

УДК 004.93'12

АВТОМАТИЗАЦИЯ ФУНКЦИОНАЛЬНОГО КОНТРОЛЯ МОДУЛЕЙ ИНДИКАЦИИ ДАТЧИКОВ ДАВЛЕНИЯ С ПРИМЕНЕНИЕМ СИСТЕМЫ МАШИННОГО ЗРЕНИЯ

И.В. Днепровский

Национальный исследовательский ядерный университет «МИФИ», Москва, 115409, Россия
e-mail: tomnylow2@gmail.com

Поступила в редакцию: 15.07.2024

После доработки: 15.07.2024

Принята к публикации: 24.09. 2024

В статье рассматривается применение различных алгоритмов машинного зрения для автоматизации функционального контроля ЖК-дисплеев датчиков давления. Анализируются нейронные сети и классические алгоритмы, а также разработанный автором алгоритм. Цель исследования – найти алгоритм, который может быть использован при разработке системы автоматического контроля дисплеев. Алгоритм должен обладать высокой точностью и минимизировать количество ложноотрицательных результатов, чтобы избежать пропуск бракованных изделий. Также он должен быть способен выявлять дефекты подсветки, отсутствие свечения сегментов, неожиданное свечение сегментов, пятна и механические повреждения на дисплее. В статье описаны этапы разработки алгоритма, включая предобработку изображений, бинаризацию, анализ и поиск дефектов. Представленные результаты тестирования алгоритма на тестовой выборке подтверждают его высокую точность и полноту. В ходе работы установлено, что ни один стандартный алгоритм не подходит для функционального контроля модулей индикации, в то время как разработанный автором алгоритм полностью удовлетворяет всем требованиям. Разработанный алгоритм использован для создания системы автоматического контроля ЖК-дисплеев, в настоящий момент применяемой в производстве датчиков давления.

Ключевые слова: машинное зрение, контроль качества, распознавание образов, жидкокристаллический дисплей.

DOI: 10.26583/vestnik.2024.5.8

EDN VPCWIG

ВВЕДЕНИЕ

Первые программные системы обработки изображений начали появляться в 1960-е гг. Тогда же начались прикладные исследования в области распознавания печатных символов [1]. Однако промышленное внедрение комплексов машинного зрения стало возможным лишь в 1980-е гг. благодаря расширению технических возможностей вычислительных систем. В качестве примера можно привести аппарат технического зрения Autovision II, представленный в 1983 г. [2], в котором использовался алгоритм поиска блобов – сгущений одноцветных пикселей для диагностики выпускаемой продукции.

С середины 1990-х гг. подход к обработке изображений основывается на модульной парадигме – обработка изображений должна происходить на различных последовательных уровнях информации: от представления объектов в виде растрового изображения к их символиче-

скому представлению. Эти принципы заложены в современные алгоритмы машинного зрения [3].

Современные технологии машинного зрения включают в себя нейронные сети, методы распознавания образов и объектов на изображениях, что делает возможным создание автономных систем, способных воспринимать и анализировать информацию в реальном времени [4]. Развитие алгоритмов машинного зрения и эволюция вычислительной техники делают данную технологию широкодоступной [5].

Используемый до настоящего времени метод контроля модулей индикации оператором, предполагающий выполнение по существующим методикам значительного числа ручных операций, не позволяет снизить число ошибок человека, неизбежно возникающих из-за общего монотонного характера многократно повторяемых им типовых операций.

Таким образом, цель настоящей работы – нахождение оптимального алгоритма машинного зрения для автоматического контроля качества ЖКИ (жидкокристаллический индикатор).

Объект исследования – алгоритмы машинного зрения.

Предмет исследования – качество распознавания дефектов ЖКИ данными алгоритмами.

Изучение существующих технологий

Системы машинного зрения, применяемые на промышленном производстве, обычно называют *системами технического зрения*. Как правило, они создаются под нужды конкретного заказчика [6], в связи с чем единый подход к их разработке отсутствует, что затрудняет непосредственное сравнение существующих решений между собой. Вместе с тем все системы технического зрения имеют схожую аппаратную и алгоритмическую структуру.

Примерами успешной реализаций такого рода оборудования является продукция компании Mallenom Systems, которая поставляется заказчику в готовом для использования виде. Несмотря на то, что конкретные аппаратно-технические решения, применяемые данными компаниями, являются коммерческой тайной [7], они, как и большинство коммерческих систем, имеют между собой много общего.

Классическое устройство для создания системы технического зрения включает в себя несколько основных компонентов [7]:

- одна или несколько камер (монохромные или цветные);
- процессор;
- программное обеспечение для компьютерного зрения;
- устройства ввода/вывода или каналы связи для отображения полученных результатов, взаимодействия с автоматическими системами.

В целом преимуществами систем технического зрения являются:

- высокая производительность, обусловленная программной обработкой данных и принятием решений;
- точность и надежность благодаря отсутствию оператора, которому свойственны усталость и ошибки;
- применимость в сложных условиях, где человеческое восприятие ограничено или невозможно.

Среди недостатков систем технического зрения стоит отметить:

- высокие финансовые затраты на их внедрение, особенно на начальном этапе;
- сложность настройки, требующая привлечения (обучения) квалифицированных специалистов для их калибровки;
- ограничения в восприятии информации по сравнению с человеком, и, как следствие этого, возникновение ошибок в работе¹.

Непосредственным аналогом разрабатываемой системы является система роботизированного контроля качества электронных плат от компании Остек. Система представляет собой роботизированный комплекс, способный осуществлять контроль расположения компонентов, качество пайки, нанесение покрытий, наличие механических дефектов на компонентах².

Данная система может быть дообучена распознаванию пользовательских дефектов на печатных платах, т.е. возможно ее применение для функционального контроля модулей ЖКИ. Исходя из этого, можно сделать вывод о том, что система использует предобученные сверточные нейронные сети.

Исходная система функционального контроля

Функциональный контроль модулей ЖКИ представляет собой процесс проверки работоспособности всех символов на дисплее, равномерность его засветки, яркость подсветки, отсутствие механических повреждений, работоспособность кнопок и микропереключателей (рис. 1).

Функциональный контроль заключается в проверке контрастности включенных символов, отсутствии свечения выключенных, равномерности подсветки дисплея³.

Такой подход имеет ряд недостатков:

¹ Как работает машинное зрение: алгоритмы и задачи // Блог компании Siderus: [Электронный ресурс]. URL: <https://siderus.ru/blog/tekhnologii/kak-rabotaet-mashinnoe-zrenie-algoritmy-i-zadachi/> (дата обращения: 04.01.2024).

² Opisanie robotizirovannoj sistemy' opticheskogo kontrolya FLEX: Robotoinspekcija [Description of the robotic optical inspection system FLEX: Robotic inspection] // Ostec company blog: [website]. Available at: <https://ostec-flex.ru/solutions/robotizirovannaya-sistema> (access date: 04.01.2024).

³ Инструкция по функциональному контролю ТЖИУ.687281.273И14 / М.: ФГУП «ВНИИА им. Н.Л. Духова», 2010. 28 с.

- низкая производительность – для проверки одного ЖКИ нужно более 30 с;
- человеческий фактор – оператор может не заметить неравномерность подсветки или дефекты контрастности сегментов.

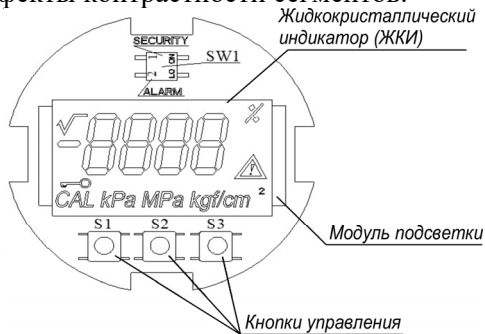


Рис. 1. Общий вид модуля ЖКИ

Проверка ЖКИ оператором занимает от 30 до 35 секунд (от 1.5 до 1.8 секунды на один символ), при должной квалификации оператора. Время проверки может увеличиваться ввиду усталости, также из-за этого оператор может пропускать некоторые шаги проверки.

Таким образом, имеющаяся система является морально устаревшей и не позволяет обеспечивать релевантный уровень производительности современного производства. Требуется ее замена системой, минимизирующей влияние человеческого фактора, для чего требуется найти алгоритм машинного зрения, способный к обнаружению всех вышеуказанных возможных дефектов.

1. АНАЛИЗ СУЩЕСТВУЮЩИХ АЛГОРИТМОВ МАШИННОГО ЗРЕНИЯ

1.1. Общий подход к анализу изображений

Подготовка изображений для применения в обучении любого алгоритма представляет собой этап, который напрямую влияет на эффективность и точность обучения модели [8]. Для достижения оптимального качества обучающего набора данных применяются различные техники.

Сжатие изображений представляет собой процесс уменьшения их размеров путем объединения соседних пикселей в один. Это позволяет уменьшить объем данных, необходимых для обучения, и сократить вычислительные затраты.

Бинаризация изображений включает преобразование многоканального формата изображения (например, RGB) в одноканальный бинарный формат, где каждый пиксель представлен

Примеры некачественно проверенных модулей приведены на рис. 2.



Рис. 2. Модули, прошедшие некачественный контроль

значением 0 или 1 в зависимости от установленного порогового значения. Пороговое значение – величина, определяющая границу яркости пикселя, выше которой он будет переведен в 1 – белый пиксель, ниже – в 0, т.е. черный пиксель. Пороговое значение может быть единым для всего изображения, определяемым заранее, или иметь различную величину в разных частях изображения, адаптируясь так, чтобы выделять контрастные детали.

В рамках решения задачи по поиску дефектов на дисплеях имеет значение обрезка изображений вдоль контура дисплея. Для достижения данной цели требуется применение алгоритмов поиска границ.

Алгоритмы поиска границ используются для выявления контуров объектов на изображении, что позволяет определить область интереса. Морфологические замыкания позволяют устранить небольшие пробелы или дефекты в контуре, обеспечивая более точное выделение границы. В работе используется алгоритм Кенни, поскольку он является наиболее распространенным и универсальным [8].

Аугментация изображений включает в себя создание дополнительных вариаций обучающих изображений путем применения различных преобразований: поворот, сдвиг, изменения масштаба и добавление шума. Этот процесс увеличивает разнообразие обучающих данных, что, в свою очередь, помогает повысить обобщающую способность модели и уменьшить риск переобучения [9]. Аугментация является неотъемлемой частью процесса обучения сверточных нейронных сетей, поскольку для их обучения требуется большой объем обучающих данных, который часто недоступен в реальных условиях.

Важно отметить, что предобработка изображений на этапе обучения должна полностью совпадать с предобработкой, применяемой к уже обученной нейросети в реальных условиях эксплуатации. Это гарантирует правильное функционирование и обеспечивает согласованность между предсказаниями модели, полученными в процессе обучения, и ее реальным поведением на практике.

1.2. Метрики качества

Для оценки эффективности предсказаний алгоритма используется тестовый набор данных, который представляет собой подмножество обучающего набора, которое не было использовано в процессе обучения. Этот набор позволяет оценить обобщающую способность на данных, которые не были известны во время обучения [9].

Matrix of confusion, также известная как матрица ошибок, является инструментом для визуализации производительности бинарной классификационной модели – модели, имеющей два ответа – 0 и 1, которые могут быть интерпретированы как принадлежность или непринадлежность к классу. Обычно считается, что класс 1 представлен меньшим числом экземпляров. Существует аналог матрицы ошибок для задачи классификации с большим числом классов.

Матрица состоит из четырех основных компонентов: true positives (TP), false positives (FP), true negatives (TN) и false negatives (FN).

Выбор между такими различными показателями оценки, как точность, полнота и прецизионность, может зависеть от конкретной задачи [8]. Точность (accuracy) измеряет общую способность модели делать правильные прогнозы для всех классов, однако неинформативна при существенно различном числе элементов в классах. Точность – отношение числа правильных прогнозов ко всем прогнозам. Если ложноположительные результаты (false positives) являются более критичными, чем ложноотрицательные (false negatives), то прецизионность (precision) можно считать более важным показателем. Если важно минимизировать количество ложно-отрицательных результатов, то полнота (recall) становится более значимой метрикой. F-мера объединяет в себе полноту и прецизионность и является универсальной, однако может не подходить для специфических задач (рис. 3).

Confusion Matrix			
<i>Actual \ Predict</i>	0	1	
0	TN	FN	$Precision = \frac{TP}{TP + FP}$
1	FP	TP	$Recall = \frac{TP}{TP + FN}$
			$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall}$
			$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$

Рис. 3. Матрица ошибок и метрики качества

При поиске дефектов ЖКИ критична минимизация числа ложноотрицательных результатов, т.е. пропуска брака в дальнейшее производство. Исходя из этого, ключевая характеристика – полнота.

При этом алгоритм не должен допускать значительного количества ложноположительных срабатываний, поскольку в таком случае потребуется частое вмешательство оператора в процесс контроля. При объеме проверки около 300 дисплеев в день и дополнительных затратах времени на повторный контроль человеком в 20 с, допустимый уровень прецизионности должен быть не ниже 0.95. Это дает дополнительные 5 мин работы оператора в день.

1.3. Тестирование качества распознавания дефектов алгоритмами

1.3.1. Сверточная нейросеть

Для создания сверточных нейронных сетей использовалась библиотека Tensorflow для языка программирования Python.

Так как при создании любой нейросети заранее неизвестно, какая структура покажет лучшие результаты [8], было проведено исследование качества распознавания фото неисправных и работоспособных дисплеев.

В ходе экспериментов использовались различные структуры нейронных сетей: AlexNet, MobileNet, сеть собственной структуры. AlexNet – сверточная нейронная сеть, которая стала прорывом в области компьютерного зрения и глубокого обучения, когда была представлена в 2012 г. [8]. Структура сети приведена на рис. 4.

MobileNet – легковесная нейронная сеть, созданная Google, специально оптимизированная для использования на мобильных устройствах и встроенных системах с ограниченными ресурсами [9]. Использование легковесной сети было предпочтительно ввиду ограниченных вычис-

лительных возможностей технологических ПК производства. Структура сети приведена на рис. 5.

Для сравнения с готовыми нейросетями была написана простая сеть, состоящая из трех свер-

точных слоев, трех слоев пуллинга, слоя преобразования в одномерный массив и двух слоев полносвязной сети (рис. 6).

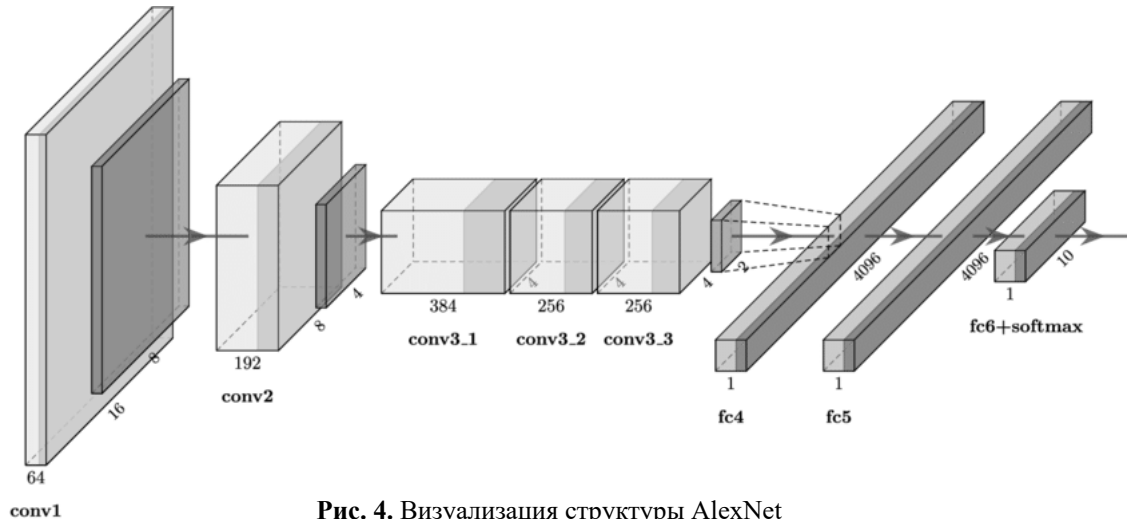


Рис. 4. Визуализация структуры AlexNet

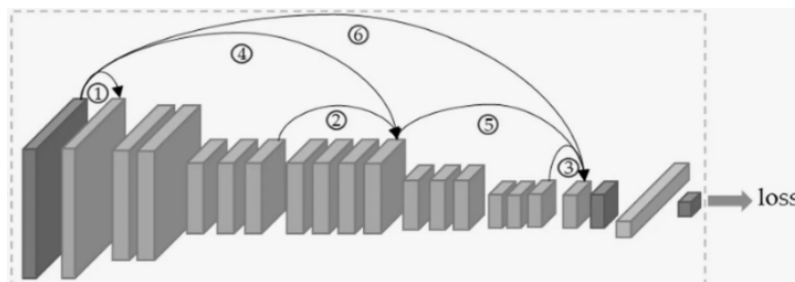


Рис. 5. Визуализация MobileNet

```
model = Sequential([
    # Сверточный слой с 32 фильтрами размером 3x3, активация ReLU, входной размер
    # изображения 200x300, 1 канал цвета
    Conv2D(32, (3, 3), activation='relu', input_shape=(200, 300, 1)),
    # Слой пуллинга с окном размером 2x2 для уменьшения размерности
    MaxPooling2D((2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    Conv2D(128, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    # Выравнивающий слой для подготовки данных к подаче на полносвязный слой
    Flatten(),
    # Полносвязный слой с 128 нейронами и активацией ReLU
    Dense(128, activation='relu'),
    # Выходной слой с 20 нейронами для классификации на 20 классов и активацией
    softmax
    Dense(20, activation='softmax')
])
```

Рис. 6. Структура разработанной сети на языке Python

Также в процессе разработки было неясно, стоит ли выносить каждый символ в отдельный класс и решать задачу мультиклассовой классификации или объединить изображения рабо-

тоспособных символов в один класс, а всех не-исправных – в другой, и решать более простую задачу бинарной классификации.

Была подготовлена обучающая выборка фотографий дисплея, с помощью алгоритма аугментации каждой исходной фотографии было поставлено в соответствие 10 фото с добавлением шума, размытия и поворота. Таким образом, обучающая выборка составила около 130 тысяч изображений.

Бинарная классификация. При разделении выборки на два класса – исправные и неисправные, наблюдается дисбаланс количества изображений. Это вызвано изначально небольшим количеством фото неисправных дисплеев ввиду числа доступных для имитации неисправностей. Всего в выборке было представлено около 30 видов неисправностей, что после аугментации составляет около 40 тысяч, т.е. около трети всей выборки. Чтобы избежать проблем, связанных с дисбалансом, количество изображений исправных дисплеев было уменьшено примерно до 40 тысяч. Вся выборка делилась на обучающую и тестовую с использованием инструментов Tensorflow, что позволяет сохранить исходное соотношение элементов в классах. Объем тестовой выборки – 20 %, обучающей – 80 %.

На рис. 7–9 представлены матрицы ошибок нейросетей, в табл. 1 приведены метрики качества.

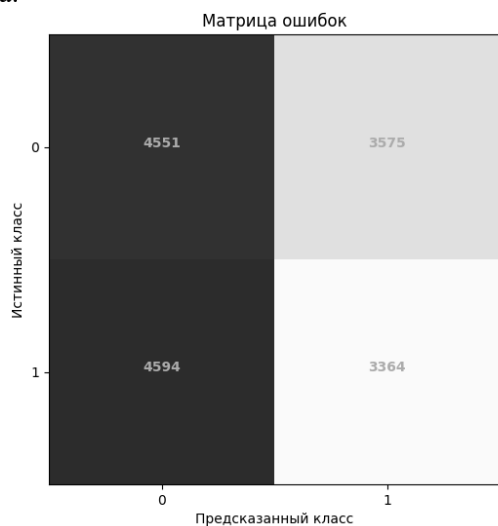


Рис. 7. Матрица ошибок AlexNet

Таблица 1. Метрики точности нейросетей при бинарной классификации

Показатель качества, ед.	AlexNet	MobileNet	Разработанная сеть
Точность	0.49	0.57	0.57
Прецизионность	0.48	0.58	0.57
Полнота	0.42	0.51	0.56

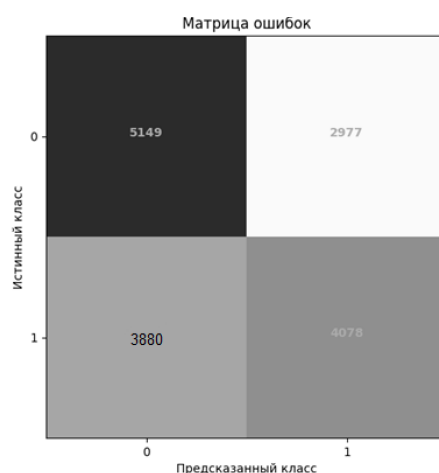


Рис. 8. Матрица ошибок MobileNet

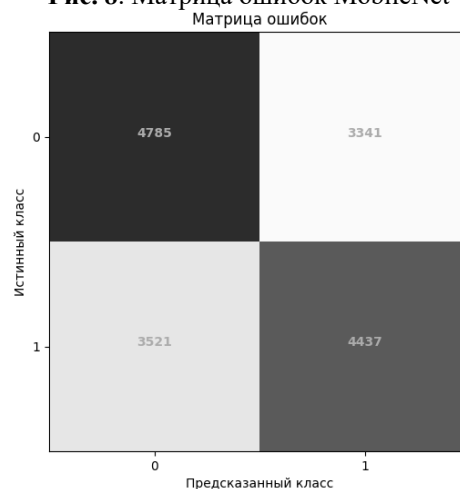


Рис. 9. Матрица ошибок разработанной сети

Из матриц точности и таблицы видно, что AlexNet дает результат, сопоставимый с простым угадыванием. Возможно, это связано со слишком сложной структурой сети, которая не смогла корректно обучиться. MobileNet также показывает неудовлетворительные результаты, однако лучшие чем AlexNet. Результаты разработанной сети схожи с MobileNet и также немногим отличаются от угадывания.

Использование бинарной классификации показало результаты на уровне случайного угадывания, нейросеть при различной внутренней структуре не могла найти общие признаки у различных рабочих символов и отличить их от нерабочих.

Многоклассовая классификация. Так как использование бинарной классификации показало неудовлетворительный результат, было решено выделить каждый исправный символ в свой класс, а все неисправные объединить в один. Таким образом, было получено 19 классов

исправных символов, т.е. примерно по 7 тысяч изображений на класс. Результаты работы сетей приведены в табл. 2, матрицы ошибок для большого числа классов теряют наглядность и поэтому не приведены. Используется макро-усреднение, так как оно лучше отображает качество распознавания классов, в которых содержится значительно меньше элементов, чем в других.

Таблица 2. Метрики точности нейросетей при многоклассовой классификации

Показатель качества, ед.	Alex-Net	Mobile-Net	Разработанная сеть
Точность	0.961	0.985	0.972
Прецизионность (макроусреднение)	0.935	0.945	0.939
Полнота (макроусреднение)	0.923	0.931	0.912

Выделение каждого символа в свой класс позволило обучить нейросети различать рабочие символы между собой с точностью более 99.5 %, но при этом ни одна из них не могла с вероятностью выше 80 % отличить сломанный символ от такого же рабочего, чем и обусловлено снижение полноты и прецизионности.

Выяснить причину, по которой ни одна из сетей не смогла уверенно распознавать неисправные символы, не удалось. Возможно, это вызвано тем, что все нерабочие сегменты в символе всегда выглядят по-разному, но при этом лишь немного отличны от рабочих. Исправный и неисправный символы приведены на рис. 10.

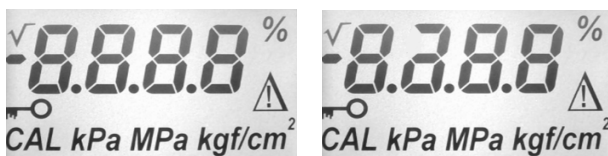


Рис. 10. Пример нерабочего сегмента в символе: слева – включены все символы, дисплей исправен; справа – включены все символы, один сегмент отсутствует

Таким образом, от использования нейросети пришлось отказаться, поскольку не удалось добиться удовлетворительного качества распознавания.

2.3.2. Классические классификаторы

При использовании классических классификаторов входное изображение преобразуется в

одномерный вектор длиной, равной произведению ширины на высоту изображения. Данный метод очень прост, но обладает существенным недостатком – теряется информация о взаимном расположении пикселей в соседних строках и столбцах. Для тестирования качества распознавания дефектов использовались алгоритмы kNN, SVM, LOF. Все указанные алгоритмы реализованы в библиотеке SciKit для Python. Тестирование качества распознавания проводилось аналогично задаче бинарной классификации для нейросетей.

Поскольку данные классификаторы требуют намного меньшего количества исходных данных для обучения, размер обучающей выборки составил 800 фотографий исправных и неисправных дисплеев, тестовой – 200. Результаты приведены на рис. 11–13 и в табл. 3.

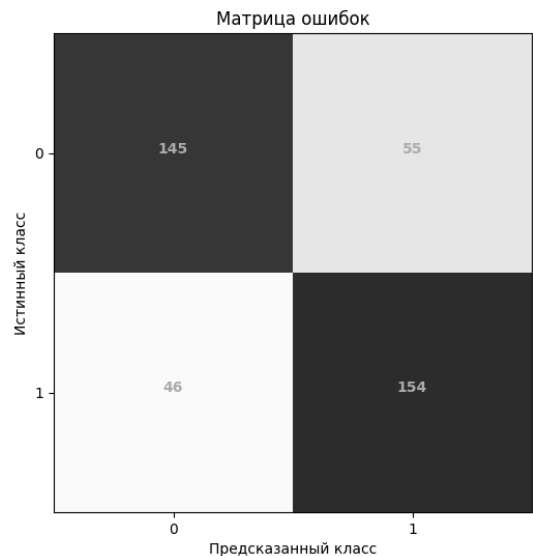


Рис. 11. Матрица ошибок для SVM

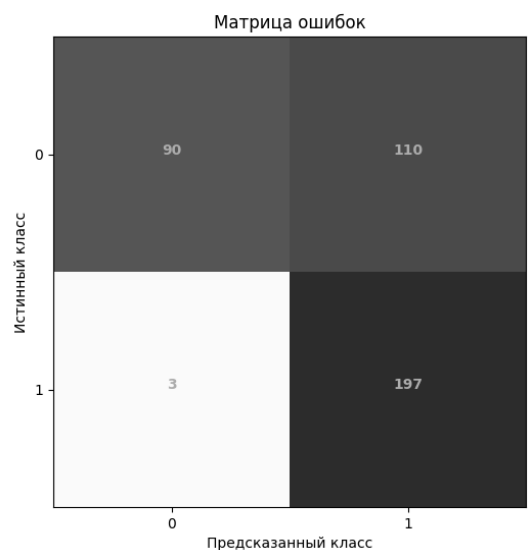


Рис. 12. Матрица ошибок для LOF

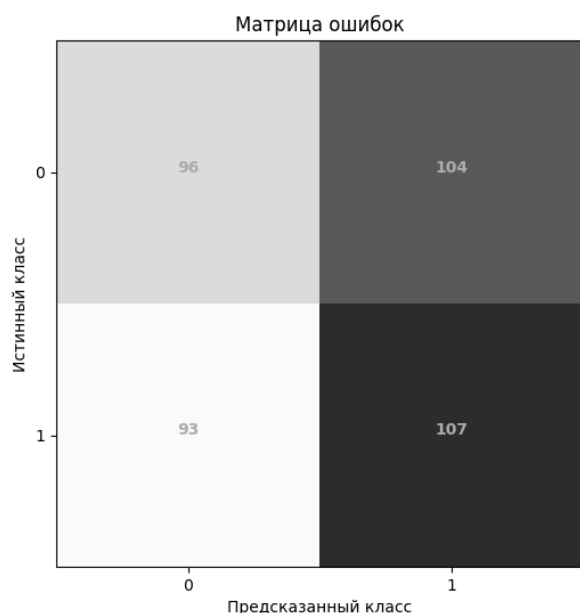


Рис. 13. Матрица ошибок для kNN

Таблица 3. Метрики качества классических классификаторов

Метрики качества, ед.	SVM	LOF	kNN
Точность	0.75	0.72	0.51
Прецизионность	0.74	0.64	0.51
Полнота	0.77	0.98	0.54

Из матриц ошибок и таблицы видно, что SVM показывает результаты лучшие, чем любая нейросеть. LOF имеет наибольшую полноту – количество ложноотрицательных результатов минимально, однако невысокая прецизионность говорит о большом количестве ложноположительных результатов, число которых выходит за рамки допустимого. kNN имеет метрики качества на уровне случайного угадывания. Также в процессе тестирования была выявлена сильная зависимость результатов от изначального качества фото – незначительные изменения угла наклона камеры приводят к сильно отличному результату работы алгоритма. Таким образом, ни один классификатор не может быть использован в разрабатываемой системе.

2.3.3. Разработанный алгоритм

Так как использование других алгоритмов показало неудовлетворительные результаты, было решено создать более простой алгоритм, не имеющий возможности интерпретации дефектов в широком диапазоне, но при этом не требующий обучения. Идея алгоритма вдохнов-

лена одной из первых систем машинного зрения – Autovision II [2].

– Работа алгоритма состоит из следующих шагов:

- фотографирование дисплея;
- обрезка фотографии по контуру дисплея и коррекция искажений перспективы;
- сжатие изображения для упрощения анализа;
- бинаризация изображения по заранее установленному порогу;
- поиск черных пикселей внутри заранее размеченных областей;
- анализ заполнения области черными пикселями;
- вынесение вердикта на основании числа черных пикселей в областях.

Обрезка изображения. Для коррекции искажений перспективы и поворота при различной установке камеры относительно проверяемого дисплея применяется программная обрезка изображения по контуру дисплея. Обрезка позволяет минимизировать влияние искажений перспективы и наклона камеры.

Процесс начинается с преобразования изначального изображения в бинарный формат, затем применяется операция морфологического замыкания контуров для устранения небольших пробелов. Далее применяется алгоритм Кенни для обнаружения и аппроксимации контуров с использованием многоугольников. Для поиска контура дисплея используется аппроксимация многоугольника четырехугольником.

Алгоритм аппроксимации стремится соединить наиболее удаленные точки таким образом, чтобы ни одна из промежуточных точек не выходила за пределы «коридора» (рис. 14)¹.

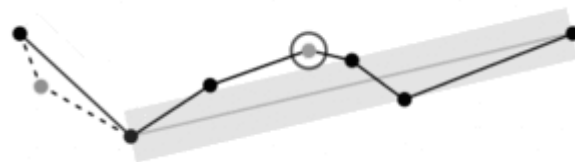


Рис. 14. Визуализация аппроксимации

При аппроксимации четырехугольником настраивается генерация четырех точек с определенным порядком – начиная с правого верх-

¹ Canny Edge Detection [Электронный ресурс]. URL: https://docs.opencv.org/4.x/da/d22/tutorial_py_canny.html (дата обращения: 04.01.2024).

него угла против часовой стрелки. Это обеспечивает одинаковую ориентацию изображений на каждой итерации.

В случае неудачной попытки аппроксимации программа сообщает об ошибке пользователю. Ошибка чаще всего вызвана неправильной установкой камеры на плату или проблемами с подсветкой. После успешной аппроксимации программа находит контур с наибольшей площадью, предполагая, что этот контур является краем дисплея. Затем происходит обрезка изображения по найденному контуру с возможностью ручной настройки отступов от края. Это сделано для повышения универсальности алгоритма при использовании различных камер. Далее на рис. 15–18 представлены основные шаги обрезки изображения.

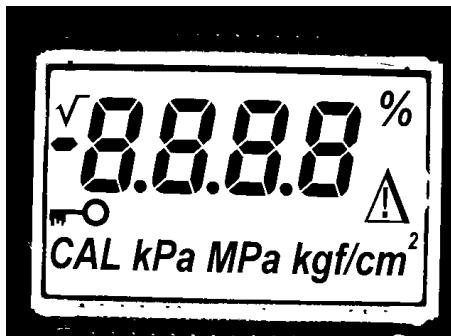


Рис. 15. Бинаризованное изображение до морфологического замыкания

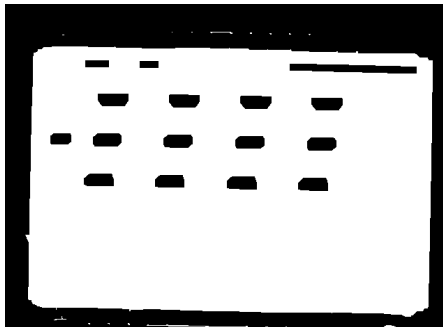


Рис. 16. Бинаризованное изображение после морфологического замыкания



Рис. 17. Фото с найденным контуром, показанным жирным прямоугольником



Рис. 18. Фото после обрезки с учетом пользовательских отступов

Можно видеть, что обрезка позволила исправить искажение поворота и выделить для дальнейшего исследования лишь значимую часть дисплея.

Предобработка изображения. После обрезки изображение представляет собой цветную фотографию дисплея. Для улучшения процесса анализа изображения применяются методы сжатия и бинаризации.

Имеется опасность потери деталей изображения при слишком сильном сжатии, например тонких линий в некоторых сегментах. С другой стороны, недостаточное сжатие приведет к замедлению процесса анализа изображения. Экспериментально было установлено, что оптимальным является размер 300 на 200 пикселей для сжатого изображения. После этапа сжатия происходит бинаризация изображения. Применение бинаризации после сжатия, а не до него, позволяет получить более четкие границы сегментов.

Бинаризация основана на использовании фиксированного порога, который был определен экспериментально и составил 130 относительных единиц из 255. Установленное значение порога обеспечивает надежное преобразование фона в белые пиксели, в случае если подсветка функционирует корректно; в противном случае часть с нерабочей подсветкой будет преобразована в черные пиксели (рис. 19).

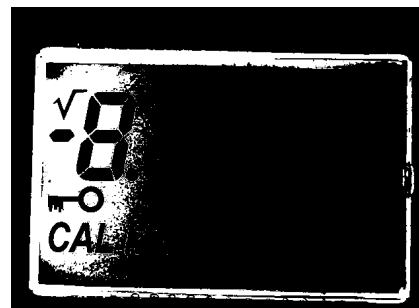


Рис. 19. Пример фото после бинаризации при нерабочем правом светодиоде подсветки

Работоспособные сегменты, обладающие необходимой контрастностью, преобразуются в черные пиксели. Если сегмент имеет низкую

контрастность, то он будет преобразован в белые пиксели в соответствии с этим порогом. Все это позволяет находить дефекты на этапе анализа.

Анализ. Для анализа применяются заранее определенные области, представленные в виде прямоугольников. Положение и размеры прямоугольников определяются двумя парами координат: левым верхним и правым нижним углами. Эти координаты сохранены в формате JSON файла в виде словаря, ключами которого являются краткие имена символов, а значениями – кортеж кортежей. Вложенные кортежи хранят информацию о координатах прямоугольников. Дополнительный кортеж требуется для обработки некоторых символов, состоящих из нескольких сегментов. Для проверки таких символов несколько прямоугольников должны проверяться за одну итерацию.

В процессе анализа и во многих других частях программы используется библиотека `numpy`, реализующая широкий спектр математических операций. Сам процесс анализа представляет собой поиск черных пикселей в `numpy` массиве, представляющем сжатое изображение. Использование массива `numpy` и методов поиска ненулевых элементов в нем позволяет увеличить скорость распознавания одного изображения более чем в 4 раза по сравнению с использованием стандартных контейнеров `Python`.

Если процент черных пикселей внутри области меньше, чем пороговое значение, то область считается пустой, в противном случае – окрашенной. Пороговое значение процента черных пикселей подобрано так, чтобы исключить шум, который мог быть ошибочно принят за черные пиксели там, где их на самом деле нет.

Алгоритм анализирует все области и делает положительное заключение, если все области, где изначально не ожидалось наличие черных пикселей на данном этапе, оказываются пустыми, а все области, где ожидалось их наличие, – окрашенными (рис. 20).

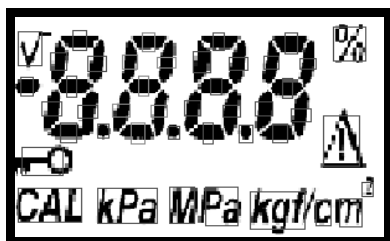


Рис. 20. Визуализация проверяемых областей на сжатой и бинаризованной фотографии

Таким образом, алгоритм позволяет обнаруживать дефекты подсветки, отсутствие свечения сегментов, неожиданное свечение сегментов, пятна и механические повреждения на дисплее (см. рис. 21).

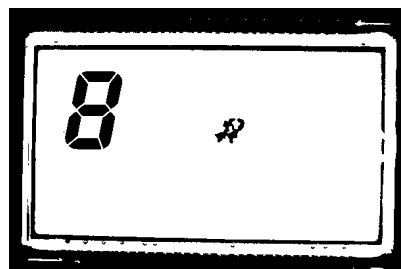


Рис. 21. Пример найденного механического повреждения на этапе проверки отдельных символов

В случае, аналогичном рис. 21, повреждения будут найдены на этапе проверки отдельных символов, если повреждение находится в области одного из символов, либо же на финальной проверке пустого дисплея. Анализ одного изображения занимает менее 0.05 с, однако ввиду инертности ЖКИ для стабилизации работы используется задержка 0.3 с, что дает время полной проверки ЖКИ приблизительно 5.5 с.

Таким образом, использование разработанного алгоритма сокращает время проверки ЖКИ более чем в пять раз.

Качество поиска дефектов. Для определения метрик качества разработанной системы было проведено тестирование 100 фотографий дисплея с различной контрастностью. Приемлемый уровень падения контрастности определяется визуально так, чтобы все сегменты сохраняли читаемость при различных углах обзора и внешнем освещении. На основе субъективного предварительного разделения фотографий на две равные группы: с несоответствиями и без них, было проведено тестирование. На рис. 22 приведен пример бинаризованного изображения и исходной фотографии, с отмеченной неисправной областью.

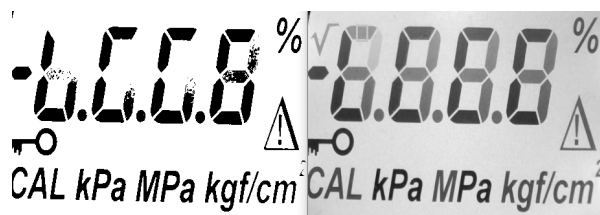


Рис. 22. Пример дисплея с низкой контрастностью символов

При этом можно видеть, что пиксели областей пониженной контрастности, например символы «CAL», «М», «kgf/cm», корректно переводятся в черные, хотя и имеют не максимальную контрастность. На рис. 23 приведена матрица ошибок для разработанной системы, а в табл. 4 – ее метрики качества.

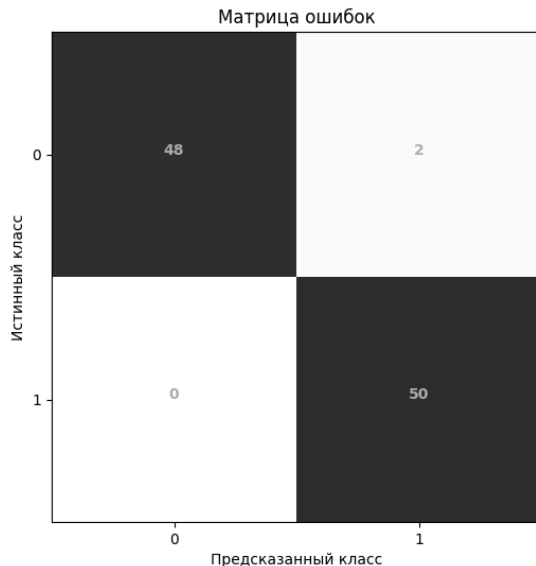


Рис. 23. Матрица ошибок для разработанной системы

Таблица 4. Метрики качества разработанной системы

Точность, ед.	0.98
Прецизионность, ед.	0.96
Полнота, ед.	1.00

Можно видеть, что полнота равна единице, т.е. система для тестовой выборки всегда обнаруживает платы с дефектами при правильной предварительной настройке, следовательно не имеет ошибки второго рода. Но при этом возможны ошибки первого рода, т.е. признание исправной платы браком. С целью недопущения признания исправного ЖКИ браком в программу, разработанную на основе данного алгоритма должно быть добавлено окно подтверждения, позволяющее оператору убедиться, что непрошедшая автоматический контроль плата действительно имеет несоответствия. Достижение точности распознавания дефектов в 100 % возможно благодаря предварительной настройке рабочего места, в ходе которой настраивается порог бинаризации. Это позволяет в дальнейшем копировать уровень восприятия контрастности человеком. Однако из-за естествен-

ного дрейфа характеристик оборудования, входящего в систему, данный порог может смещаться, что в дальнейшем может приводить к неверной бинаризации. Для того чтобы избежать этого, система должна проходить периодический контроль и настройку.

РЕЗУЛЬТАТЫ И ИХ ЗНАЧЕНИЕ

В ходе исследования было определено, что ни одна из использовавшихся сверточных нейронных сетей, а также ни один классический классификатор не имели требуемого уровня полноты или прецизионности (см. табл. 1–3).

Разработанный алгоритм напротив, полностью удовлетворяет требованиям (см. табл. 4). Исходя из этого, данный алгоритм был положен в основу создания системы автоматизированного функционального контроля модулей индикации. Разработанное на его основе приложение позволяет управлять устройством проверки и камерой, что, в свою очередь, увеличивает скорость проверки одного дисплея более чем в пять раз по сравнению с проверкой человеком.

Прототип системы проверки, разработанной на основе данного алгоритма, был передан на участок функционального контроля для проверки тестовой партии плат модулей индикации. По результатам тестирования алгоритм показал полную пригодность для решения данной задачи.

По итогам лабораторных испытаний, а также по результатам тестов в производственном отделе планируется внедрение данной системы на участок функционального контроля.

Полученные результаты могут быть не вполне ожидаемыми, так как в последние годы нейросети получили широкое распространение. Однако, как говорится в [10], они не являются оптимальным решением для всех случаев.

Полученные результаты подтверждают то, что классические алгоритмы машинного зрения все еще могут находить применение в решении реальных задач.

При необходимости разработанный алгоритм допускает изменение для контроля других ЖКИ, что может быть использовано для проверки модулей индикации последующих поколений датчиков давления.

СПИСОК ЛИТЕРАТУРЫ

1. David E.E., Selfridge O.G. 1962. Eyes and Ears for Computers // Proceedings of the IRE, 1962, V. 50, Iss. 5. P. 1093–1101. <https://doi.org/10.1109/JRPROC.1962.288011>.
2. Theodore L. Warren, Kenneth R. Whelan, Arnold G. Reinhold. Vision system (Patent No. US4577344A). United States, 1983. Assignee: Acuity Imaging.
3. Емельянов С.Г. Методы и средства обработки изображений // Приборостроение. 2009. Т. 52. № 2. С. 11–12.
4. Vinokurov I.V. Using a convolutional neural network to recognize text elements in poor quality scanned images // Program systems: theory and applications, 2022. V. 13. № 3 (54). P. 45–59.
5. Rokunuzzaman, Jayasuriya, H.P. Development of a low cost machine vision system for sorting of tomatoes // Agricultural Engineering International: The CIGR Journal, 2013. V. 15. P. 173–180.
6. Царева Е. Машинное зрение для контроля качества упаковки выпускаемой продукции // Тара и упаковка, 2019. № 2. С. 10–12. URL: https://www.mallenom.ru/Docs/Machine_vision_article_may2019.pdf (дата обращения: 19.04.2024).
7. Раков Н.С., Пальмов С.В. Машинное зрение // Форум молодых ученых. 2018. № 4 (20). URL: <https://cyberleninka.ru/article/n/mashinnoe-zrenie> (дата обращения: 19.04.2024).
8. Гудфеллоу Я., Бенджио И. Глубокое обучение. М.: ДМК Пресс, 2018. 652 с.
9. Мюллер А., Гвидо С. Введение в машинное обучение с помощью Python. М.: Гевиста, 2017.
10. Ferguson Max K, Ronay A., Lee Y-T.T, Law K.H. Detection and Segmentation of Manufacturing Defects with Convolutional Neural Networks and Transfer Learning // Smart Sustain. Manuf. Syst., 2018, V. 2(1), P. 137–164. <http://doi.org/10.1520/SSMS20180033>.

Vestnik Natsional'nogo Issledovatel'skogo Yadernogo Universiteta «MIFI», 2024, vol. 13, no. 5, pp. 358–370

**AUTOMATION OF FUNCTIONAL CONTROL
OF PRESSURE SENSOR DISPLAY MODULES USING MACHINE VISION SYSTEMS**

I.V. Dneprovskii

National Research Nuclear University «MEPhI», Moscow, 115409, Russia

e-mail: tomnylow2@gmail.com

Received July 15, 2024; revised July 15, 2024; accepted September 24, 2024

The article considers the application of various machine vision algorithms for automation of functional testing of LCD displays of pressure sensors. Neural networks and classical algorithms, as well as the algorithm developed by the author, are analyzed. The purpose of the study is to find an algorithm that can be used in the development of an automatic display testing system. The algorithm should be highly accurate and minimize the number of false negative results in order to avoid missing defective products. It should also be able to detect backlight defects, lack of segment glow, unexpected segment glow, spots and mechanical damage on the display. The article describes the stages of algorithm development, including image preprocessing, binarization, analysis and defect search. The presented results of testing the algorithm on a test sample confirm its high accuracy and completeness. In the course of the work, it was found that no standard algorithm is suitable for the functional testing of display modules, while the algorithm developed by the author fully meets all the requirements. The developed algorithm is used to create an automatic LCD display testing system, currently used in the production of pressure sensors.

Keywords: machine vision, quality control, pattern recognition, liquid crystal display.

REFERENCES

1. David E.E., Selfridge O.G. 1962. Eyes and Ears for Computers. Proceedings of the IRE, 50, 1093–1101. <https://doi.org/10.1109/JRPROC.1962.288011>.
2. Theodore L. Warren, Kenneth R. Whelan, Arnold G. Reinhold. Vision system (Patent No. US4577344A). United States, 1983. Assignee: Acuity Imaging.
3. Emelyanov S.G. Metody i sredstva obrabotki izobrazhenij [Methods and means of image processing].

Priborostroenie, 2009. Vol. 52, No. 2. Pp. 11–12 (in Russian).

4. *Vinokurov I.V.* Using a convolutional neural network to recognize text elements in poor quality scanned images. Program systems: theory and applications. 2022. Vol. 13, no. 3 (54). Pp. 45–59.

5. *Rokunuzzaman, Jayasuriya, H.P.* Development of a low cost machine vision system for sorting of tomatoes. Agricultural Engineering International: The CIGR Journal, 2013. Vol. 15. Pp. 173–180.

6. *Tsareva E.* Mashinnoe zrenie dlya kontrolya kachestva upakovki vypuskaemoj produkcii [Machine vision for quality control of product packaging]. Tara i upakovka, 2019. No. 2. Pp. 10–12 (in Russian). Available at: https://www.mallenom.ru/Docs/Machine_vision_article_may2019.pdf (accessed: 19.04.2024).

7. *Rakov N.S., Palmov S.V.* Mashinnoe zrenie [Machine vision] // Forum molody'x ucheny'x [Forum of young scientists]. 2018. No. 4 (20). Available at: <https://cyberleninka.ru/article/n/mashinnoe-zrenie> (accessed: 19.04.2024).

8. *Goodfellow Y., Bengio I.* Glubokoe obuchenie [Deep learning]. Moscow, DMK Press Publ., 2018. 652 p.

9. *Muller A.* Vvedenie v mashinnoe obuchenie s pomoshch'yu Python [Introduction to Machine Learning with Python]. Moscow, Gevissta Publ., 2017.

10. *Ferguson Max K., Ronay A., Lee Y-T.T, Law K.H.* Detection and Segmentation of Manufacturing Defects with Convolutional Neural Networks and Transfer Learning. Smart Sustain. Manuf. Syst., 2018, Vol. 2(1), Pp. 137–164. <http://doi.org/10.1520/SSMS20180033>.