

РАСПОЗНАВАНИЕ ИНДУСТРИАЛЬНЫХ БАРКОДОВ DATAMATRIX С ПРОИЗВОЛЬНЫМ УГЛОМ НАКЛОНА И ПОВОРОТА КАМЕРЫ

Кручинин А.Ю.

Оренбургский государственный университет,
ООО «ИнтБуСофт»

Аннотация

Представлен алгоритм распознавания двумерных промышленных кодов DataMatrix, позволяющий идентифицировать коды, нанесённые ударно-точечным или капельным методами на поверхности со сложным фоном. Алгоритм может адаптироваться в случаях изменения фона на основе смены режима бинаризации.

Ключевые слова: распознавание баркодов, DataMatrix, бинаризация изображений, управление процессом распознавания.

Введение

Технология распознавания промышленных кодов DataMatrix активно применяется в промышленности при нанесении на поверхности, на которые невозможно нанести/наклеить напечатанный код. Например, на металлические трубы. Подобные коды наносятся или ударно-точечным, или капельным методами. Существующие промышленные решения для распознавания промышленных кодов DataMatrix, в том числе компании SICK AG (Германия), не позволяют распознавать коды в условиях с повышенным шумовым фоном, который возникает при недостаточном отличии точек кода от шума.

Несмотря на то, что существуют алгоритмы распознавания промышленных кодов, например [1, 2], данные алгоритмы обладают рядом недостатков: неустойчивость поведения при произвольном наклоне и повороте камеры, большая чувствительность к помехам. Поэтому был разработан собственный алгоритм распознавания промышленных кодов DataMatrix с использованием результатов, полученных в работе [3]. В качестве требований к алгоритму выступили следующие условия:

- размеры радиуса точек кода меняются в пределах от 3 до 6 пикселей;
- размеры кода – от 10×10 до 32×32 ;
- наклон камеры по отношению к плоскости кода – от 90 до 45 градусов;
- поворот камеры может быть любой (код может быть повернут в плоскости на произвольный угол);
- поверхность, на которую нанесён код, и условия освещения могут меняться.

Требования были сформулированы на основе условий работы промышленных роботов компании Интелмет Технолоджис.

1. Алгоритм распознавания

Как и в большинстве методов распознавания графических образов, первоначально необходимо осуществить детектирование кода. Для этого можно анализировать перепады яркости, использовать морфологический градиент или метод обнаружения бинарных объектов, представленный в работе [4]. В большинстве случаев детектирование применяется для увеличения производительности распознавания при

анализе больших изображений. Поскольку в работе предполагается анализ изображений в формате 640×480 пикселей, поступающих в форме видеопотока с промышленного робота, то этап детектирования будет игнорирован (код занимает много места на изображении, поэтому детектирование лишено смысла). Дальнейшую работу алгоритма можно разбить на несколько этапов, что представлено на рис. 1.

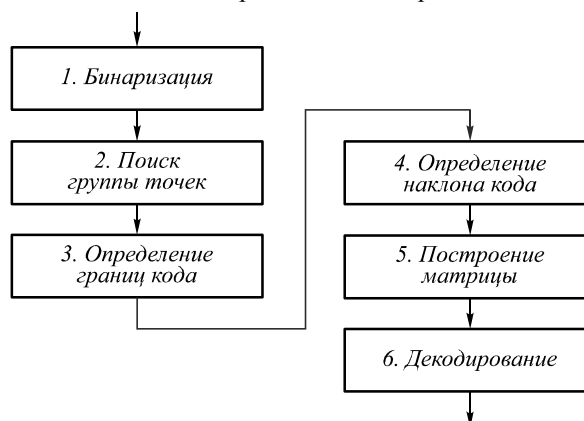


Рис. 1. Этапы работы алгоритма распознавания промышленных баркодов

1. Бинаризация. Этап бинаризации имеет огромное значение по причине того, что здесь происходит выделение необходимой информации. На вход подаётся полноцветное или в градациях серого изображение I , а на выходе – изображение с двумя цветами I_b :

$$I_b(x, y) = f_b(I(x, y)), \quad (1)$$

где f_b – функция бинаризации.

Неправильная бинаризация приведёт или к потере информации, или к недостаточному выделению точечных признаков. Подобрать бинаризацию, оптимальную для любых возможных случаев, очень тяжело. На рис. 2 показан пример промышленного DataMatrix и результатов обработки пороговой бинаризации.

Как видно из рисунка, такой бинаризацией не всегда возможно получение нужных данных. Так, на рис. 2б показано, что выделены не все данные, а на рис. 2в слишком много шумов. На настоящий момент известно достаточно много критериев бинарности (бимодальности), например, критерий Отсу, Бернсена, Эйквеля, Ниблэка и т.п. Эти критерии можно было бы использовать для вычисления оптимального

порога, однако этот подход не годится в случаях с переменной освещённостью в различных участках изображения. В этом случае можно использовать адаптивный метод бинаризации в окрестности каждой точки (подобная процедура есть в библиотеке компьютерного зрения OpenCV). Однако в случае, когда фоновая поверхность не является однотонной, подобный метод будет давать значительное количество шумов и невысокое качество бинаризации в целом. Поэтому использование того же критерия Отсу, но не глобально, а локально даст более качественные результаты. Например, в работе [5] предлагается локальное использование критерия Ниблэка для бинаризации штриховых кодов. Существуют и комбинации локального подхода бинаризации с методами выделения краёв [6], среди которых методы Собела, Кенни, морфологического градиента и т.п.

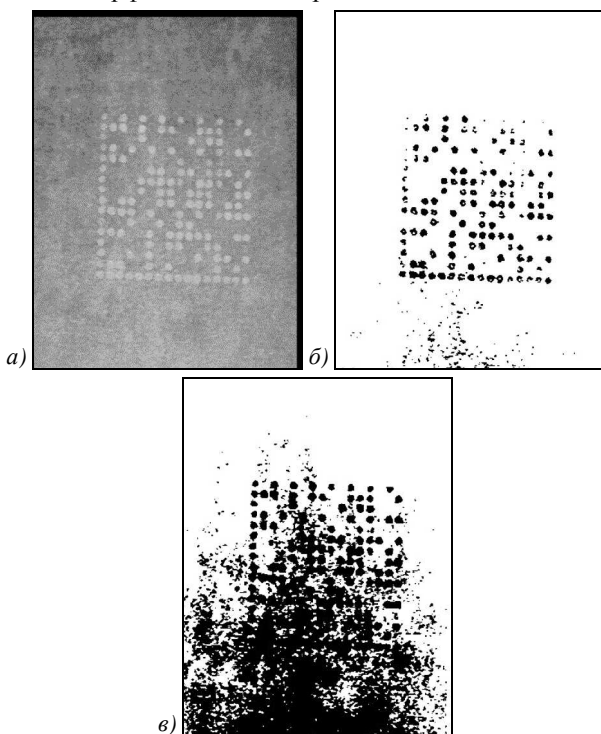


Рис. 2. Промышленный DataMatrix (а), пороговая бинаризация (б), пороговая бинаризация с более низким порогом (в)

Особенностью распознавания промышленных DataMatrix-кодов, нанесённых на поверхность металла, например, труб, является широкий разброс условий распознавания (рис. 3), на которые влияют способы нанесения меток, свойства металла и освещение.

Для того, чтобы выбрать подходящие методы бинаризации, проводился вычислительный эксперимент, который сводился к тому, чтобы определить, сколько в результате будет распознано кодов. В эксперимент были включены изображения, охватывающие большинство возможных условий освещения, варианты углов кода, качества нанесения кода, качества и цвета поверхности. На основании этого выбрано несколько методов бинаризации, которые представлены в табл. 1.

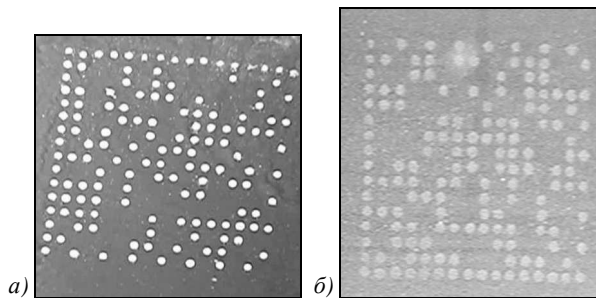


Рис. 3. Примеры условий распознавания промышленных кодов DataMatrix

Табл. 1. Методы бинаризации и настройки по обработке изображений

	Обработка перед бинаризацией	Бинаризация	Обработка после бинаризации
1	–	Блочная бинаризация	–
2	–	Блочная бинаризация	Дилатация
3	Размытие	Блочная бинаризация	Дилатация
4	Размытие	Пороговая бинаризация (170)	–
5	Размытие	Пороговая бинаризация (185)	–

Размытие перед бинаризацией осуществлялось медианным фильтром с размером окна 3×3. Дилатация использовалась после бинаризации с размером окна 3×3. Пороговая бинаризация задавалась фиксированно по уровню яркости 170 и 185 (по причине того, что освещение находится в некотором диапазоне, и данными значениями уровня порога можно достаточно качественно провести бинаризацию для большого количества изображений с кодами – яркость белых точек не должна падать ниже 170). Блочная бинаризация осуществлялась при работе с блоками изображения размером 40×40 пикселей. Пороговое значение яркости в блоке вычислялось на основе следующего алгоритма (в синтаксисе языка Си):

$$(M + k > p \ \&\& \ M + k < 255) ? t = M + k : t = p, \quad (2)$$

где M – среднее значение яркости в блоке, k – коэффициент отличия (использовалось значение 30), p – минимальное значение порога (использовалось значение 170), t – результирующее пороговое значение яркости для блока.

Однако в изображениях со сложным фоном даже все описанные методы бинаризации не позволяли качественно обработать изображение. Поэтому ещё одним используемым методом стало сравнение с эталонным закрашенным кругом, который характеризовал точку кода. Процедура использует скользящее окно размером $w \times h$ пикселей со вписанным в него кругом радиусом r , последовательно сравнивая все пиксели тестируемого изображения I для вычисления корреляции по следующей формуле:

$$R(x, y) = \frac{\sum_{x', y'} (T(x', y') I(x + x', y + y'))}{\sqrt{\sum_{x', y'} T(x', y')^2 \sum_{x', y'} I(x + x', y + y')^2}}, \quad (3)$$

где R – изображение результата сравнения, x и y – координаты на результирующем изображении, $0 \leq x' < w$ и $0 \leq y' < h$ – координаты в окне шаблона круга, T – изображение шаблона. На рис. 4а показан результат обработки изображения (рис. 2а). На рис. 4б – результат пороговой бинаризации. После этого вызывалась функция поиска контуров для локализации точек, и в центре каждого найденного контура восстанавливалась точка заданного радиуса (рис. 4в).

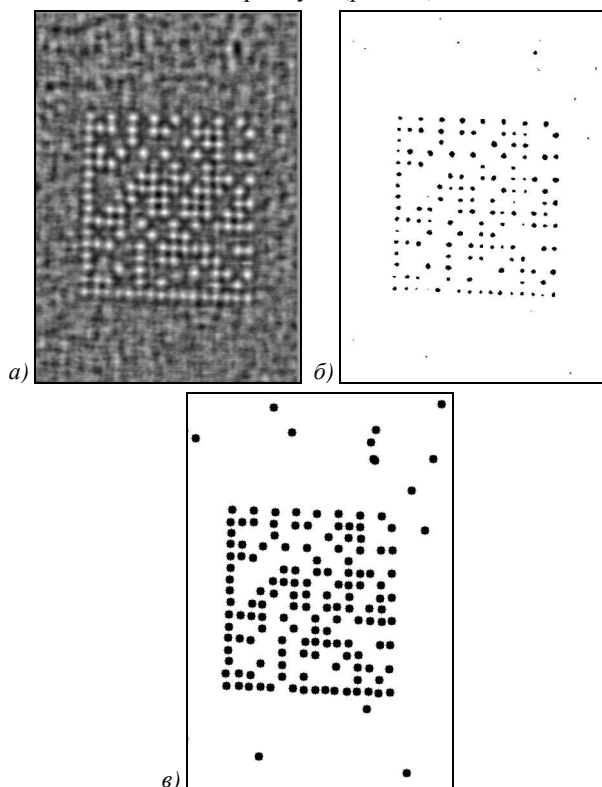


Рис. 4. Обработанное изображение с кодом DataMatrix методом корреляционного сравнения с эталоном круга (а), бинаризованное изображение (б), изображение с точками восстановленного размера (в)

При смене условий распознавания, когда ни один из перечисленных методов не был экспериментально опробован в подобных условиях (например, освещение более слабое и цвет точки кода начинается со значения яркости 150), может возникнуть ситуация, при которой ни один из перечисленных методов (в том числе в табл. 1) не подойдет. В такой ситуации необходимо провести дополнительный вычислительный эксперимент – формируется набор тестовых изображений для данного освещения, и перебираются возможные режимы бинаризации с тестами на их качество работы, после чего формируется список используемых бинаризованных изображений для данных настроек оборудования. В реальных условиях определенных режимов бинаризации оказалось достаточно, и после

года эксплуатации на различных предприятиях не потребовалось добавлять новые режимы.

2. *Поиск группы точек.* Из всех выделенных на изображении участков, которые могут считаться точками, необходимо выделить группу точек, которая относится к коду:

$$B = f_c(\text{Contours}(I_b)), \quad (4)$$

где группа точек B представлена в виде множества $\{p_1, p_2, \dots, p_i, \dots, p_N\}$, N – количество точек в группе, p_i – точка, представленная в виде (x, y) , f_c – функция, выбирающая подходящую группу точек из найденных контуров на изображении I_b .

Алгоритм выделения группы точек после бинаризации:

- поиск контуров, по размеру подходящих для точек кода (заранее задается некоторый диапазон), на этом же этапе можно сравнивать выделенный контур с окружностью (нет необходимости при сравнении с эталоном);
- дополнительная фильтрация шумов – удаление точек, которые значительно меньше среднего значения;
- для каждой из найденных точек определяется количество соседей – точек, которые находятся на расстоянии меньше некоторого порогового значения, пороговое значение определяется в зависимости от режима нанесения точек на поверхность и должно быть несколько больше максимального расстояния между центрами наносимых точек;
- разбиение точек по группам – если все точки можно перебрать (т.е. все укладываются в пороговое расстояние между точками), то тогда все точки принадлежат одной группе;
- формируется и выдается группа с наибольшим количеством точек (если кодов несколько на изображении, то можно выдавать несколько групп).

3. *Определение границ кода.* На данном этапе необходимо определить точную границу кода:

$$P = f_p(B), P = (p'_1, p'_2, p'_3, p'_4), \quad (5)$$

где f_p – функция, возвращающая границы кода в виде четырех крайних точек.

Алгоритм состоит из двух шагов:

- определение минимального ограничивающего прямоугольника любым из известных методов (различные библиотеки, в том числе и используемая библиотека OpenCV, поддерживают данную функцию), результат показан на рис. 5а;
- уточнение каждой из 4 линий: к каждой линии определяются ближайшие точки (не больше 5–8), для каждой комбинации пар точек генерируются линии, а линия, пересекающая максимальное количество точек, считается оптимальной границей (рис. 5б);
- по пересечениям линий определяются краевые точки.

4. *Определение наклона кода.* Поскольку ось камеры может быть не точно перпендикулярна плоскости кода, то будут изменяться размеры точек (ячеек) ко-

да. В этом случае код будет принимать вид не квадрата, а четырёхугольника, похожего на ромб или трапецию [3]. Для определения изменения размеров ячеек необходимо анализировать разреженные края кода DataMatrix (на рис. 5б – верхний и правый края). На рис. 6а показано изменение размера пар ячеек для случая, когда ось камеры практически перпендикулярна плоскости, где расположен DataMatrix код 16×16. На рис. 6б показано изменение размера пар ячеек для примера, когда камера существенно наклонена относительно оси.

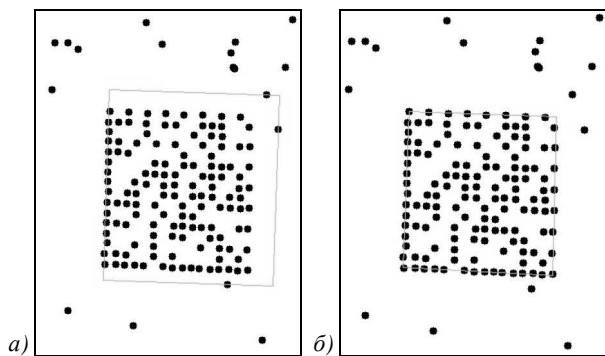


Рис. 5. Минимальный ограничивающий прямоугольник (а), уточнённый прямоугольник (б)

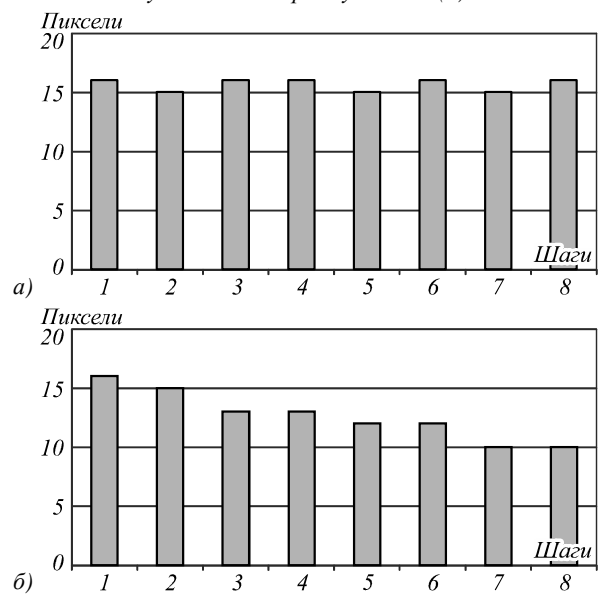


Рис. 6. Изменение размера в пикселях шагов (пар ячеек по разреженному краю кода) при перпендикулярной оси камеры к плоскости (а), при значительном наклоне камеры (б)

Любое изменение наклона должно фиксироваться для использования на следующем этапе. Поэтому по полученным данным (рис. 6) вычисляются линейные тренды изменения размеров для обеих сторон:

$$C = (L_1, L_2), \tag{6}$$

где L_1 и L_2 – формулы линий в следующем виде:

$$s_i = a * i + b, \tag{7}$$

где s – размер пары ячеек, i – номер пары, a и b – параметры линии.

На данном этапе также определяются размеры кода, например, 16×16 или 32×32 и т.п. Это делается по подсчёту тех же самых пар ячеек (рис. 6).

5. *Построение матрицы.* Для восстановления матрицы ячеек кода необходимо знать границы матрицы (этап 3), пропорции изменения размеров ячеек и количество ячеек (этап 4).

Согласно формулам линий изменений для каждой стороны кода определяются оптимальные размеры пар ячеек:

$$\sum_{i=1}^m s_i = len, \tag{8}$$

где m – общее количество пар (1/2 количества ячеек), len – длина стороны.

Несмотря на то, что формулы линии определяются для двух сторон, относительные изменения размеров пар ячеек применяются и для противоположных сторон.

Таким образом, каждая сторона размечается по ширине ячеек, т.е. определяются точки на каждой стороне. Соответствующие пары точек с противоположных сторон кода позволяют построить линии, которые будут формировать сетку для определения матрицы (рис. 7а).

Если детектируется код DataMatrix и он хорошо выделен после бинаризации, то матрица будет построена правильно (рис. 7а). Если произошли ошибки при бинаризации или участок изображения не является кодом, то матрица будет построена неправильно (рис. 7б). Чтение информации происходит по определению заполнения чёрными пикселями ячеек [3].

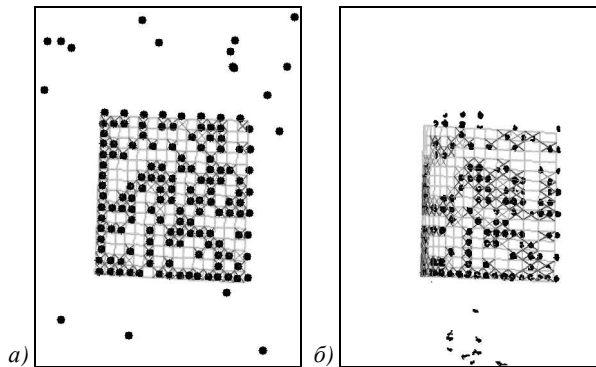


Рис. 7. Правильное построение матрицы после бинаризации шаблонным методом (а), неправильное построение матрицы после бинаризации с номером 4 из табл. 1 (б)

6. *Декодирование.* Этап восстановления информации связан с вызовом функции, отвечающей за декодирование Рида–Соломона. При сильном повреждении кода (или ложном кандидате на DataMatrix код) будет возвращена ошибка, в противном случае – текст. Существует достаточное количество бесплатных библиотек, позволяющих осуществить декодирование Рида–Соломона, например, libfec.

2. Результаты работы

Использование промышленных кодов DataMatrix происходит в сложных условиях, что означает перемен-

чивую среду распознавания и ограниченное время на принятие решения. Разработанный алгоритм был опробован и внедрён в автоматизированную систему по производству труб. Программное обеспечение разрабатывалось с учётом требований, описанных в работе [7].

Поскольку условия распознавания меняются (рис. 3), то для одних типов условий подходят одни виды бинаризации, для других – другие. Логично было бы предположить, что нужно попробовать распознать код, поочерёдно вызывая алгоритм распознавания с различными видами бинаризации, пока код не будет распознан. Однако в системах реального времени такое невозможно – операция должна быть выполнена в определённых временных рамках. И если учесть, что, помимо распознавания, вычислительная система должна выполнять и другие функции, то время распознавания должно быть минимизировано. Поэтому использование нескольких режимов бинаризации поочерёдно для одного кода должно быть жёстко ограничено. Например, 2 или 3 метода с учётом производительности вычислительной системы.

Для экспериментов были использованы 5 режимов бинаризации, указанных в табл. 1, и 4 бинаризации сравнения с эталоном закрашиваемого круга, радиусами 3, 4, 5 и 6 пикселей. Были использованы 3 различные группы кодов, различающихся по фоновой поверхности, освещению, размерам точек кодов. В сумме использовалось 500 различных кодов. В табл. 2 показаны результаты распознавания: 1-я группа – очень сложные для распознавания случаи (даже на взгляд человека практически невозможно распознать), 2-я группа – средние по сложности случаи, 3-я группа – самые простые для распознавания случаи.

Табл. 2. Достоверность распознавания для различных настроек методов бинаризации

Бинаризация	1-я группа	2-я группа	3-я группа
Табл.1 – №1	0,020	0,008	1,000
Табл.1 – №2	0,039	0,047	0,853
Табл.1 – №3	0,176	0,047	0,938
Табл.1 – №4	0,039	0,333	0,665
Табл.1 – №5	0,020	0,109	0,684
Эталон – 3 пикселя	0,412	0,667	0,000
Эталон – 4 пикселя	0,196	0,775	0,059
Эталон – 5 пикселей	0,098	0,698	0,327
Эталон – 6 пикселей	0,000	0,047	0,004

Из табл. 2 видно, что только для 3-й группы один из методов бинаризации даёт 100 % результатов распознавания. Здесь не говорится про ложные срабатывания, поскольку они отсутствуют за счёт кодирования Рида-Соломона. Для того, чтобы увеличить количество распознанных кодов, можно комбинировать методы бинаризации. Т.е. сначала осуществляется распознавание для одного вида бинаризации, а если код не найден, то для другого. На рис. 8 показано на примере выборки из 130 элементов 2-й группы, какими методами и сколько достигается правильных результатов распознавания. Если скомбинировать 6-й и 7-й методы, то достоверность повышается до 0,883 по сравнению с 0,775.

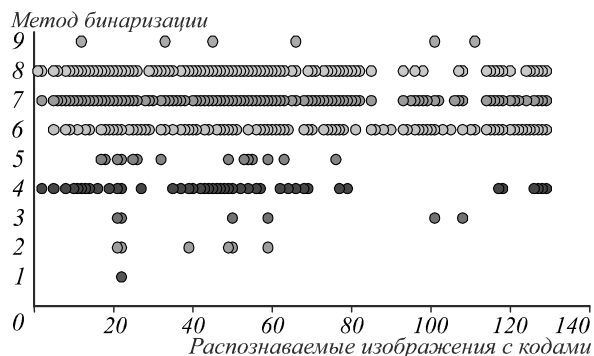


Рис. 8. Результаты распознавания тестовой выборки с использованием различных методов бинаризации (кружками показаны правильно распознанные образы)

В решаемой задаче условия менялись плавно – сначала на конвейере одна партия труб, а потом другая. Поэтому с каждым из используемых методов бинаризации необходимо связывать переменную, которая будет оценивать эффективность метода по количеству правильно распознанных кодов за некоторый интервал времени (или кадров). Например, использовать три метода бинаризации, а метод, которому соответствуют низкие результаты по распознанным образам, заменять на случайный метод, выбранный из доступных методов.

Разработанный алгоритм и программное обеспечение применяется в ООО «ИнТБуСофт» (<http://intbusoft.ru>) и распространяется для компаний, занимающихся автоматизацией производства, в том числе производства труб.

Литература

1. **Dita, I.** A scanning method for industrial data matrix codes marked on spherical surfaces / I. Dita, V. Gui, M. Ottesteanu, F. Quint // SITE'12 Proceedings of the 11th international conference on Telecommunications and Informatics, Proceedings of the 11th international conference on Signal Processing. – Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS) Stevens Point, 2012. – P. 38-42.
2. **Dita, I.** Using mean shift algorithm in the recognition of industrial data matrix codes / I. Dita, V. Gui, M. Ottesteanu, F. Quint // SITE'12 Proceedings of the 11th international conference on Telecommunications and Informatics, Proceedings of the 11th international conference on Signal Processing. – Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS) Stevens Point, 2012. – P. 174-179.
3. **Кручинин, А.Ю.** Алгоритм распознавания двумерных графических кодов с произвольным углом поворота и наклона камеры / А.Ю. Кручинин // Материалы седьмой всероссийской научно-практической конференции (с международным участием) «Современные информационные технологии в науке, образовании и практике». – Оренбург: ИПК ГОУ ОГУ, 2008. – С. 193-196.
4. **Кручинин, А.Ю.** Обнаружение искусственных бинарных объектов в сценах наблюдения на основе блочной сегментации изображений с использованием гистограммных признаков // Вестник компьютерных и информационных технологий. – 2011. – № 11. – С. 9-12.
5. **Yang, H.** Binarization of low-quality barcode images captured by mobile phones using local window of adaptive location and size / H. Yang, A.C. Kot, X. Jiang // IEEE

- Transactions on Image Processing. – 2012. – Vol. 21(1). – P. 418-425.
6. **Satish, M.** Edge Assisted Fast Binarization Scheme for Improved Vehicle License Plate Recognition / M. Satish, V.L. Lajish, S.K. Kopparapu // Communications (NCC), 2011 National Conference on. – 2011, 28-30 January. – P. 1-5.
 7. **Кручинин, А.Ю.** Особенности разработки программных систем распознавания образов реального времени // Автоматизация в промышленности. – 2011. – № 7. – С. 53-56.
3. **Kruchinin, A.Yu.** Algorithm of recognition of two-dimensional graphic codes with any angle of rotation and a chamber inclination // Materials of the seventh All-Russia scientifically-practical conference (with the international participation) «The Modern Inform Technology in a Science, Formation and Practice». – Orenburg: OSU, 2008. – P. 193-196. – (In Russian).
 4. **Kruchinin, A.Yu.** Detection of synthetic binary objects in the scenes observations on the basis of a block image segmentation using histogram features // Vestnik komp'yuternykh i informatsionnykh tekhnologii. – 2011. – V. 11. – P. 9-12. – (In Russian).
 5. **Yang, H.** Binarization of low-quality barcode images captured by mobile phones using local window of adaptive location and size / H. Yang, A.C. Kot, X. Jiang // IEEE Transactions on Image Processing. – 2012. – Vol. 21(1). – P. 418-425.
 6. **Satish, M.** Edge Assisted Fast Binarization Scheme for Improved Vehicle License Plate Recognition / M. Satish, V.L. Lajish, S.K. Kopparapu // Communications (NCC), 2011 National Conference on. – 2011, 28-30 January. – P. 1-5.
 7. **Kruchinin, A.Yu.** Specifics of development of software systems for real-time pattern recognition // Automation and Remote Control. – 2013. – Vol. 74, Issue 9. – P. 1599-1605.

References

1. **Dita, I.** A scanning method for industrial data matrix codes marked on spherical surfaces / I. Dita, V. Gui, M. Ottesteanu, F. Quint // SITE'12 Proceedings of the 11th international conference on Telecommunications and Informatics, Proceedings of the 11th international conference on Signal Processing. – Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS) Stevens Point, 2012. – P. 38-42.
2. **Dita, I.** Using mean shift algorithm in the recognition of industrial data matrix codes / I. Dita, V. Gui, M. Ottesteanu, F. Quint // SITE'12 Proceedings of the 11th international conference on Telecommunications and Informatics, Proceedings of the 11th international conference on Signal Processing. – Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS) Stevens Point, 2012. – P. 174-179.

INDUSTRIAL DATAMATRIX BARCODE RECOGNITION FOR AN ARBITRARY CAMERA ANGLE AND ROTATION

A.Yu. Kruchinin

Orenburg State University,
IntBuSoft Ltd

Abstract

We introduce an algorithm for the recognition of two-dimensional industrial codes DataMatrix that would allow one to identify codes applied by a shock-point or a drip method on the surface with a complex background. The algorithm can be adapted to a varying background by changing the binarization mode.

Key words: recognition of barcodes, DataMatrix, binarization image, control of the recognition process.

Сведения об авторе

Кручинин Александр Юрьевич, кандидат технических наук, доцент кафедры вычислительной техники и защиты информации Оренбургского государственного университета, генеральный директор ООО «ИнтБуСофт». Область научных интересов: обработка графических изображений, распознавание образов, компьютерное зрение, искусственный интеллект. Автор более 35 научных работ.

E-mail: kruchinin-al@mail.ru, kruchinin@vidikon.com.

Aleksandr Yurievich Kruchinin, candidate of the technical sciences, associate professor of Computing Equipment and Information Security department in Orenburg State University, general director of IntBuSoft Ltd. Research interests are computer graphics processing, pattern recognition, computer vision, artificial intelligence. Author of more than 35 scientific papers.



Поступила в редакцию 24 апреля 2014 г.