

УДК 004.932

СОЗДАНИЕ АЛГОРИТМА ДЛЯ ОПРЕДЕЛЕНИЯ РАЗМЕРОВ И ПЛОТНОСТИ ДЕФЕКТОВ  
С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ КОМПЬЮТЕРНОГО ЗРЕНИЯ

И. М. ОЛЬХОВСКАЯ

(Представлено: канд. физ.-мат. наук, доц. С. А. ВАБИЩЕВИЧ)

*Представлен алгоритм обработки изображений поверхности полупроводника, содержащей выявленные травлением дефекты, позволяет повысить достоверность и объективность обработки визуальной информации при определении количества и плотности дефектов на образце. Применение настоящего алгоритма может способствовать как получению значимых научных результатов в области материаловедения полупроводников, так и расширению возможностей промышленной дефектоскопии, позволяя отслеживать качество изделий электронной промышленности не только на конечном этапе, но и непосредственно во время производственного процесса.*

**Введение.** Компьютерная обработка и распознавание изображений представляет собой быстро развивающееся направление в компьютерном моделировании. Данный вид обработки может использоваться для получения информации с изображения (например, распознавание номерного знака автомобиля, подсчет числа и типа клеток в поле микроскопа и т.д.). Компьютерное зрение – это одна из самых востребованных областей на данном этапе развития глобальных цифровых компьютерных технологий. Оно требуется на производстве, при управлении роботами, при автоматизации процессов, в медицинских и военных приложениях, при наблюдении со спутников и при работе с персональными компьютерами, в частности поиске цифровых изображений [2]. Машинное обучение (machine learning / ML) в общем и компьютерное зрение (computer vision / CV), в частности, используют глубокие архитектуры, позволяющие детектировать многочисленные объекты разных категорий, предсказывать пространственное расположение объектов друг относительно друга и многое другое. Однако, в большинстве случаев практических применений компьютерного зрения программное обеспечение не носит универсальный характер, поскольку компьютеры предварительно запрограммированы для решения отдельных определенных задач. В данной работе будет рассмотрена возможность его применения в материаловедении полупроводниковых материалов для определения плотности дефектов материала [1].

Целью работы является разработка алгоритма программы, позволяющей определить дефекты упаковки в полупроводниках, выявленных при травлении материала, с помощью компьютерного зрения.

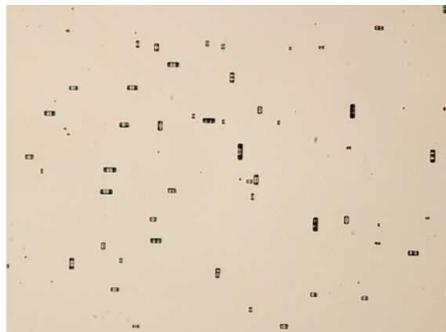
**Создание алгоритма.** Для разработки программного продукта был выбран язык программирования Python, потому что по сравнению с кодом, написанным на таких языках, как C / C ++, Java, код, написанный на Python должен быть 2 ~ 10 раз меньше, то есть время, затраченное на разработку программы будет значительно меньше. Еще одно преимущество – понятность кода, поскольку синтаксическая особенность Python – выделение блоков кода отступами, что значительно упрощает зрительное восприятие программ, написанных на этом языке. Программы, написанные на языке программирования Python, не переводятся в машинный код, а сразу выполняются программой-интерпретатором. Это позволяет запускать код на любой платформе с установленным заранее интерпретатором.

Последовательность обработки изображения программой сводилась к следующим этапам:

- получение изображения;
- предварительная обработка (произведение цветовой коррекции изображения), выделение контуров дефектов;
- детектирование(сегментация) – выбор участков или отдельных точек, которые содержат нужный для обработки объект – дефект;
- подсчет количества дефектов на выбранном участке.

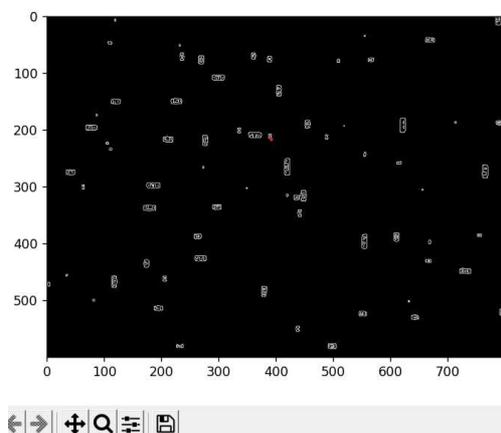
**Входные данные изображения.** Каждое изображение состоит из набора пикселей. Пиксель – это строительный блок изображения. Если представить изображение в виде сетки, то каждый квадрат в сетке содержит один пиксель, где точке с координатой (0, 0) соответствует верхний левый угол изображения, входными данными являются изображения дефектов упаковки, выявленных при травлении полупроводника. Дефект упаковки представляет собой локальную область в кристалле, в которой нарушена правильная последовательность укладки атомных плоскостей [1]. На выбранном изображении (рис.1) дефекты располагаются по всей площади фотографии. Большинство из них представляют собой многоугольники. Размер изображения 800\*600 пикселя. Всё изображение представлено в цветовом пространстве RGB. Каждая из трёх компонент представлена целым числом в диапазоне от 0 до 255 включительно, которое указывает как «много» цвета содержится. Исходя из того, что каждая компонента представлена в диапазоне [0,255], то для того, чтобы представить насыщенность каждого цвета, нам будет достаточно 8-битного целого беззнакового числа.

**Обработка входных данных.** Для обработки изображений была выбрана библиотека OpenCV [2-3]. OpenCV (Open Source Computer Vision Library) – это открытая библиотека для работы с алгоритмами компьютерного зрения, машинным обучением и обработкой изображений. Работает на Windows, Linux, MacOS, Android. Для удобства обработки данных изображение проходит две стадии обработки: предварительная и основная.



**Рисунок 1. – Входное цветное изображение с фигурами травления типа «дефекты упаковки» на поверхности кремния (увеличение x200)**

**Предварительная обработка изображения.** Для удобства нахождения дефектов травления, изображение переводим в другое оттеночное пространство цветов: серое, для этого применяем функцию `cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)`. Первый аргумент, передаваемый функции – это `img`, обработанный аргумент, содержащий путь к изображению, загруженному в программу. Поскольку мы сохранили изображение в той же папке, просто передадим имя файла изображения. Второй аргумент конвертирует изображение в оттенки серого. После обработки изображения «фильтром» серого цвета спектр цветов сузился, что позволило упростить процесс нахождения контуров дефектов, то есть на данном этапе (подсчет количества дефектов) нас не интересует «глубина» дефекта, поэтому внутренняя часть дефекта закрашивается черным.



**Рисунок 2. – Изображение после обработки «фильтром» с выделенными контурами фигур травления**

**Поиск «замкнутых» дефектов.** Дефектами на изображении являются фигуры травления полупроводника. На фотографии большинство имеют замкнутую форму, т.е. представлены в виде треугольников, прямоугольников, кругов или других фигур более сложной геометрической формы. Для нахождения самих контуров применяется функция `cv2.findContours()`. Следует отметить, что параметры, принимаемые функцией `cv2.findContours()`, являются двоичными изображениями, то есть черно-белыми (не в градациях серого), поэтому сначала необходимо прочитать преобразованное изображение в градациях серого. И затем преобразуется в двоичную карту. Для нахождения количества сторон контуров применяется функция `cv2.approxPolyDP()`, где мы указываем минимальное количество сторон, которое может быть у дефекта, тем самым задавая форму минимально возможного N-угольника. После подсчета всех дефектов для нахождения их плотности мы делим общее количество дефектов на количество пикселей изображения, после чего принимая во внимание первоначальные размеры образца переводим плотность из (число/пиксель) в единицы СИ.

**Заключение.** Представленный алгоритм обработки изображений поверхности полупроводника, содержащей выявленные травлением дефекты, позволяет повысить достоверность и объективность обработки визуальной информации при определении количества и плотности дефектов на образце. Это имеет важное значение для материаловедения твердых тел, поскольку современное высокотехнологичное производство повышает требования к контролю качества выпускаемой продукции. Применение настоящего алгоритма позволит получать не только значимые научные результаты, но и расширит возможности промышленной дефектоскопии, позволяя отслеживать качество изделий электронной промышленности не только на конечном этапе, но и непосредственно во время производственного процесса.

#### ЛИТЕРАТУРА

1. Рейви, К. Дефекты и примеси в полупроводниковом кремнии / К. Рейви; пер. с англ. В.В. Высоцкой, П.П.Поздеева, Т.М. Ткачевой, О.П. Федоровой. – М.: Мир, 1984. – 472 с.
2. Alberto Fernández Villán, Mastering OpenCV 4 with Python. A practical guide covering topics from image processing, augmented reality to deep learning with OpenCV 4 and Python 3.7//ISBN-9781789344912, – 2019, – 532 p.
3. Дэвид А. Форсайт, Джин Понс, Компьютерное зрение. Современный подход. 928 стр., с ил.; ISBN 5-8459-0542-7, 0-13-085198-1; формат 70x100/16; твердый переплетофсетная2004, 1 кв.; Вильямс.