

УДК 004.932

ИСПОЛЬЗОВАНИЕ КОМПЬЮТЕРНОГО ЗРЕНИЯ ДЛЯ ПОИСКА КОНТУРОВ ОБЪЕКТОВ

С.И. РОГОВСКИЙ

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

Рассмотрены вопросы использования цифровой обработки изображений поверхности при испытаниях на микротвердость для определения геометрических размеров отпечатков, зоны деформации. Построен алгоритм обработки изображения и реализована программа обработки. Результаты моделирования могут быть использованы при испытаниях полимерных пленок на микротвердость для определения прочностных характеристик: микротвердости, трещиностойкости, удельной энергии отслаивания.

Введение. Информационные технологии широко распространены во всех сферах жизнедеятельности человека. Они дают нам возможность быстро и наиболее точно проводить расчеты, обрабатывать информацию с наибольшей точностью и исключить субъективность в анализе результатов.

Целью настоящей работы была разработка программы, позволяющей получить информацию о прочностных характеристиках материалов на основании анализа отпечатков, образующихся при микроиндентировании образцов.

Физические аспекты проекта. Современный процесс производства полупроводниковых приборов предусматривает ряд технологических обработок, в результате которых, помимо прочих, изменяются механические свойства материалов, что, в свою очередь, может приводить к появлению микротрещин, царапин, сколов и иных дефектов поверхности. Для определения прочностных характеристик материалов используются различные методики, однако наиболее приближенным к реальному технологическому процессу является микроиндентирование, т. к. данный метод позволяет наиболее точно моделировать контактное взаимодействие абразивных частиц с обрабатываемым материалом [1]. Существует необходимость получения программного продукта для распознавания цифровых изображений, позволяющего проводить на основании анализа фотоснимков полный расчет прочностных характеристик материала. В качестве среды разработки программного продукта был выбран язык программирования Python. Схема обработки и анализа данных цифровых изображений сводилась к следующим этапам: анализ входных данных изображения; обработка входных данных; использование медианного фильтра; сегментация изображения; поиск контуров изображения; отображение контуров на изображении; вычисление геометрических параметров объекта фотографирования.

Входные данные изображения. Входными данными являются фотоизображения отпечатков при индентировании полимера. В центре наблюдается отпечаток пирамиды индентора в полимерной пленке, окруженный навалами полимера. Зона навалов представляет сферическую форму. Изображение получено методом дифференциально-интерференционного контраста [2], позволяющего выявить неровности поверхности полимера. Изображение состоит из набора пикселей. Если представить изображение в виде сетки, то каждый квадрат в сетке содержит один пиксель, где квадрат с координатами [0,0] – это левый верхний пиксель. Размер изображения 200x200 пикселей, что в совокупности составляет 40000 пикселей. Все пиксели в изображении представлены в цветовом пространстве RGB (red, green, blue – красный, зелёный, синий), где одно значение для красной компоненты, одно для зелёной и одно для синей. Каждая из трёх компонент выражена целым числом в диапазоне от 0 до 255 включительно, которое указывает как «много» цвета содержится.

Обработка входных данных. Для обработки данных изображений выбрана библиотека OpenCV и язык программирования Python [3]. OpenCV (Open Source Computer Vision Library) – одна из самых популярных библиотек для приложений по компьютерному зрению. OpenCV-Python – это Python-версия интерфейса для OpenCV. Наличие кода на языке C/C++ данной версии библиотеки в бэкенде гарантирует быстроту работы библиотеки, а Python-обертка во фронтенде обеспечивает легкость настройки и развертывания. Благодаря этому OpenCV-Python является отличным решением для высоконагруженных вычислительных программ по компьютерному зрению. Также в библиотеке содержится огромное количество алгоритмов, есть интерфейсы для многих языков программирования – Java, Ruby, Matlab, Lua, C/C++.

Фильтрация исходных данных. Входное изображение содержит шум, который мешает анализу изображения. Существует много различных типов шума, таких как гауссовский шум, шум «соли» и «перца» и т.д. Мы можем удалить этот шум из изображения, применив фильтр, который удаляет его или, по крайней мере, сводит к минимуму его влияние. Существует множество различных фильтров, например усредняющий фильтр (box filter), медианный фильтр (median filter), фильтр типов волн (модовый фильтр, mode filter), фильтр Гаусса (Gaussian filter) и многие другие. Для удаления шума в нашем алгоритме обработки используем медианный фильтр. Медианный фильтр – один из видов цифровых фильтров, широко используемый в цифровой обработке сигналов и изображений для уменьшения уровня шума. Медианное сглаживание широко используется в алгоритмах обнаружения контуров, потому что при определенных условиях

оно сохраняет края, удаляя шум. Для поиска медианы требуется расположить все значения цветов соседних пикселей по возрастанию и взять средний элемент с этого ряда. Если значений в ряде четное количество, то нужно взять среднее значение двух средних элементов. Полученное число и будет являться медианой. Элементы для поиска медианы берутся с соседних пикселей, которые образуют матрицу и центральным элементом является обрабатываемый пиксель.

Сегментация изображения. Фон характеризуется постоянным значением цветов, а для следа характерно резкое изменение значения цвета. Чтобы выделить контур, нужно отделить след (объект) от фона. Для отделения следа от фона используется сегментация. Как обсуждалось выше, изображение состоит из компонентов RGB и каждый компонент имеет значения от 0 до 255. Если установить в программе значение сегментации 140, то все значения компонентов выше 140 примут значения 255, а компоненты со значениями ниже 140 примут значение 0. После сегментации на изображении след имеет более отчетливый контур. Т.е. фон стал черным и принял значения 0, а область внутри следа стала более яркой и приняла соответствующее значение.

Поиск контуров. Контур объекта – это его видимый край, который отделяет след от фона. В действительности большинство методов анализа изображений работают именно с контурами, а не с пикселями как таковыми. Библиотека OpenCV реализует удобные методы для детектирования и манипуляции с контурами изображения. Для поиска контуров используется функция *cv2.FindContours()*. Данная функция принимает три значения: изображение, режим группировки и метод упаковки. Согласно своему формату, данная функция поиска контуров принимает изображение в оттенках серого. Оттенки серого – цветовой режим изображений, которые отображаются в оттенках серого цвета, размещённые в виде таблицы в качестве эталонов яркости белого цвета. Яркость в цветовом пространстве оттенков серого определяется по формуле [3]:

$$Y = 0,2989 \cdot r + 0,5870 \cdot g + 0,1140 \cdot b ,$$

где r , g и b – интенсивность (в диапазоне от 0 до 255) соответственно красной, зелёной и синей составляющих определяемого цвета.

Чтобы перевести изображение в цветовое пространство оттенков серого применяется функция *cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)*. После применения функции поиска контуров изображение готово для обработки. Полученные контуры следа нужно отобразить на исходном изображении при помощи функции *cv2.drawContours()*. Функция принимает изображение, контур, цвет контура (заданный в палитре RGB), толщину контура в пикселях.

На изображении может выделиться несколько маленьких ненужных контуров, которые нужно убрать. Перед отображением контуров на изображении нужно их отсортировать по площади. При помощи функции *cv2.contourArea()* можно получить значение площади выбранного контура. Функция принимает контур и выдает значение площади в пикселях. Если контур будет слишком мал, то он не будет отображаться на изображении и не будет влиять на результат измерений. Функция расчета площади задается выражением: $S = cv2.contourArea(contours[i])$.

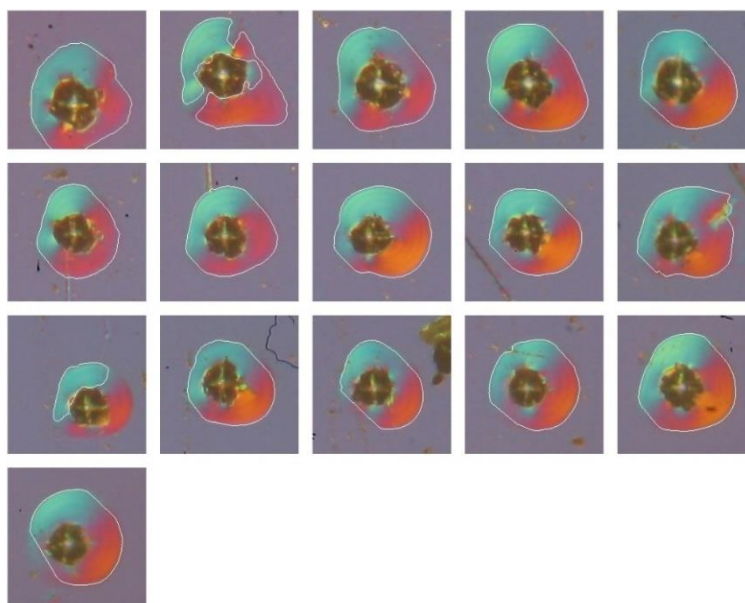


Рисунок. – Результаты отображения контуров

Из-за неравномерного освещения мы получили несколько контуров, которые нужно объединить в один. Контур состоит из точек, и при соединении точек контура получили объект, который совпал со следом. Полученный след совпадает с исходным обрабатываемым изображением, контур к которому является целью обработки. Для получения полного контура остается применить функцию поиска контуров и отобразить полученный контур на исходном изображении. Окончательный результат обработки изображений показан на рисунке.

Реализация данного алгоритма поиска контура может быть использована для определения геометрических параметров отпечатков при индентировании полимеров, что дает возможность рассчитать такие прочностные характеристики как микротвердость, трещиностойкость K_{Ic} (коэффициент интенсивности напряжений) и удельную энергию отслаивания пленки, которая является характеристикой адгезии полимерной пленки к основе [4–6].

Заключение. Таким образом, реализован алгоритм обработки цветных изображений, полученных при фотографировании поверхности полимера после испытаний на микротвердость, имеющий своей целью определение геометрических размеров отпечатков, контуров областей навалов, длин трещин и других объектов. Это имеет существенное значение для объективного определения прочностных характеристик материалов и для автоматизации процесса измерения в материаловедении.

ЛИТЕРАТУРА

1. Литвинов, Ю.М. Методология определения механических свойств полупроводниковых материалов с помощью метода непрерывного вдавливания индентора/ Ю.М. Литвинов, М.Ю.Литвинов// Известия вузов. Материалы электронной техники. – 2004. – № 4. – С.11-16.
2. Анисович, А.Г. Оптические эффекты при микроскопии неметаллических материалов/ А.Г.Анисович // Литье и металлургия. – 2017. – №1. – С.110-114.
3. Bradski, G. Learning OpenCV. Computer vision with the OpenCV library/ G.Bradski, A.Kaehler //O'Reilly Media, Inc., – 2008. – 580 p.
4. Вабищевич, С.А. Физико-механические свойства облученных пленок диазохинон-новолачного фоторезиста на кремнии/ С.А.Вабищевич [и др.]// Вестник Полоцкого государственного университета. Серия С. Фундаментальные науки. Физика – 2020. – № 12. – С.60-64.
5. Прочностные свойства структур фоторезист-кремний, γ -облученных и имплантированных ионами В⁺ и Р⁺ / С.А.Вабищевич [и др.] // Вестник Полоцкого государственного университета. Серия С. Фундаментальные науки. Физика – 2016. – № 12. – С.30-36.
6. Vabishchevich S.A., Brinkevich S.D., Brinkevich D.I., Prosolovich V.S. Adhesion of diazoquinon-novolac photoresist films with implanted boron and phosphorus ions to single-crystal silicon// High energy chemistry. – 2020. – V.54, № 1. – P.46-50.