron O'Shea, Ryan Nash. AnIntroduction to Convolutional Neural Networks. 2015 - (11). Ст. 04. URL: <https://archive.org/details/arxiv-1511.08458>. (дата обращения: 01.10.2024).

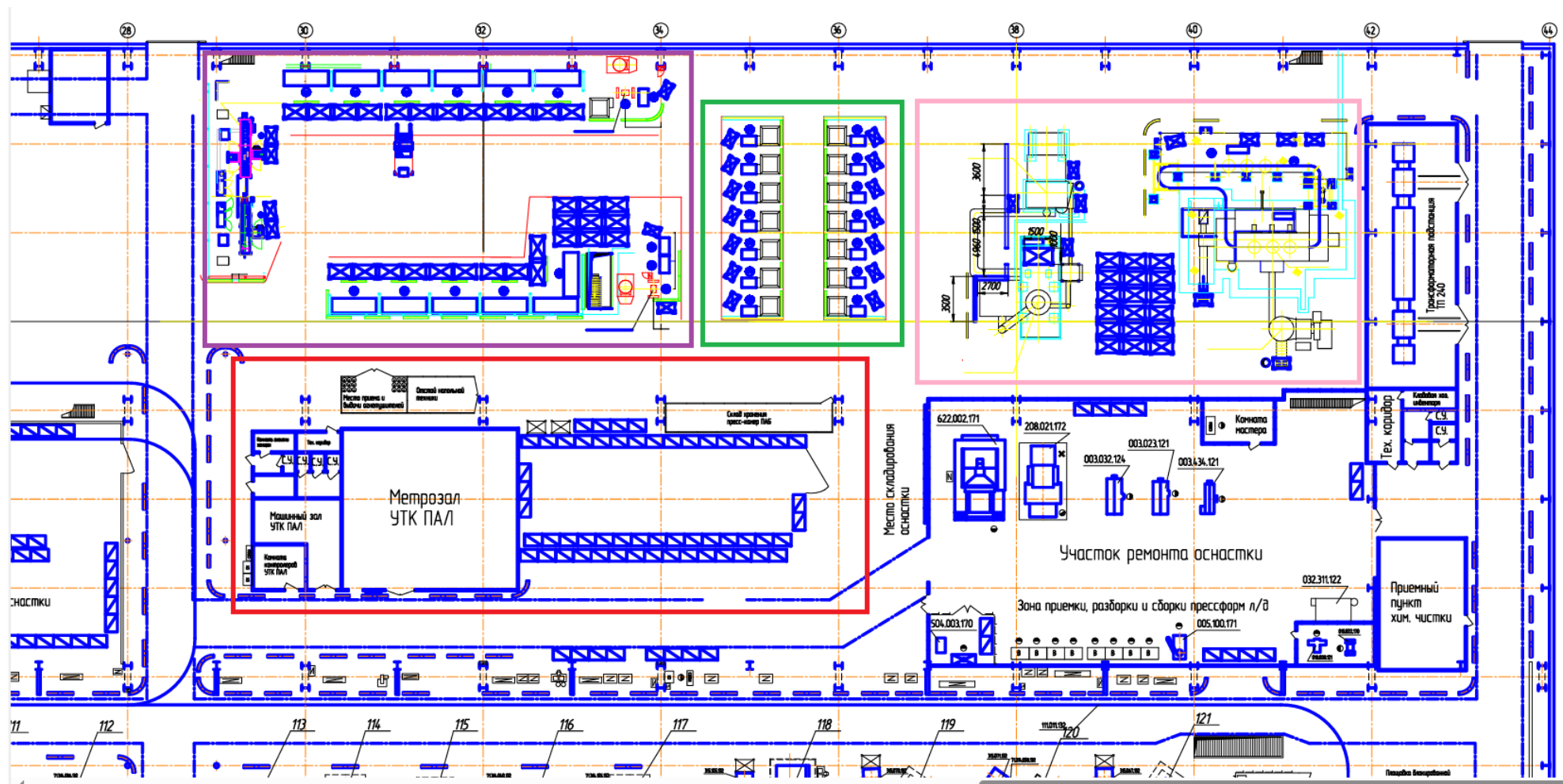
23. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation. – 1989. - 1(4), 541–551. URL: <https://ieeexplore.ieee.org/document/6795724>. (дата обращения: 11.02.2025).

24. Zeiler, M.D., Fergus, R.: Stochastic pooling for regularization of deep convolutional neural networks. arXiv preprint arXiv:1301.3557 (2013). URL: <https://arxiv.org/abs/1301.3557>. (дата обращения: 26.03.2025).

25. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Computer Vision–ECCV 2014, pp. 818–833. Springer (2014).

Приложение А

Участок производственного корпуса 21



Приложение Б

Калибровка

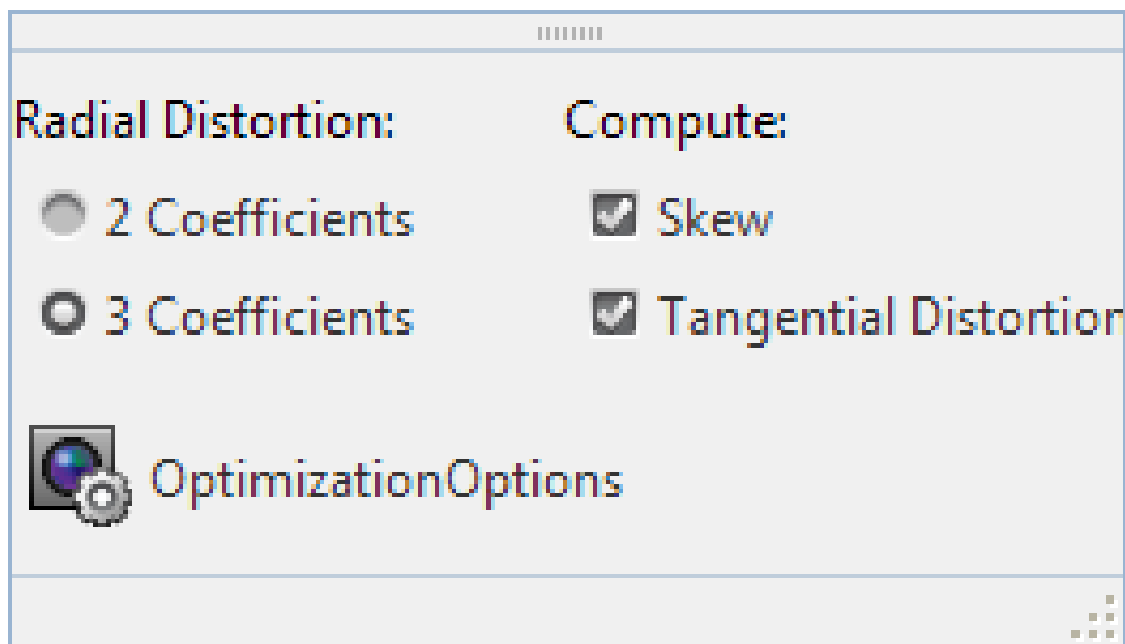


Рисунок А.1 - Параметры работы функции Calibration

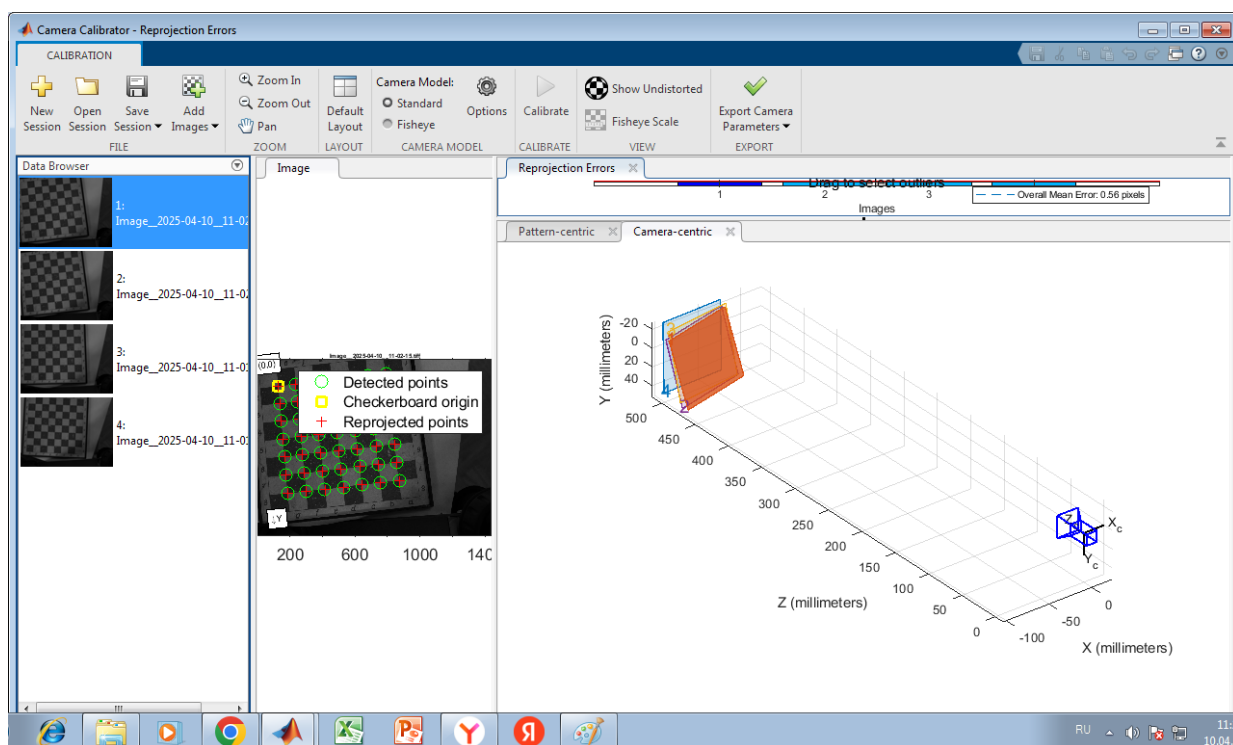


Рисунок А.2 - Изображение построенного расстояния от объектива камеры до поверхности калибровочной доски

Продолжение приложения Б

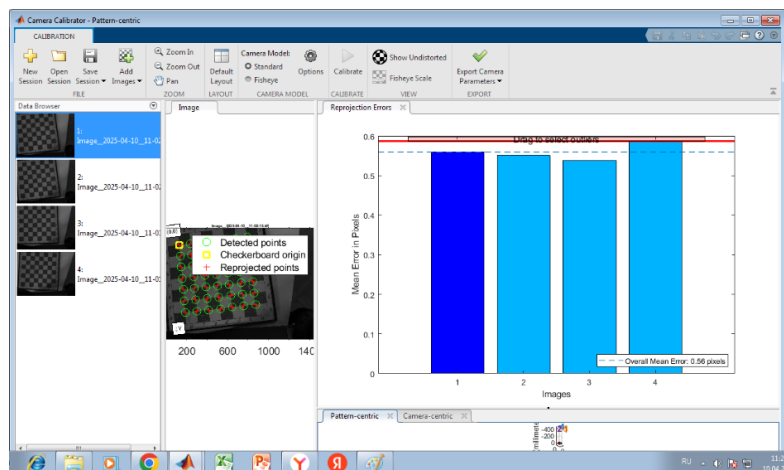


Рисунок А.3 - Среднее отклонение точек сетчатой структуры от истинного положения

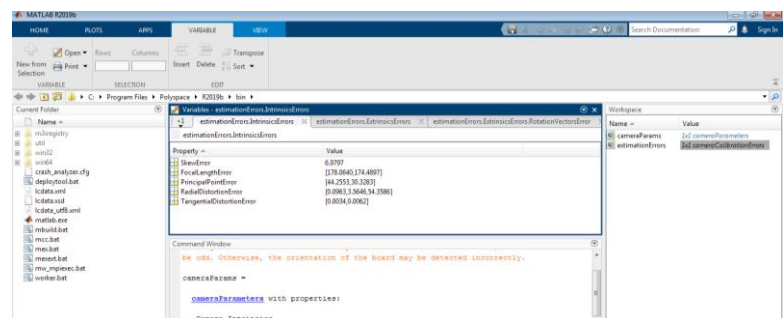


Рисунок А.4 - Получаемые параметры работы функции Camera Calibration

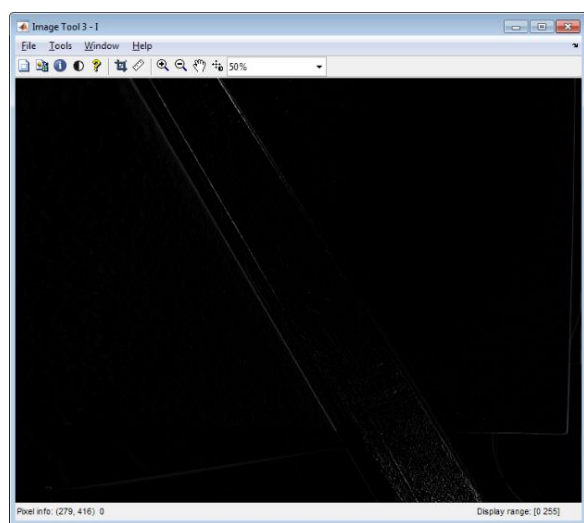


Рисунок А.5 - Пример вычитания исходного и истинного изображения поверхности детали

Продолжение приложения Б

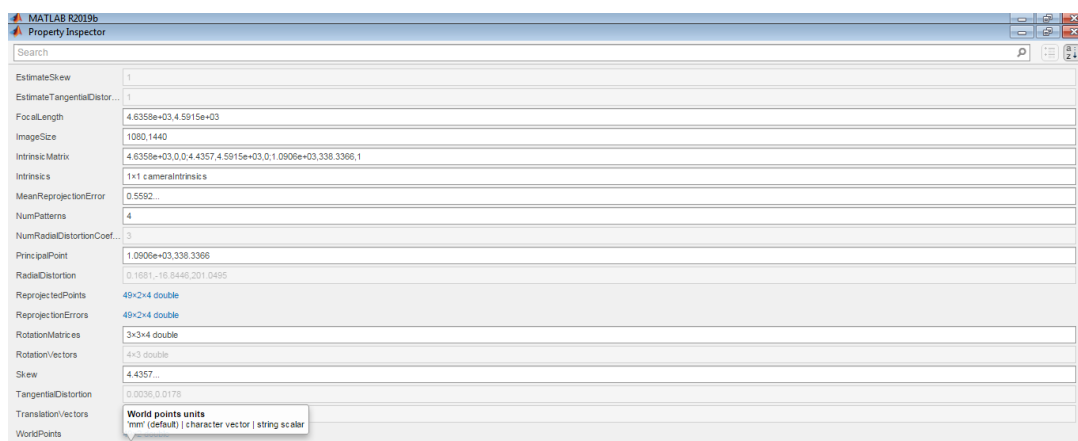


Рисунок А.6 - Получаемые параметры работы функции Camera Calibration

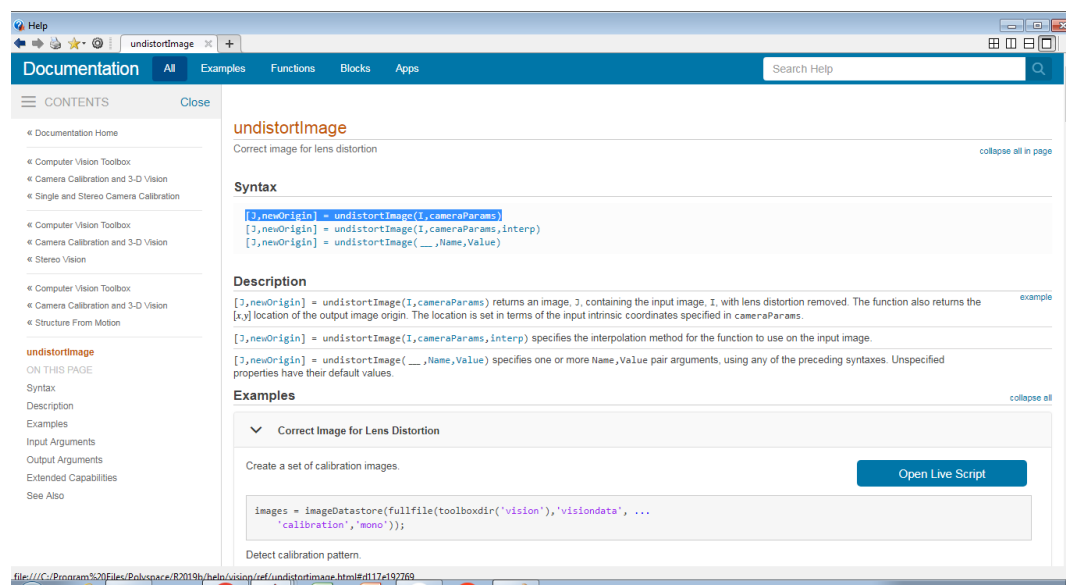


Рисунок А.7 - Справка работы функции

Приложение В

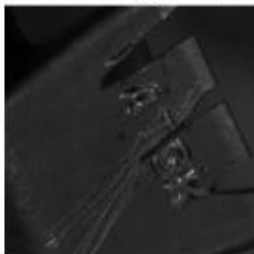
Результаты машинного обучения 2-х классового исходного кода

```
print(" .class_indices:", train_generator.class_indices) # Словарь сопоставления имен классов с индексами
print(" .samples:", train_generator.samples) # Общее количество образцов
print(" .batch_size:", train_generator.batch_size) # Размер пакета
print(" .target_size:", train_generator.target_size) # Целевой размер изображения
print(" .color_mode:", train_generator.color_mode) # Цветовой режим
print(" .class_mode:", train_generator.class_mode) # Режим классов
print(" .shuffle:", train_generator.shuffle) # Перемешивание
```

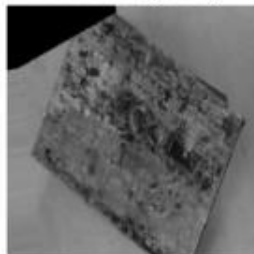
```
Found 2238 images belonging to 2 classes.
Found 2238 images belonging to 2 classes.
Классы: {'all': 0, 'noall': 1}
Размер пакета изображений (data_batch.shape): (16, 400, 400, 1)
Размер пакета меток (labels_batch.shape): (16, 2)
Тип данных изображений (data_batch.dtype): float32
Тип данных меток (labels_batch.dtype): float32
Метки в пакете (labels_batch): [[0. 1.]
 [1. 0.]
 [1. 0.]
 [1. 0.]
 [0. 1.]
 [1. 0.]
 [1. 0.]
 [1. 0.]
 [0. 1.]
 [1. 0.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [1. 0.]
 [1. 0.]
 [0. 1.]
 [1. 0.]]
```



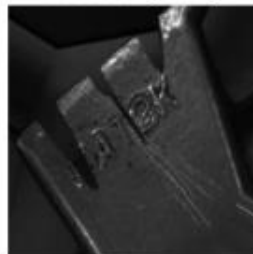
Метка: [1. 0.]



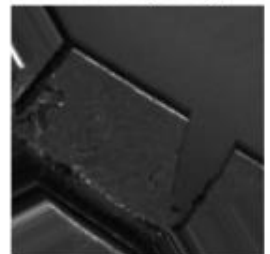
Метка: [1. 0.]



Метка: [1. 0.]



Метка: [1. 0.]



```
[ ] import matplotlib.pyplot as plt
import matplotlib.image as mpimg
image_path = '/content/train/ủ%ấầ-ấầ/Image__2025-02-27__11-11-19.tiff'

# Загрузите изображение
img = mpimg.imread(image_path)

# Выведите изображение
plt.imshow(img)
plt.axis('off') # Отключить оси
plt.show()
```


Продолжение приложения В

```
x = Dense(64, activation="relu")(x)

# Выход: [x, y, width, height, class_id]
output = Dense(2, activation="softmax")(x)
model = tf.keras.Model(inputs, output)
model.compile(optimizer="adam", loss="categorical_crossentropy", metrics=["accuracy"])
model.summary()
```

Model: "functional_9"

Layer (type)	Output Shape	Param #
input_layer_6 (InputLayer)	(None, 400, 400, 1)	0
conv2d_8 (Conv2D)	(None, 400, 400, 32)	320
max_pooling2d_8 (MaxPooling2D)	(None, 200, 200, 32)	0
conv2d_9 (Conv2D)	(None, 200, 200, 64)	18,496
max_pooling2d_9 (MaxPooling2D)	(None, 100, 100, 64)	0
flatten_5 (Flatten)	(None, 640000)	0
dense_9 (Dense)	(None, 64)	40,960,064
dense_10 (Dense)	(None, 2)	130

Total params: 40,979,010 (156.32 MB)

Trainable params: 40,979,010 (156.32 MB)

Non-trainable params: 0 (0.00 B)

```
Epoch 1/20: 64.216s/step - accuracy: 0.708 - loss: 2.8956000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 2/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 3/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 4/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 5/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 6/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 7/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 8/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 9/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 10/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 11/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 12/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 13/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 14/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 15/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 16/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 17/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 18/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 19/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
Epoch 20/20: 64.188s/step - accuracy: 0.878 - loss: 0.35760000000000004 - val_accuracy: 0.808 - val_loss: 0.352
```

Precision per class: [0.90066225 0.92592593]

Recall per class: [0.98076923 0.68965517]

F1-score per class: [0.93901036 0.79051383]

Confusion Matrix:

```
[[408  8]
 [ 45 100]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.90	0.98	0.94	416
1	0.93	0.69	0.79	145
accuracy			0.91	561
macro avg	0.91	0.84	0.86	561
weighted avg	0.91	0.91	0.90	561

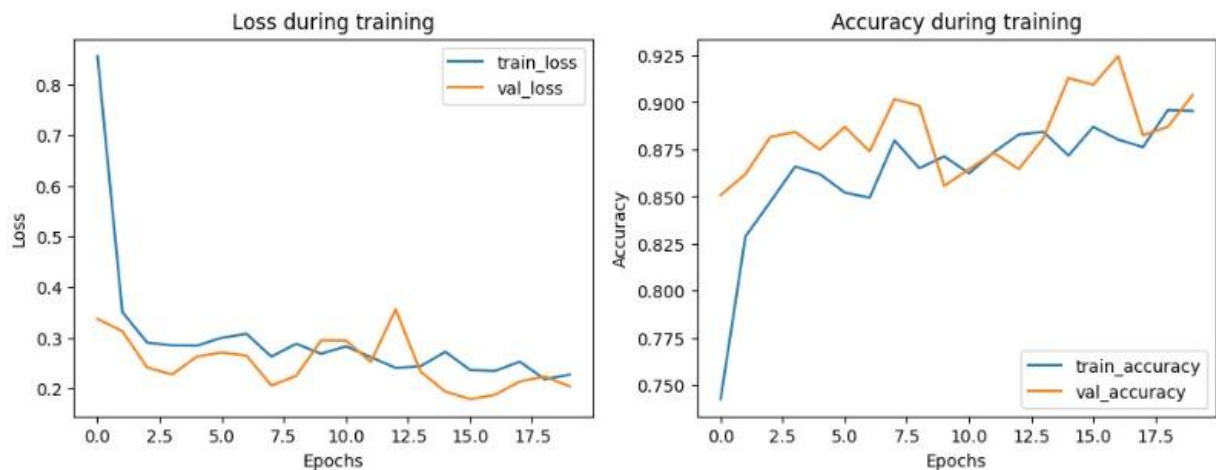
Продолжение приложения В

```
[ ] print(f'F1 Score: {f1}')
    print('Confusion Matrix:')
    print(conf_matrix)
    print('Classification Report:')
    print(class_report)
```

```
Accuracy: 0.9055258467023173
Precision: [0.90066225 0.92592593]
Recall: [0.98076923 0.68965517]
F1 Score: [0.93901036 0.79051383]
Confusion Matrix:
[[408  8]
 [ 45 100]]
Classification Report:

```

	precision	recall	f1-score	support
0	0.90	0.98	0.94	416
1	0.93	0.69	0.79	145
accuracy			0.91	561
macro avg	0.91	0.84	0.86	561
weighted avg	0.91	0.91	0.90	561



```
# Пример использования:
train_input_dir = "/content/train" # Замените на путь к вашей папке train
train_output_dir = "/content/train_rgb_images" # Замените на путь, куда сохранить train RGB
test_input_dir = "/content/test" # Замените на путь к вашей папке test
test_output_dir = "/content/test_rgb_images" # Замените на путь, куда сохранить test RGB

grayscale_to_rgb(train_input_dir, train_output_dir)
grayscale_to_rgb(test_input_dir, test_output_dir)

print("Преобразование завершено.")
```

```
Найдены классы: ['noall', 'all']
Обработка класса: noall
Преобразование изображений в noall: 100%|██████████| 578/578 [00:01<00:00, 515.09it/s]
Обработка класса: all
Преобразование изображений в all: 100%|██████████| 1660/1660 [00:04<00:00, 391.89it/s]
Найдены классы: ['noall', 'all']
Обработка класса: noall
Преобразование изображений в noall: 100%|██████████| 145/145 [00:00<00:00, 368.55it/s]
Обработка класса: all
Преобразование изображений в all: 100%|██████████| 416/416 [00:01<00:00, 282.16it/s]Преобразование завершено.
```

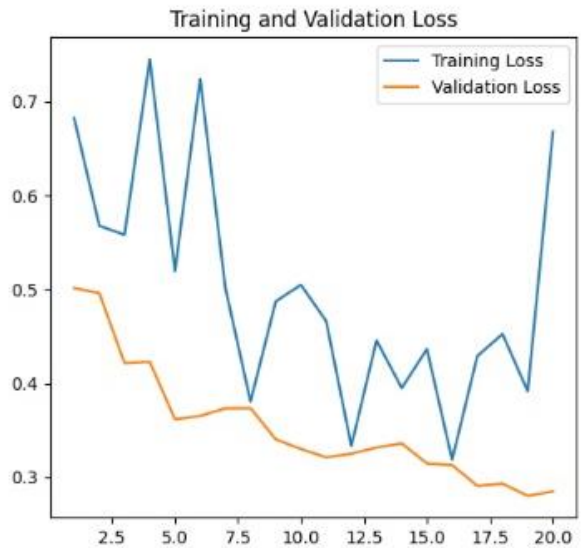
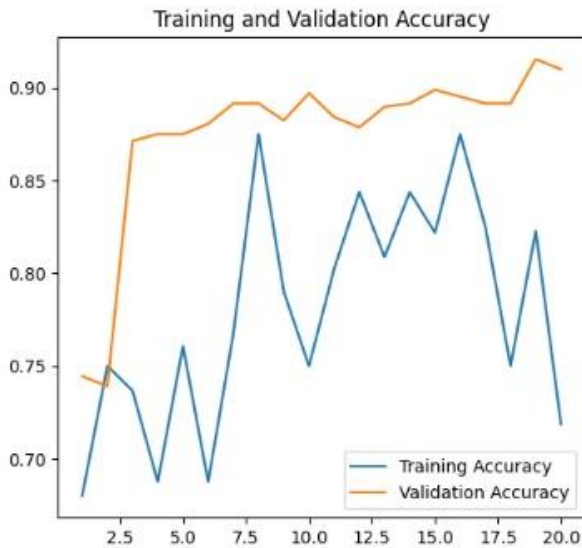
Продолжение приложения В

```
Классы: {'noall': 0, 'all': 1}
Размер пакета изображений (data_batch.shape): (32, 224, 224, 3)
Размер пакета меток (labels_batch.shape): (32, 2)
Тип данных изображений (data_batch.dtype): float32
Тип данных меток (labels_batch.dtype): float32
Метки в пакете (labels_batch): [[0. 1.]
```

```

/home/ai120/python.11/dist/packages/tensorflow/trainers/data_adapter.py:dataset_adapter.py:121: UserWarning: Your 'PyDataset' class should call super().__init__(**kwargs) in its constructor. **kwargs can include: 'workers', 'use_multiprocessing', 'max_queue_size'. Do not pass those arguments to '__init__', as they will be ignored.
self._warn_if_super_not_called()
Epoch 1/20
4000 41 565m/step - accuracy: 0.8568 - loss: 0.7888 - val_accuracy: 0.7465 - val_loss: 0.5813
Epoch 2/20
4000 131 194m/step - accuracy: 0.7888 - loss: 0.5670 - val_accuracy: 0.8121 - dist-packages/tensorflow/trainers/epoch_iterator.py:180: UserWarning: Your input ran out of data; interrupting training. Make sure that your dataset or generator can generate at least 'steps_per_epoch * epochs' batches. You
4000 40 41m/step - accuracy: 0.7888 - loss: 0.5478 - val_accuracy: 0.7708 - val_loss: 0.4959
Epoch 3/20
4000 41 552m/step - accuracy: 0.7384 - loss: 0.5682 - val_accuracy: 0.8713 - val_loss: 0.4214
Epoch 4/20
4000 5 72m/step - accuracy: 0.6875 - loss: 0.7462 - val_accuracy: 0.8758 - val_loss: 0.4228
Epoch 5/20
4000 41 595m/step - accuracy: 0.7483 - loss: 0.5395 - val_accuracy: 0.8758 - val_loss: 0.3816
Epoch 6/20
4000 41 51m/step - accuracy: 0.6875 - loss: 0.7241 - val_accuracy: 0.8805 - val_loss: 0.3858
Epoch 7/20
4000 384 512m/step - accuracy: 0.7616 - loss: 0.5139 - val_accuracy: 0.8915 - val_loss: 0.3732
Epoch 8/20
4000 41 52m/step - accuracy: 0.8758 - loss: 0.3884 - val_accuracy: 0.8915 - val_loss: 0.3754
Epoch 9/20
384 408m/step - accuracy: 0.7917 - loss: 0.4891 - val_accuracy: 0.8824 - val_loss: 0.3462
Epoch 10/20
4000 5 72m/step - accuracy: 0.7588 - loss: 0.5587 - val_accuracy: 0.8971 - val_loss: 0.3299
Epoch 11/20
4000 41 534m/step - accuracy: 0.7989 - loss: 0.4878 - val_accuracy: 0.8842 - val_loss: 0.3218
Epoch 12/20
4000 5 77m/step - accuracy: 0.8418 - loss: 0.3331 - val_accuracy: 0.8797 - val_loss: 0.3249
Epoch 13/20
780 409m/step - accuracy: 0.8898 - loss: 0.4278 - val_accuracy: 0.8897 - val_loss: 0.3134
Epoch 14/20
4000 5 72m/step - accuracy: 0.8418 - loss: 0.3548 - val_accuracy: 0.8915 - val_loss: 0.3109
Epoch 15/20
4000 41 574m/step - accuracy: 0.8384 - loss: 0.4488 - val_accuracy: 0.8989 - val_loss: 0.3145
Epoch 16/20
4000 41 52m/step - accuracy: 0.8758 - loss: 0.3181 - val_accuracy: 0.8915 - val_loss: 0.3128
Epoch 17/20
796 588m/step - accuracy: 0.8231 - loss: 0.4187 - val_accuracy: 0.8915 - val_loss: 0.2986
Epoch 18/20
4000 41 52m/step - accuracy: 0.7988 - loss: 0.4524 - val_accuracy: 0.8915 - val_loss: 0.2998
Epoch 19/20
4000 41 550m/step - accuracy: 0.8145 - loss: 0.3989 - val_accuracy: 0.9104 - val_loss: 0.2988
Epoch 20/20
4000 5 72m/step - accuracy: 0.7288 - loss: 0.4688 - val_accuracy: 0.9099 - val_loss: 0.2847
OSError: Incomplete

```



Продолжение приложения В

```
Precision per class: [0.07894737 0.08948546]
Recall per class: [0.02163462 0.27586207]
F1-score per class: [0.03396226 0.13513514]
Confusion Matrix:
[[ 9 407]
 [105 40]]
Classification Report:
      precision    recall  f1-score   support

     0       0.08      0.02      0.03       416
     1       0.09      0.28      0.14       145

 accuracy          0.09       561
 macro avg       0.08      0.15      0.08       561
 weighted avg    0.08      0.09      0.06       561
```

```
print(f'Accuracy: {accuracy}')
print(f'Precision: {precision}')
print(f'Recall: {recall}')
print(f'F1 Score: {f1}')
print('Confusion Matrix:')
print(conf_matrix)
print('Classification Report:')
print(class_report)
```

```
Accuracy: 0.3
Precision: [0.75 0. 0. 0. ]
Recall: [1. 0. 0. 0.]
F1 Score: [0.85714286 0. 0. 0. ]
Confusion Matrix:
[[3 0 0 0]
 [1 0 2 0]
 [0 0 0 2]
 [0 2 0 0]]
Classification Report:
      precision    recall  f1-score   support

     0       0.75      1.00      0.86         3
     1       0.00      0.00      0.00         3
     2       0.00      0.00      0.00         2
     3       0.00      0.00      0.00         2

 accuracy          0.30        10
 macro avg       0.19      0.25      0.21        10
 weighted avg    0.23      0.30      0.26        10
```

```
[ ]
# ОТЧЕТ
print("Classification Report:")
print(classification_report(y_true, y_pred, target_names=class_labels))
Classification Report:
      precision    recall  f1-score   support

    Car       0.58      0.57      0.58       584
    Truck     0.43      0.44      0.43       425

 accuracy          0.52      1009
 macro avg       0.51      0.51      0.50      1009
 weighted avg    0.52      0.52      0.52      1009
```

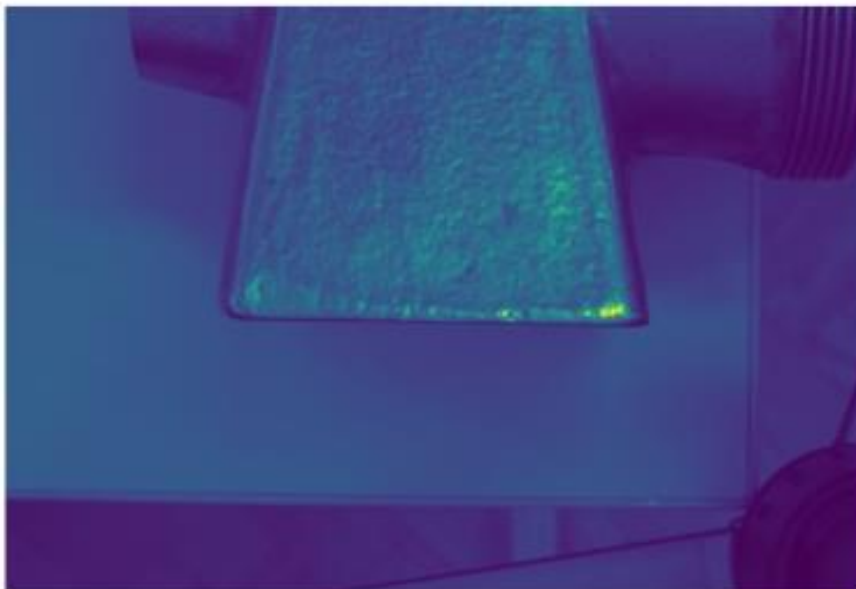

Приложение Г

Результаты машинного обучения 2-х классового кода на предобученной нейронной сети

```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
image_path = '/content/train/ủấầ-án/Image_2025-02-27__11-11-19.tiff'

# Загрузите изображение
img = mpimg.imread(image_path)

# Выведите изображение
plt.imshow(img)
plt.axis('off') # Отключить оси
plt.show()
```



```
# Пример использования:
train_input_dir = "/content/train" # Замените на путь к вашей папке train
train_output_dir = "/content/train_rgb_images" # Замените на путь, куда сохранить train RGB
test_input_dir = "/content/test" # Замените на путь к вашей папке test
test_output_dir = "/content/test_rgb_images" # Замените на путь, куда сохранить test RGB

grayscale_to_rgb(train_input_dir, train_output_dir)
grayscale_to_rgb(test_input_dir, test_output_dir)

print("Преобразование завершено.")
```

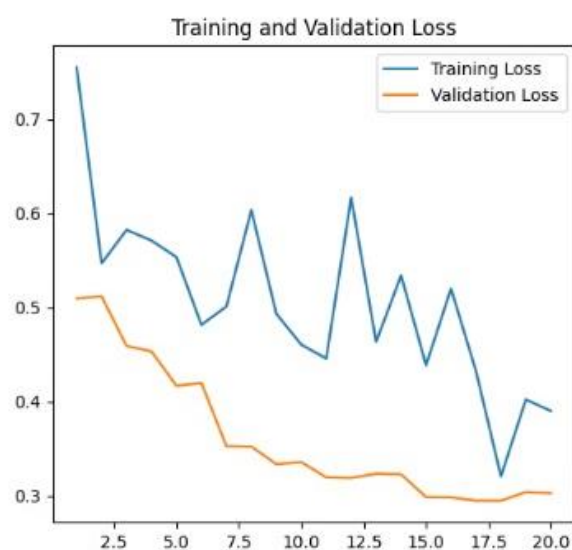
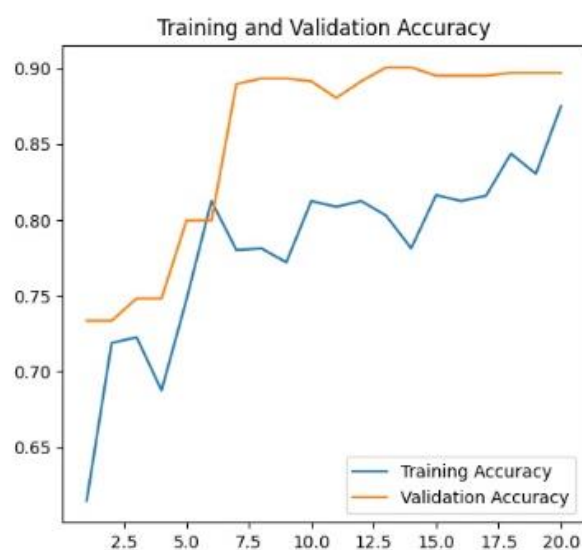
```
найлены классы: ['noall', 'all']
Обработка класса: noall
Преобразование изображений в noall: 100%|██████████| 578/578 [00:01<00:00, 491.37it/s]
Обработка класса: all
Преобразование изображений в all: 100%|██████████| 1660/1660 [00:04<00:00, 397.33it/s]
найлены классы: ['noall', 'all']
Обработка класса: noall
Преобразование изображений в noall: 100%|██████████| 145/145 [00:00<00:00, 398.53it/s]
Обработка класса: all
Преобразование изображений в all: 100%|██████████| 416/416 [00:01<00:00, 400.45it/s]Преобразование завершено.
```

Продолжение приложения Г

```
Классы: {'noall': 0, 'all': 1}
Размер пакета изображений (data_batch.shape): (32, 224, 224, 3)
Размер пакета меток (labels_batch.shape): (32, 2)
Тип данных изображений (data_batch.dtype): float32
Тип данных меток (labels_batch.dtype): float32
Метки в пакете (labels_batch): [[0. 1.]
 [1. 0.]
 [0. 1.]
 [0. 1.]
 [1. 0.]
 [1. 0.]
 [0. 1.]
 [0. 1.]
 [1. 0.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [1. 0.]
 [1. 0.]
 [1. 0.]
 [0. 1.]
 [1. 0.]
 [1. 0.]
 [1. 0.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [1. 0.]
 [0. 1.]
 [0. 1.]
 [0. 1.]
 [1. 0.]
 [0. 1.]
 [0. 1.]]

/usr/local/lib/python3.11/dist-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:121: UserWarning: Your 'PyDataset' class should call self._warn_if_super_not_called()
  self._warn_if_super_not_called()
Epoch 1/20 ----- 46s 612ms/step - accuracy: 0.5859 - loss: 0.8782 - val_accuracy: 0.7335 - val_loss: 0.5097
Epoch 2/20 ----- 12s 179ms/step - accuracy: 0.7188 - loss: 0.5470 - val_accuracy: 0.7335 - val_loss: 0.5120
1/69 ----- 12s 179ms/step - accuracy: 0.7188 - loss: 0.5470 - val_accuracy: 0.7335 - val_loss: 0.5120
self._interrupted_warning()
Epoch 3/20 ----- 3s 48ms/step - accuracy: 0.7188 - loss: 0.5470 - val_accuracy: 0.7335 - val_loss: 0.5120
Epoch 4/20 ----- 82s 696ms/step - accuracy: 0.7146 - loss: 0.6015 - val_accuracy: 0.7482 - val_loss: 0.4592
Epoch 5/20 ----- 3s 48ms/step - accuracy: 0.6875 - loss: 0.5713 - val_accuracy: 0.7482 - val_loss: 0.4535
Epoch 6/20 ----- 67s 520ms/step - accuracy: 0.7466 - loss: 0.5457 - val_accuracy: 0.7996 - val_loss: 0.4168
Epoch 7/20 ----- 3s 49ms/step - accuracy: 0.8125 - loss: 0.4817 - val_accuracy: 0.7996 - val_loss: 0.4198
Epoch 8/20 ----- 41s 595ms/step - accuracy: 0.7721 - loss: 0.5137 - val_accuracy: 0.8897 - val_loss: 0.3528
Epoch 9/20 ----- 5s 71ms/step - accuracy: 0.7812 - loss: 0.6036 - val_accuracy: 0.8934 - val_loss: 0.3523
Epoch 10/20 ----- 41s 594ms/step - accuracy: 0.7597 - loss: 0.5077 - val_accuracy: 0.8934 - val_loss: 0.3335
Epoch 11/20 ----- 3s 48ms/step - accuracy: 0.8125 - loss: 0.4686 - val_accuracy: 0.8915 - val_loss: 0.3357
Epoch 12/20 ----- 82s 646ms/step - accuracy: 0.8046 - loss: 0.4523 - val_accuracy: 0.8885 - val_loss: 0.3195
Epoch 13/20 ----- 3s 49ms/step - accuracy: 0.8125 - loss: 0.6170 - val_accuracy: 0.8915 - val_loss: 0.3188
Epoch 14/20 ----- 82s 696ms/step - accuracy: 0.8884 - loss: 0.4644 - val_accuracy: 0.9087 - val_loss: 0.3233
Epoch 15/20 ----- 5s 72ms/step - accuracy: 0.7812 - loss: 0.5342 - val_accuracy: 0.9087 - val_loss: 0.3227
Epoch 16/20 ----- 38s 543ms/step - accuracy: 0.8184 - loss: 0.4476 - val_accuracy: 0.8952 - val_loss: 0.2985
Epoch 17/20 ----- 4s 51ms/step - accuracy: 0.8125 - loss: 0.5202 - val_accuracy: 0.8952 - val_loss: 0.2982
Epoch 18/20 ----- 41s 594ms/step - accuracy: 0.8090 - loss: 0.4466 - val_accuracy: 0.8952 - val_loss: 0.2947
Epoch 19/20 ----- 4s 51ms/step - accuracy: 0.8438 - loss: 0.3206 - val_accuracy: 0.8971 - val_loss: 0.2946
Epoch 20/20 ----- 74s 536ms/step - accuracy: 0.8417 - loss: 0.3992 - val_accuracy: 0.8971 - val_loss: 0.3038
Epoch 21/20 ----- 4s 49ms/step - accuracy: 0.8750 - loss: 0.3080 - val_accuracy: 0.8971 - val_loss: 0.3027
18/18 ----- 5s 215ms/step
Accuracy: 0.8983957219251337
```

Продолжение приложения Г



```

true_classes = valid_generator.classes
class_labels = list(valid_generator.class_indices.keys())
print(f"Accuracy: {accuracy_score(true_classes, predicted_classes)}")
print(f'Precision: {precision}')
print(f'Recall: {recall}')
print(f'F1 Score: {f1}')
print('Confusion Matrix:')
print(conf_matrix)
print('Classification Report:')
print(class_report)

```

18/18 ————— 3s 180ms/step

Accuracy: 0.8983957219251337

Precision: [0.81428571 0.9263658]

Recall: [0.7862069 0.9375]

F1 Score: [0.8 0.93189964]

Confusion Matrix:

[[114 31]

[26 390]]

Classification Report:

	precision	recall	f1-score	support
0	0.81	0.79	0.80	145
1	0.93	0.94	0.93	416
accuracy			0.90	561
macro avg	0.87	0.86	0.87	561
weighted avg	0.90	0.90	0.90	561

Продолжение приложения Г

<pre> dictionary = model_fit(train_generator, steps_per_epoch=steps_per_epoch, epochs=60, validation_data=validation_generator, validation_steps=validation_steps) </pre>						
Epoch 1/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 1.0012	val_accuracy: 0.9992	val_loss: 0.9978
Epoch 2/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.9774	val_accuracy: 0.9994	val_loss: 0.9758
Epoch 3/60	128/128	0% 128ms/step	accuracy: 0.9992	loss: 0.9598	val_accuracy: 0.9992	val_loss: 0.9587
Epoch 4/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.9397	val_accuracy: 0.9998	val_loss: 0.9378
Epoch 5/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.9198	val_accuracy: 0.9998	val_loss: 0.9178
Epoch 6/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.8998	val_accuracy: 0.9998	val_loss: 0.8978
Epoch 7/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.8798	val_accuracy: 0.9998	val_loss: 0.8778
Epoch 8/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.8598	val_accuracy: 0.9998	val_loss: 0.8578
Epoch 9/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.8398	val_accuracy: 0.9998	val_loss: 0.8378
Epoch 10/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.8198	val_accuracy: 0.9998	val_loss: 0.8178
Epoch 11/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.7998	val_accuracy: 0.9998	val_loss: 0.7978
Epoch 12/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.7798	val_accuracy: 0.9998	val_loss: 0.7778
Epoch 13/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.7598	val_accuracy: 0.9998	val_loss: 0.7578
Epoch 14/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.7398	val_accuracy: 0.9998	val_loss: 0.7378
Epoch 15/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.7198	val_accuracy: 0.9998	val_loss: 0.7178
Epoch 16/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.6998	val_accuracy: 0.9998	val_loss: 0.6978
Epoch 17/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.6798	val_accuracy: 0.9998	val_loss: 0.6778
Epoch 18/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.6598	val_accuracy: 0.9998	val_loss: 0.6578
Epoch 19/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.6398	val_accuracy: 0.9998	val_loss: 0.6378
Epoch 20/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.6198	val_accuracy: 0.9998	val_loss: 0.6178
Epoch 21/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.5998	val_accuracy: 0.9998	val_loss: 0.5978
Epoch 22/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.5798	val_accuracy: 0.9998	val_loss: 0.5778
Epoch 23/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.5598	val_accuracy: 0.9998	val_loss: 0.5578
Epoch 24/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.5398	val_accuracy: 0.9998	val_loss: 0.5378
Epoch 25/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.5198	val_accuracy: 0.9998	val_loss: 0.5178
Epoch 26/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.4998	val_accuracy: 0.9998	val_loss: 0.4978
Epoch 27/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.4798	val_accuracy: 0.9998	val_loss: 0.4778
Epoch 28/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.4598	val_accuracy: 0.9998	val_loss: 0.4578
Epoch 29/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.4398	val_accuracy: 0.9998	val_loss: 0.4378
Epoch 30/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.4198	val_accuracy: 0.9998	val_loss: 0.4178
Epoch 31/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.3998	val_accuracy: 0.9998	val_loss: 0.3978
Epoch 32/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.3798	val_accuracy: 0.9998	val_loss: 0.3778
Epoch 33/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.3598	val_accuracy: 0.9998	val_loss: 0.3578
Epoch 34/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.3398	val_accuracy: 0.9998	val_loss: 0.3378
Epoch 35/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.3198	val_accuracy: 0.9998	val_loss: 0.3178
Epoch 36/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.2998	val_accuracy: 0.9998	val_loss: 0.2978
Epoch 37/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.2798	val_accuracy: 0.9998	val_loss: 0.2778
Epoch 38/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.2598	val_accuracy: 0.9998	val_loss: 0.2578
Epoch 39/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.2398	val_accuracy: 0.9998	val_loss: 0.2378
Epoch 40/60	128/128	0% 128ms/step	accuracy: 0.9998	loss: 0.2198	val_accuracy: 0.9998	val_loss:

Продолжение приложения Г

```
plt.figure(figsize=(6, 6))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=class_labels, yticklabels=class_labels)
plt.xlabel('Предсказанные классы')
plt.ylabel('Истинные классы')
plt.title('Confusion Matrix')
plt.show()
```

[]

```
# Отчёт
print("Classification Report:")
print(classification_report(y_true, y_pred, target_names=class_labels))
```

Classification Report:

	precision	recall	f1-score	support
Car	0.58	0.57	0.58	584
Truck	0.43	0.44	0.43	425
accuracy			0.52	1009
macro avg	0.51	0.51	0.50	1009
weighted avg	0.52	0.52	0.52	1009

Приложение Д

Результаты работы машинного кода 5-классового

Копия_блокнота__farsh_patterns_full3_ipynb_ipy...

Файл Изменить Вид Вставка Среда выполнения Инструменты Справка

Команды + Код + Текст

▶

```
print(f" Тренировочных изображений: {len(train_images)}")
print(f" Тестовых изображений: {len(test_images)}")

print("Разделение данных завершено.")
```

↗

```
Обнаружены классы: ['nodefects', 'attrition', 'pores', 'scratch', 'label']
Класс: nodefects
Всего изображений: 30
Тренировочных изображений: 24
Тестовых изображений: 6
Класс: attrition
Всего изображений: 791
Тренировочных изображений: 632
Тестовых изображений: 159
Класс: pores
Всего изображений: 360
Тренировочных изображений: 288
Тестовых изображений: 72
Класс: scratch
Всего изображений: 451
Тренировочных изображений: 360
Тестовых изображений: 91
Класс: label
Всего изображений: 474
Тренировочных изображений: 379
Тестовых изображений: 95
Разделение данных завершено.
```

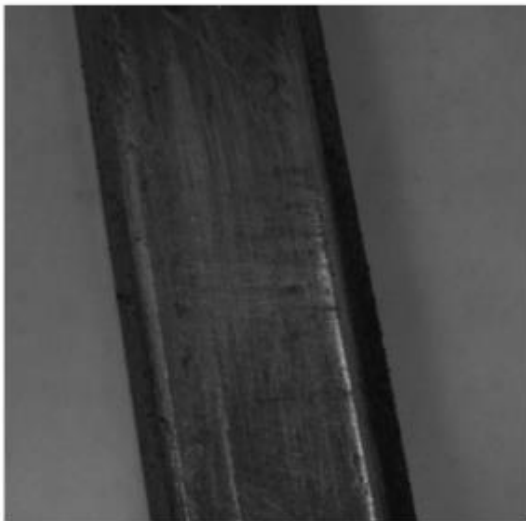
▶

```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
image_path = '/content/train/attrition/def11.jpg'

# Загрузите изображение
img = mpimg.imread(image_path)

# Выведите изображение
plt.imshow(img)
plt.axis('off') # Отключить оси
plt.show()
```

↗



Продолжение приложения Д

```
train_input_dir = "/content/train" # Замените на путь к вашей папке train
train_output_dir = "/content/train_rgb_images" # Замените на путь, куда сохранить train RGB
test_input_dir = "/content/test" # Замените на путь к вашей папке test
test_output_dir = "/content/test_rgb_images" # Замените на путь, куда сохранить test RGB

grayscale_to_rgb(train_input_dir, train_output_dir)
grayscale_to_rgb(test_input_dir, test_output_dir)

print("Преобразование завершено.")
```

```
Найдены классы: ['nodefects', 'attrition', 'pores', 'scratch', 'label']
Обработка класса: nodefects
Преобразование изображений в nodefects: 100%|██████████| 24/24 [00:01<00:00, 14.42it/s]
Обработка класса: attrition
Преобразование изображений в attrition: 100%|██████████| 632/632 [00:01<00:00, 347.60it/s]
Обработка класса: pores
Преобразование изображений в pores: 100%|██████████| 288/288 [00:00<00:00, 612.12it/s]
Обработка класса: scratch
Преобразование изображений в scratch: 100%|██████████| 360/360 [00:00<00:00, 511.37it/s]
Обработка класса: label
Преобразование изображений в label: 100%|██████████| 379/379 [00:01<00:00, 335.09it/s]
Найдены классы: ['nodefects', 'attrition', 'pores', 'scratch', 'label']
Обработка класса: nodefects
Преобразование изображений в nodefects: 100%|██████████| 6/6 [00:00<00:00, 15.12it/s]
Обработка класса: attrition
Преобразование изображений в attrition: 100%|██████████| 159/159 [00:00<00:00, 341.93it/s]
Обработка класса: pores
Преобразование изображений в pores: 100%|██████████| 72/72 [00:00<00:00, 578.97it/s]
Обработка класса: scratch
Преобразование изображений в scratch: 100%|██████████| 91/91 [00:00<00:00, 503.95it/s]
Обработка класса: label
Преобразование изображений в label: 100%|██████████| 95/95 [00:00<00:00, 372.95it/s] Преобразование завершено.
```

Model: "mobilenetv2_1.0b_224"

Layer (type)	Output Shape	Param #	Connected to
input_layer_2 (InputLayer)	(None, 224, 224, 3)	0	-
Conv1 (Conv2D)	(None, 112, 112, 32)	864	input_layer_2[0]-
bn_Conv1 (BatchNormalization)	(None, 112, 112, 32)	128	Conv1[0][0]
Conv1_relu (ReLU)	(None, 112, 112, 32)	0	bn_Conv1[0][0]
expanded_conv_dept_ (DepthwiseConv2D)	(None, 112, 112, 32)	288	Conv1_relu[0][0]
expanded_conv_dept_ (BatchNormalization)	(None, 112, 112, 32)	128	expanded_conv_dept_
expanded_conv_dept_ (ReLU)	(None, 112, 112, 32)	0	expanded_conv_dept_
expanded_conv_proj_ (Conv2D)	(None, 112, 112, 16)	512	expanded_conv_dept_
expanded_conv_proj_ (BatchNormalization)	(None, 112, 112, 16)	64	expanded_conv_dept_
block_1_expand (Conv2D)	(None, 112, 112, 96)	1,536	expanded_conv_dept_
block_1_expand_BN (BatchNormalization)	(None, 112, 112, 96)	384	block_1_expand[0]
block_1_expand_relu (ReLU)	(None, 112, 112, 96)	0	block_1_expand_BN
block_1_pad (ZeroPadding2D)	(None, 113, 113, 96)	0	block_1_expand_relu
block_1_depthwise (DepthwiseConv2D)	(None, 56, 56, 96)	864	block_1_pad[0][0]
block_1_depthwise_ (BatchNormalization)	(None, 56, 56, 96)	384	block_1_depthwise

block_9_expand_BN (BatchNormalization)	(None, 14, 14, 384)	1,536	block_9_expand[0]
block_9_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_9_expand_BN
block_9_depthwise (DepthwiseConv2D)	(None, 14, 14, 384)	3,456	block_9_expand_relu
block_9_depthwise_ (BatchNormalization)	(None, 14, 14, 384)	1,536	block_9_depthwise
block_9_depthwise_ (ReLU)	(None, 14, 14, 384)	0	block_9_depthwise_
block_9_project (Conv2D)	(None, 14, 14, 64)	24,576	block_9_depthwise_
block_9_project_BN (BatchNormalization)	(None, 14, 14, 64)	256	block_9_project
block_9_add (Add)	(None, 14, 14, 64)	0	block_8_add[0][0], block_9_project_
block_10_expand (Conv2D)	(None, 14, 14, 384)	24,576	block_9_add[0][0]
block_10_expand_BN (BatchNormalization)	(None, 14, 14, 384)	1,536	block_10_expand
block_10_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_10_expand_BN
block_10_depthwise (DepthwiseConv2D)	(None, 14, 14, 384)	3,456	block_10_expand_relu
block_10_depthwise_ (BatchNormalization)	(None, 14, 14, 384)	1,536	block_10_depthwise
block_10_depthwise_ (ReLU)	(None, 14, 14, 384)	0	block_10_depthwise_
block_10_project (Conv2D)	(None, 14, 14, 96)	36,864	block_10_depthwise_
block_10_project_BN (BatchNormalization)	(None, 14, 14, 96)	384	block_10_project
block_11_expand	(None, 14, 14, 55,296)		block_10_project_

Продолжение приложения Д

block_1_depthwise_ (ReLU)	(None, 56, 56, 96)	0	block_1_depthwise_ (ReLU)	(None, 56, 56, 96)	0	block_10_project_ (BatchNormalizatio...	(None, 14, 14, 96)	384	block_10_project_ (BatchNormalizatio...
block_1_project_ (Conv2D)	(None, 56, 56, 24)	2,304	block_1_depthwise_ (ReLU)	(None, 56, 56, 24)	0	block_11_expand_ (BatchNormalizatio...	(None, 14, 14, 576)	2,304	block_11_expand_ (BatchNormalizatio...
block_1_project_BN (BatchNormalizatio...	(None, 56, 56, 24)	96	block_1_project_ (Conv2D)	(None, 56, 56, 144)	3,456	block_11_expand_re_ (ReLU)	(None, 14, 14, 576)	0	block_11_expand_ (BatchNormalizatio...
block_2_expand_ (Conv2D)	(None, 56, 56, 144)	3,456	block_1_project_ (Conv2D)	(None, 56, 56, 144)	576	block_11_depthwise_ (DepthwiseConv2D)	(None, 14, 14, 576)	5,184	block_11_expand_ (BatchNormalizatio...
block_2_expand_BN (BatchNormalizatio...	(None, 56, 56, 144)	576	block_2_expand[0] (ReLU)	(None, 56, 56, 144)	0	block_11_depthwise_ (BatchNormalizatio...	(None, 14, 14, 576)	2,304	block_11_depthwi...
block_2_expand_relu (ReLU)	(None, 56, 56, 144)	0	block_2_expand_B_ (ReLU)	(None, 56, 56, 144)	1,296	block_11_depthwise_ (ReLU)	(None, 14, 14, 576)	0	block_11_depthwi...
block_2_depthwise_ (DepthwiseConv2D)	(None, 56, 56, 144)	1,296	block_2_expand_r_ (ReLU)	(None, 56, 56, 144)	576	block_11_project_ (Conv2D)	(None, 14, 14, 96)	55,296	block_11_depthwi...
block_2_depthwise_ (BatchNormalizatio...	(None, 56, 56, 144)	576	block_2_depthwise_ (ReLU)	(None, 56, 56, 144)	0	block_11_project_BN (BatchNormalizatio...	(None, 14, 14, 96)	384	block_11_project_ (BatchNormalizatio...
block_2_depthwise_ (ReLU)	(None, 56, 56, 144)	0	block_2_project_ (Conv2D)	(None, 56, 56, 24)	3,456	block_11_add (Add)	(None, 14, 14, 96)	0	block_10_project_ block_11_project_...
block_2_project_ (Conv2D)	(None, 56, 56, 24)	3,456	block_2_project_BN (BatchNormalizatio...	(None, 56, 56, 24)	96	block_12_expand_ (Conv2D)	(None, 14, 14, 576)	55,296	block_11_add[0][...
block_2_add (Add)	(None, 56, 56, 24)	0	block_2_project[0] (ReLU)	(None, 56, 56, 24)	0	block_12_expand_BN (BatchNormalizatio...	(None, 14, 14, 576)	2,304	block_12_expand[...
block_3_expand_ (Conv2D)	(None, 56, 56, 144)	3,456	block_2_add [0][0] (ReLU)	(None, 56, 56, 144)	0	block_12_expand_re_ (ReLU)	(None, 14, 14, 576)	0	block_12_expand_ (BatchNormalizatio...
block_3_expand_BN (BatchNormalizatio...	(None, 56, 56, 144)	576	block_3_expand[0] (ReLU)	(None, 56, 56, 144)	0	block_12_depthwise_ (DepthwiseConv2D)	(None, 14, 14, 576)	5,184	block_12_expand_ (BatchNormalizatio...
block_3_expand_relu (ReLU)	(None, 56, 56, 144)	0	block_3_expand_r_ (ReLU)	(None, 56, 56, 144)	0	block_12_depthwise_ (BatchNormalizatio...	(None, 14, 14, 576)	2,304	block_12_depthwi...
block_3_pad (ZeroPadding2D)	(None, 57, 57, 144)	0	block_3_project_ (Conv2D)	(None, 28, 28, 144)	1,296	block_12_depthwise_ (ReLU)	(None, 14, 14, 576)	0	block_12_depthwi...
block_3_depthwise_ (DepthwiseConv2D)	(None, 28, 28, 144)	1,296	block_3_project_BN (BatchNormalizatio...	(None, 28, 28, 32)	4,608	block_12_project_ (Conv2D)	(None, 14, 14, 96)	55,296	block_12_depthwi...
			block_3_project[0] (ReLU)	(None, 28, 28, 32)	128	block_12_project_BN (BatchNormalizatio...	(None, 14, 14, 96)	384	block_12_project_...
			block_4_expand_ (Conv2D)	(None, 28, 28, 192)	6,144	block_13_expand_ (Conv2D)	(None, 14, 14, 576)	55,296	block_12_add[0][...
			block_4_expand_BN (BatchNormalizatio...	(None, 28, 28, 192)	768	block_13_expand_BN (BatchNormalizatio...	(None, 14, 14, 576)	2,304	block_13_expand[...
			block_4_expand_relu (ReLU)	(None, 28, 28, 192)	0	block_13_expand_re_ (ReLU)	(None, 14, 14, 576)	0	block_13_expand_ (BatchNormalizatio...
			block_4_depthwise_ (DepthwiseConv2D)	(None, 28, 28, 192)	1,728	block_13_pad (ZeroPadding2D)	(None, 15, 15, 576)	0	block_13_expand_ (BatchNormalizatio...
			block_4_depthwise_ (BatchNormalizatio...	(None, 28, 28, 192)	768	block_13_depthwise_ (DepthwiseConv2D)	(None, 7, 7, 576)	5,184	block_13_pad[0][...
			block_4_depthwise_ (ReLU)	(None, 28, 28, 192)	0	block_13_depthwise_ (BatchNormalizatio...	(None, 7, 7, 576)	2,304	block_13_depthwi...
			block_4_project_ (Conv2D)	(None, 28, 28, 32)	6,144	block_13_depthwise_ (ReLU)	(None, 7, 7, 576)	0	block_13_depthwi...
			block_4_project_BN (BatchNormalizatio...	(None, 28, 28, 32)	128	block_13_project_ (Conv2D)	(None, 7, 7, 160)	92,160	block_13_depthwi...
			block_4_add (Add)	(None, 28, 28, 32)	0	block_13_project_BN (BatchNormalizatio...	(None, 7, 7, 160)	640	block_13_project_...
			block_5_expand_ (Conv2D)	(None, 28, 28, 192)	6,144	block_14_expand_ (Conv2D)	(None, 7, 7, 960)	153,600	block_13_project_...
			block_5_expand_BN (BatchNormalizatio...	(None, 28, 28, 192)	768	block_14_expand_BN (BatchNormalizatio...	(None, 7, 7, 960)	3,840	block_14_expand[...
			block_5_expand_relu (ReLU)	(None, 28, 28, 192)	0	block_14_expand_re_ (ReLU)	(None, 7, 7, 960)	0	block_14_expand_ (BatchNormalizatio...
			block_5_depthwise_ (DepthwiseConv2D)	(None, 28, 28, 192)	1,728	block_14_depthwise_ (DepthwiseConv2D)	(None, 7, 7, 960)	8,640	block_14_expand_ (BatchNormalizatio...
						block_14_depthwise_ (BatchNormalizatio...	(None, 7, 7, 960)	3,840	block_14_depthwi...
						block_14_depthwise_ (ReLU)	(None, 7, 7, 960)	0	block_14_depthwi...
						block_14_project_ (Conv2D)	(None, 7, 7, 160)	153,600	block_14_depthwi...

Продолжение приложения Д

block_5_depthwise_ (DepthwiseConv2D)	(None, 48, 48, 192)	4,748	block_5_expand_r_			
block_5_depthwise_ (BatchNormalizatio_)	(None, 28, 28, 192)	768	block_5_depthwis_			
block_5_depthwise_ (ReLU)	(None, 28, 28, 192)	0	block_5_depthwis_			
block_5_project (Conv2D)	(None, 28, 28, 32)	6,144	block_5_depthwis_			
block_5_project_BN (BatchNormalizatio_)	(None, 28, 28, 32)	128	block_5_project[-			
block_5_add (Add)	(None, 28, 28, 32)	0	block_4_add[0][0- block_5_project_-			
block_5_expand (Conv2D)	(None, 28, 28, 192)	6,144	block_5_add[0][0]			
block_5_expand_BN (BatchNormalizatio_)	(None, 28, 28, 192)	768	block_5_expand[0-			
block_5_expand_relu (ReLU)	(None, 28, 28, 192)	0	block_5_expand_B-			
block_5_pad (ZeroPadding2D)	(None, 29, 29, 192)	0	block_5_expand_r_			
block_5_depthwise_ (DepthwiseConv2D)	(None, 14, 14, 192)	1,728	block_5_pad[0][0]			
block_5_depthwise_ (BatchNormalizatio_)	(None, 14, 14, 192)	768	block_5_depthwis_			
block_5_depthwise_ (ReLU)	(None, 14, 14, 192)	0	block_5_depthwis_			
block_5_project (Conv2D)	(None, 14, 14, 64)	12,288	block_5_depthwis_			
block_5_project_BN (BatchNormalizatio_)	(None, 14, 14, 64)	256	block_5_project[-			
block_7_expand (Conv2D)	(None, 14, 14, 384)	24,576	block_5_project_-			
block_7_expand_BN (BatchNormalizatio_)	(None, 14, 14, 384)	1,536	block_7_expand[0-			
block_7_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_7_expand_B-			
block_7_depthwise_ (DepthwiseConv2D)	(None, 14, 14, 384)	3,456	block_7_expand_r_			
block_7_depthwise_ (BatchNormalizatio_)	(None, 14, 14, 384)	1,536	block_7_depthwis_			
block_7_depthwise_ (ReLU)	(None, 14, 14, 384)	0	block_7_depthwis_			
block_7_project (Conv2D)	(None, 14, 14, 64)	24,576	block_7_depthwis_			
block_7_project_BN (BatchNormalizatio_)	(None, 14, 14, 64)	256	block_7_project[-			
block_7_add (Add)	(None, 14, 14, 64)	0	block_6_project_- block_7_project_-			
block_8_expand (Conv2D)	(None, 14, 14, 384)	24,576	block_7_add[0][0]			
block_8_expand_BN (BatchNormalizatio_)	(None, 14, 14, 384)	1,536	block_8_expand[0-			
block_8_expand_relu (ReLU)	(None, 14, 14, 384)	0	block_8_expand_B-			
block_8_depthwise_ (DepthwiseConv2D)	(None, 14, 14, 384)	3,456	block_8_expand_r_			
block_8_depthwise_ (BatchNormalizatio_)	(None, 14, 14, 384)	1,536	block_8_depthwis_			
block_8_depthwise_ (ReLU)	(None, 14, 14, 384)	0	block_8_depthwis_			
block_8_project (Conv2D)	(None, 14, 14, 64)	24,576	block_8_depthwis_			
block_8_project_BN (BatchNormalizatio_)	(None, 14, 14, 64)	256	block_8_project[-			
block_8_add (Add)	(None, 14, 14, 64)	0	block_7_add[0][0- block_8_project_-			
block_9_expand (Conv2D)	(None, 14, 14, 384)	24,576	block_8_add[0][0]			
block_14_project_BN (BatchNormalizatio_)	(None, 7, 7, 160)	640	block_14_project_			
block_14_add (Add)	(None, 7, 7, 160)	0	block_13_project_			
block_15_expand (Conv2D)	(None, 7, 7, 960)	153,600	block_14_add[0][			
block_15_expand_BN (BatchNormalizatio_)	(None, 7, 7, 960)	3,840	block_15_expand[-			
block_15_expand_re_ (ReLU)	(None, 7, 7, 960)	0	block_15_expand_-			
block_15_depthwise_ (DepthwiseConv2D)	(None, 7, 7, 960)	8,640	block_15_expand_-			
block_15_depthwise_ (BatchNormalizatio_)	(None, 7, 7, 960)	3,840	block_15_depthwl_			
block_15_depthwise_ (ReLU)	(None, 7, 7, 960)	0	block_15_depthwl_			
block_15_project (Conv2D)	(None, 7, 7, 160)	153,600	block_15_depthwl_			
block_15_project_BN (BatchNormalizatio_)	(None, 7, 7, 160)	640	block_15_project_			
block_15_add (Add)	(None, 7, 7, 160)	0	block_14_add[0][			
block_16_expand (Conv2D)	(None, 7, 7, 960)	153,600	block_15_add[0][			
block_16_expand_BN (BatchNormalizatio_)	(None, 7, 7, 960)	3,840	block_16_expand[-			
block_16_expand_re_ (ReLU)	(None, 7, 7, 960)	0	block_16_expand_-			
block_16_depthwise_ (DepthwiseConv2D)	(None, 7, 7, 960)	8,640	block_16_expand_-			
block_16_depthwise_ (BatchNormalizatio_)	(None, 7, 7, 960)	3,840	block_16_depthwl_			
block_16_depthwise_ (ReLU)	(None, 7, 7, 960)	0	block_16_depthwl_			
block_15_depthwise_ (BatchNormalizatio_)	(None, 7, 7, 960)	3,840	block_15_depthwl_			
block_15_depthwise_ (ReLU)	(None, 7, 7, 960)	0	block_15_depthwl_			
block_15_project (Conv2D)	(None, 7, 7, 160)	153,600	block_15_depthwl_			
block_15_project_BN (BatchNormalizatio_)	(None, 7, 7, 160)	640	block_15_project_			
block_15_add (Add)	(None, 7, 7, 160)	0	block_14_add[0][			
block_16_expand (Conv2D)	(None, 7, 7, 960)	153,600	block_15_add[0][			
block_16_expand_BN (BatchNormalizatio_)	(None, 7, 7, 960)	3,840	block_16_expand[-			
block_16_expand_re_ (ReLU)	(None, 7, 7, 960)	0	block_16_expand_-			
block_16_depthwise_ (DepthwiseConv2D)	(None, 7, 7, 960)	8,640	block_16_expand_-			
block_16_depthwise_ (BatchNormalizatio_)	(None, 7, 7, 960)	3,840	block_16_depthwl_			
block_16_depthwise_ (ReLU)	(None, 7, 7, 960)	0	block_16_depthwl_			
block_16_project (Conv2D)	(None, 7, 7, 320)	307,200	block_16_depthwl_			
block_16_project_BN (BatchNormalizatio_)	(None, 7, 7, 320)	1,280	block_16_project_			
Conv_1 (Conv2D)	(None, 7, 7, 1280)	409,600	block_16_project_			
Conv_1_bn (BatchNormalizatio_)	(None, 7, 7, 1280)	5,120	Conv_1[0][0]			
out_relu (ReLU)	(None, 7, 7, 1280)	0	Conv_1_bn[0][0]			

Продолжение приложения Д

```
img_width, img_height = 224, 224 # Размер изображений
batch_size = 32 # Размер батча
train_dir = '/content/train_rgb_images' # Путь к тренировочным данным
val_dir = '/content/test_rgb_images' # Путь к валидационным данным
#Создайте экземпляры двух классов генераторов изображений:
train_datagen = ImageDataGenerator(
    rescale=1./255,
    data_format='channels_last',
    rotation_range=30, # случайные повороты на +-30 градусов
    horizontal_flip=True, # случайное отражение по горизонтали
    fill_mode='reflect', # заполнение пустых мест отражением краёв
    validation_split=0.2
)

valid_datagen = ImageDataGenerator(
    rescale=1./255,
    data_format='channels_last',
    validation_split=0.2
)
```

```
[ ] # Размер батча:
batch_size=32

train_generator = train_datagen.flow_from_directory(
    directory=train_dir,
    target_size=(224, 224),
    classes=['nodefects', 'attrition', 'pores', 'scratch', 'label'],
    class_mode='categorical',
    batch_size=batch_size,
    shuffle=True,
    seed=42,
)
```

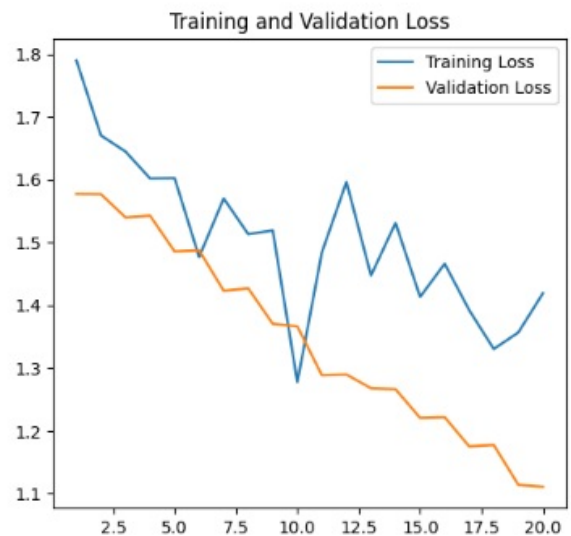
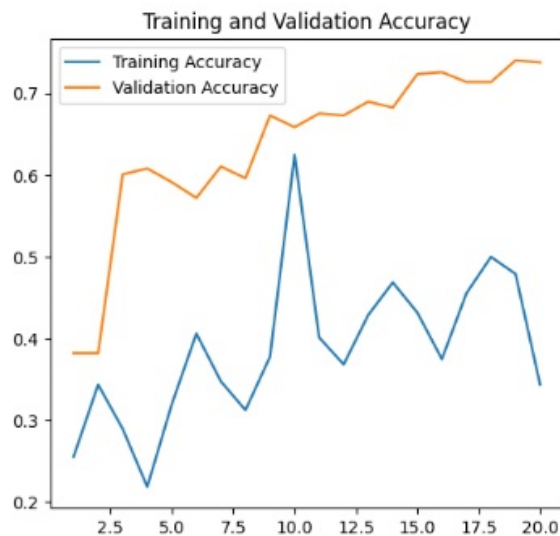
```
Классы: {'nodefects': 0, 'attrition': 1, 'pores': 2, 'scratch': 3, 'label': 4}
Размер пакета изображений (data_batch.shape): (32, 224, 224, 3)
Размер пакета меток (labels_batch.shape): (32, 5)
тип данных изображений (data_batch.dtype): float32
тип данных меток (labels_batch.dtype): float32
Метки в пакете (labels_batch): [[0. 0. 0. 0. 1.]
[0. 0. 1. 0. 0.]
[0. 0. 0. 0. 1.]
[0. 1. 0. 0. 0.]
[0. 1. 0. 0. 0.]
[0. 0. 1. 0. 0.]
[0. 1. 0. 0. 0.]
[0. 0. 0. 0. 1.]
[0. 0. 0. 0. 1.]
[0. 0. 0. 1. 0.]
[0. 0. 1. 0. 0.]
[0. 1. 0. 0. 0.]
[0. 0. 0. 0. 1.]
[0. 0. 0. 1. 0.]
[0. 1. 0. 0. 0.]
[0. 1. 0. 0. 0.]
[0. 0. 0. 1. 0.]
[0. 1. 0. 0. 0.]
[0. 0. 0. 0. 1.]
[0. 0. 1. 0. 0.]
[0. 0. 0. 0. 1.]
[0. 0. 0. 1. 0.]
[0. 0. 1. 0. 0.]
[0. 0. 0. 0. 1.]
[0. 0. 0. 1. 0.]
[0. 0. 0. 0. 1.]
[0. 1. 0. 0. 0.]
[0. 0. 1. 0. 0.]
[0. 0. 0. 0. 1.]]
```


Продолжение приложения Д

```

4070 /usr/local/lib/python3.10/dist-packages/tensorflow/trainers/data_adapter.py:DatasetIterator(11): warning: Your PyDataset class should call 'super().__init__(**kwargs)' in its constructor. **kwargs can include 'worker', 'use_multi_processing', 'max_queue_size'. Do not pass those arguments to 'list()', as they will be ignored.
4071 Epoch 1/200
4072 37s 79ms/step - accuracy: 0.2338 - loss: 1.7608 - val_accuracy: 0.3822 - val_loss: 1.5777
4073
4074 /usr/local/lib/python3.10/dist-packages/tensorflow/trainers/epoch_iterator.py:187: warning: Your input ran out of data; interrupting training. Make sure that your dataset or generator can generate at least 'steps_per_epoch * epochs' batches. You may want to use 'tf.data.Dataset.from_generator()' with 'tf.data.experimental.CACHE' or 'tf.data.experimental.PREFETCH' options.
4075 Epoch 2/200
4076 41 71ms/step - accuracy: 0.3408 - loss: 1.6788 - val_accuracy: 0.4822 - val_loss: 1.5775
4077
4078 Epoch 3/200
4079 41 1s/step - accuracy: 0.2706 - loss: 1.6558 - val_accuracy: 0.4818 - val_loss: 1.5482
4080
4081 Epoch 4/200
4082 55 57ms/step - accuracy: 0.2338 - loss: 1.6824 - val_accuracy: 0.4882 - val_loss: 1.5439
4083
4084 Epoch 5/200
4085 56 57ms/step - accuracy: 0.2525 - loss: 1.6201 - val_accuracy: 0.5551 - val_loss: 1.4881
4086
4087 Epoch 6/200
4088 3s 18ms/step - accuracy: 0.4882 - loss: 1.4772 - val_accuracy: 0.5721 - val_loss: 1.4878
4089
4090 Epoch 7/200
4091 37s 57ms/step - accuracy: 0.3478 - loss: 1.5711 - val_accuracy: 0.6308 - val_loss: 1.4234
4092
4093 Epoch 8/200
4094 55 54ms/step - accuracy: 0.3125 - loss: 1.5136 - val_accuracy: 0.5962 - val_loss: 1.4273
4095
4096 Epoch 9/200
4097 56 57ms/step - accuracy: 0.3858 - loss: 1.5125 - val_accuracy: 0.6731 - val_loss: 1.3787
4098
4099 Epoch 10/200
4100 3s 16ms/step - accuracy: 0.6258 - loss: 1.2778 - val_accuracy: 0.6587 - val_loss: 1.3667
4101
4102 Epoch 11/200
4103 56 57ms/step - accuracy: 0.4884 - loss: 1.4885 - val_accuracy: 0.6755 - val_loss: 1.2888
4104
4105 Epoch 12/200
4106 3s 18ms/step - accuracy: 0.5884 - loss: 1.5584 - val_accuracy: 0.6771 - val_loss: 1.2508
4107
4108 Epoch 13/200
4109 41s 834ms/step - accuracy: 0.4255 - loss: 1.4483 - val_accuracy: 0.6895 - val_loss: 1.2888
4110
4111 Epoch 14/200
4112 55 54ms/step - accuracy: 0.4888 - loss: 1.5315 - val_accuracy: 0.6827 - val_loss: 1.2685
4113
4114 Epoch 15/200
4115 37s 57ms/step - accuracy: 0.4183 - loss: 1.4117 - val_accuracy: 0.7236 - val_loss: 1.2288
4116
4117 Epoch 16/200
4118 3s 18ms/step - accuracy: 0.3754 - loss: 1.4485 - val_accuracy: 0.7268 - val_loss: 1.2228
4119
4120 Epoch 17/200
4121 56 57ms/step - accuracy: 0.4585 - loss: 1.3894 - val_accuracy: 0.7139 - val_loss: 1.1704
4122
4123 Epoch 18/200
4124 55 54ms/step - accuracy: 0.5888 - loss: 1.3386 - val_accuracy: 0.7139 - val_loss: 1.1777
4125
4126 Epoch 19/200
4127 41s 780ms/step - accuracy: 0.4788 - loss: 1.3512 - val_accuracy: 0.7486 - val_loss: 1.1542
4128
4129 Epoch 20/200
4130 3s 18ms/step - accuracy: 0.3408 - loss: 1.4333 - val_accuracy: 0.7388 - val_loss: 1.1111
4131
4132 Accuracy: 0.7099/0.5876/0.6220

```



```
print(f'F1 Score: {f1}')
print('Confusion Matrix:')
print(conf_matrix)
print('Classification Report:')
print(class_report)
```

```

14/14 4s 268ms/step
Accuracy: 0.7399527186761229
Precision: [0. 0.65702479 0. 0.8630137 0.84259259]
Recall: [0. 1. 0. 0.69230769 0.95789474]
F1 Score: [0. 0.79301746 0. 0.76829268 0.89655172]
Confusion Matrix:
[[ 0  3  0  2  1]
 [ 0 159  0  0  0]
 [ 0  61  0  5  6]
 [ 0  18  0 63 10]
 [ 0  1  0  3 91]]
Classification Report:

```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	6
1	0.66	1.00	0.79	159
2	0.00	0.00	0.00	72
3	0.86	0.69	0.77	91
4	0.84	0.96	0.90	95
accuracy			0.74	423
macro avg	0.47	0.53	0.49	423
weighted avg	0.62	0.74	0.66	423

Продолжение приложения Д

```

Found 1683 images belonging to 5 classes.
Found 1683 images belonging to 5 classes.
Классы: {'attrition': 0, 'label': 1, 'nodefects': 2, 'pores': 3, 'scratch': 4}
Размер пакета изображений (data_batch.shape): (16, 400, 400, 1)
Размер пакета меток (labels_batch.shape): (16, 5)
Тип данных изображений (data_batch.dtype): float32
Тип данных меток (labels_batch.dtype): float32
Метки в пакете (labels_batch): [[1. 0. 0. 0. 0.]
 [0. 0. 0. 1. 0.]
 [0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0.]
 [0. 0. 0. 1. 0.]
 [0. 1. 0. 0. 0.]
 [0. 0. 0. 1. 0.]
 [0. 1. 0. 0. 0.]
 [1. 0. 0. 0. 0.]
 [1. 0. 0. 0. 0.]
 [0. 1. 0. 0. 0.]
 [0. 0. 0. 0. 1.]
 [0. 0. 0. 0. 1.]
 [1. 0. 0. 0. 0.]
 [0. 1. 0. 0. 0.]
 [0. 1. 0. 0. 0.]]

```

Model: "functional_25"

Layer (type)	Output shape	Param #
input_layer_6 (InputLayer)	(None, 400, 400, 1)	0
conv2d (Conv2D)	(None, 400, 400, 32)	320
max_pooling2d (MaxPooling2D)	(None, 200, 200, 32)	0
conv2d_1 (Conv2D)	(None, 200, 200, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 100, 100, 64)	0
flatten_2 (Flatten)	(None, 640000)	0
dense_4 (Dense)	(None, 64)	40,960,064
dense_5 (Dense)	(None, 5)	325

Total params: 40,979,205 (156.32 MB)

Trainable params: 40,979,205 (156.32 MB)

```
Non-trainable params: 0 (0.00 B)
```

[illegible]

Продолжение приложения Д

```

➡ Precision per class: [1.          0.94791667 1.          0.984375  0.86868687]
Recall per class: [1.          0.95789474 0.83333333 0.875      0.94505495]
F1-score per class: [1.          0.95287958 0.90909091 0.92647059 0.90526316]
Confusion Matrix:
[[159  0  0  0  0]
 [  0 91  0  0  4]
 [  0  1  5  0  0]
 [  0  0  0 63  9]
 [  0  4  0  1 86]]
Classification Report:

```

	precision	recall	f1-score	support
0	1.00	1.00	1.00	159
1	0.95	0.96	0.95	95
2	1.00	0.83	0.91	6
3	0.98	0.88	0.93	72
4	0.87	0.95	0.91	91
accuracy			0.96	423
macro avg	0.96	0.92	0.94	423
weighted avg	0.96	0.96	0.96	423

```

[ ] # Вариант 2: Использование 'micro' усреднения (подходит, когда важен общий результат)
precision_micro = precision_score(true_classes, predicted_classes, average='micro')
recall_micro = recall_score(true_classes, predicted_classes, average='micro')
f1_micro = f1_score(true_classes, predicted_classes, average='micro')

print("Micro-averaged precision:", precision_micro)

```

```

🔍 print(f'Accuracy: {accuracy}')
print(f'Precision: {precision}')
print(f'Recall: {recall}')
print(f'F1 Score: {f1}')
print('Confusion Matrix:')
print(conf_matrix)
print('Classification Report:')
print(class_report)

```

```

➡ Accuracy: 0.9550827423167849
Precision: [1.          0.94791667 1.          0.984375  0.86868687]
Recall: [1.          0.95789474 0.83333333 0.875      0.94505495]
F1 Score: [1.          0.95287958 0.90909091 0.92647059 0.90526316]
Confusion Matrix:
[[159  0  0  0  0]
 [  0 91  0  0  4]
 [  0  1  5  0  0]
 [  0  0  0 63  9]
 [  0  4  0  1 86]]
Classification Report:

```

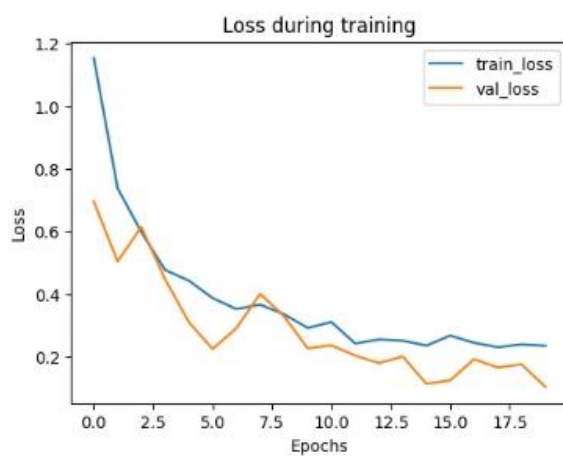
	precision	recall	f1-score	support
0	1.00	1.00	1.00	159
1	0.95	0.96	0.95	95
2	1.00	0.83	0.91	6
3	0.98	0.88	0.93	72
4	0.87	0.95	0.91	91
accuracy			0.96	423
macro avg	0.96	0.92	0.94	423
weighted avg	0.96	0.96	0.96	423

Продолжение приложения Д

```
plt.legend()

# График точности (accuracy)
plt.subplot(1, 2, 2)
plt.plot(history.history['accuracy'], label='train_accuracy')
plt.plot(history.history['val_accuracy'], label='val_accuracy')
plt.title('Accuracy during training')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()

plt.show()
```



Спецификация

91