

МАРКИРОВКА ДИНАМИЧЕСКИХ ОБЪЕКТОВ С ПОМОЩЬЮ ШТРИХ-КОДА

И.О.Титов, Г.М.Емельянов

Институт электронных и информационных систем НовГУ, TitovIlya@yandex.ru

Рассматривается метод локализации и распознавания штрих-кода на динамическом объекте, используемый как средство маркировки. Процесс обработки происходит в режиме реального времени.

Ключевые слова: локализация, распознавание, штрих-код, маркировка, оптические искажения, Гауссиана

This article contains the description of bar code localization and recognition method. Bar code locates on dynamic object and uses as facility of the marking. The handling process occurs in real time mode.

Keywords: localization, recognition, bar code, marking, optical distortion, Gaussian

Введение

Задача локализации и распознавания штрих-кода появилась вследствие необходимости идентификации динамических объектов на одной видеосцене. При применении стандартных средств маркировки, таких как буквенные или числовые идентификаторы, возможны неточности, связанные как с оптическими искажениями (смазывание, отсутствие фокуса, зашумленность и небольшое разрешение исходного изображения), так и с физическими дефектами маркеров. Штрих-код же имеет несколько степеней защиты и гораздо лучше подходит для такого класса задач.



Рис.1. Расфокусированное изображение UPC-A штрих-кода

Существующие методы

Стандартные коммерческие технологии распознавания штрих-кодов [1-4] базируются на методах детектирования границ. Такие технологии используют только локальную информацию и при сильных размытия сигнала имеют высокий коэффициент ошибки. К тому же расчет наиболее истинного сигнала, из всех восстановленных вариантов, требует много времени, что недопустимо в системах реального времени. Методы, основанные на классификаторе Байеса [5], комбинируют известные данные о геометрической форме штрих-кода, включая все допустимые геометрические конфигурации и возможные геометрические искажения, а также данные угловых характеристик, основанных на гистограмме яркостей. Некоторые модификации содержат признак контрольного числа, который позволяет исправлять легкие ошибки, максимум в паре значений. Методы, основанные на таком подходе, весьма результативны и робастны по отношению к размытым исходным данным, но очень требовательны к вычислительным ресурсам.

Исходные данные

Исходные данные представляют собой одномерный сигнал, построенный на основе гистограммы интенсивности вдоль штрих-кода как усреднение сканирующих линий, проходящих параллельно на небольшом расстоянии друг от друга. Этот подход позволяет компенсировать ошибки, возникающие по причине шумов и искажений в одной из линий, и в то же время позволяет получать корректное усреднение при перспективных искажениях [1,4].

Базовые принципы метода

В общем случае задача состоит из следующих этапов:

- первичное обнаружение границ штрих-кода;
- подготовка исходных данных;
- определение размера одного кодового бита;
- выделение 12 блоков, содержащих информационные биты;
- нахождение наиболее соответствующего кода для каждого из 12 блоков;
- проверка контрольной суммы.

В силу малых размеров изображения границы значащих блоков необходимо вычислять в дробных величинах, иначе погрешность на границах блоков может быть недопустимо велика. Сдвиг границ в одном направлении на 0,5 точки приводит к тому, что происходит смещение границ всех значащих битов — следовательно, отдельный бит кода может быть взят с ошибкой на 25% своего размера, при значении 2 точки на один бит данных. Получается, что значение дискретного сигнала, приходящееся на граничную точку, по необходимости трактуется как принадлежащее обоим блокам в пропорции, соответствующей положению границы.

В таких условиях нахождение кодовых последовательностей, наиболее соответствующих исходным данным, проще всего производить методом аппроксимации реального сигнала эталонными данными (см. рис.2).

Необходимо построить эталонные сигналы для всех кодовых комбинаций, и выбрать тот вариант, который наиболее близок к истинному.

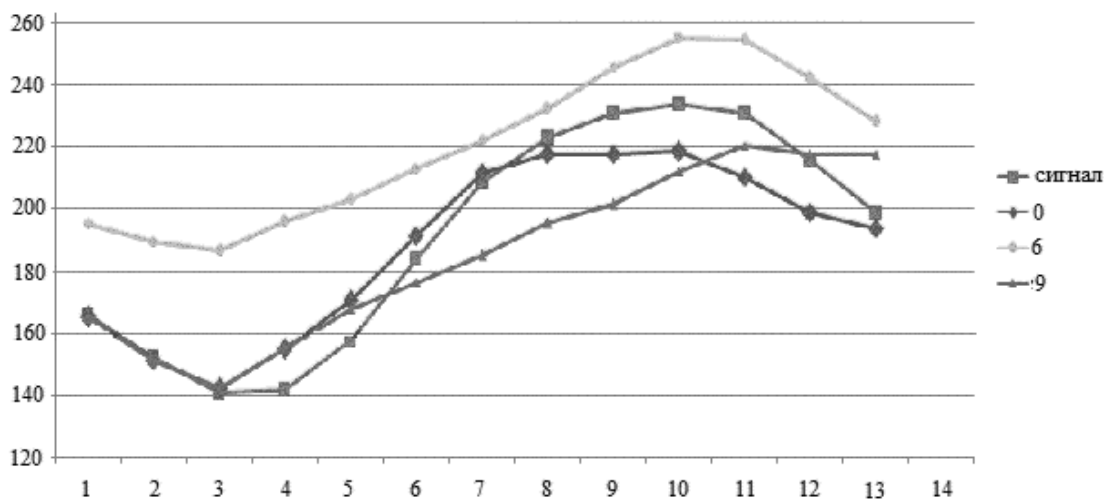


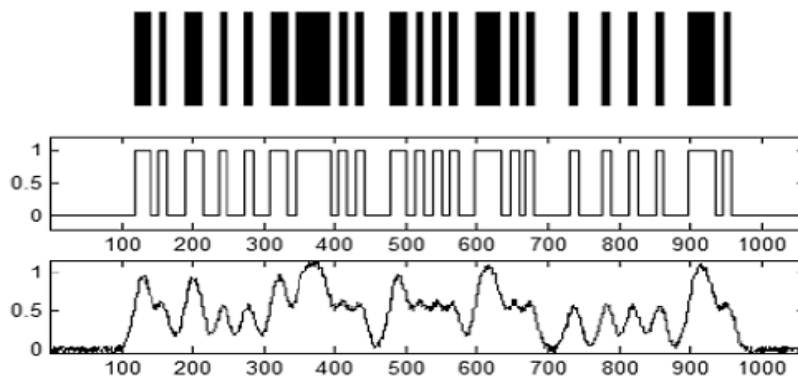
Рис.2. Аппроксимация исходного сигнала (0) значениями 0, 6 и 9

Размытие. Гауссиана

Если штрих-код при сканировании находится не в фокусе, изображение получается нерезким [2,3]. Сигнал можно представить следующим образом:

$$U_0 = G_{m,\sigma} \cdot U_{ид} + n,$$

где $G_{m,\sigma}$ — функция расфокусировки; $U_{ид}$ — идеально резкий сигнал; n — шум. На рис.3 представлены $U_{ид}$ и U_0 .

Рис.3. В центре идеальный сигнал $U_{ид}$, ниже — размытый сигнал U_0

Восстановление резкости такого изображения должно опираться на знание о природе размытия и представляет собой сложную вычислительную задачу (поиск оптимального исходного сигнала).

Общее число возможных кодовых комбинаций достигает 2^n (где n — ширина блока с одной цифрой, ~14-16), и даже при наличии хорошего критерия оптимизации задача поиска наилучшего остается весьма трудоемкой. В то же время количество корректных комбинаций для каждого блока ограничено.

Более рационально применение метода аппроксимации эталонным сигналом, дополненного размывающим фильтром. Реальные данные в каждом блоке сравниваются поочередно с эталонами, пропущенными через фильтр. Экспериментально проверено, что реальные данные оказываются весьма близки

к эталону, полученному Гауссовым размытием идеальных данных.

Таким образом, дискретный фильтр Гаусса применяется к каждой точке одномерного сигнала (значение сигнала в точке обусловлено значениями точек в некоторой окрестности; значения накапливаются с соответствующими весами):

$$x'_i = \sum_{k=-r}^r x_{i+k} \cdot G_{m,\sigma}(k),$$

где $G_{m,\sigma}(k)$ — веса для фильтра; r — радиус окна фильтра.

Обнаружение и уточнение границ

Локализовать штрих-код можно на основе краевых пикселей и направления градиента. Для этого применяется оператор Собеля [2-4]. По результатам применения оператора получается первая оценка границ штрих-кода.

Метод Собеля дает приблизительное положение границ. Ошибка в нахождении границ может достигать нескольких точек. Для более точного определения границ строится битовая последовательность, представляющая собой сторожевой паттерн и пустое («белое») поле значительной ширины (чтобы не перепутать сторожевой паттерн с такой же кодовой

подпоследовательностью — 0 101, — следующей за ним). Левый паттерн — 000101, правый — 101000.

В качестве критерия наилучшего положения границы служит минимум ошибки при сравнении реальных данных с построенным эталоном паттерна. Оценка аппроксимации при уточнении каждой из границ производится отдельно.

Критерий для левой границы выглядит следующим образом: если $best_code[i]=k_i$, где

$$diff[i, k_i] = \min_j (diff[i, j]), j = 0, \dots, 9, i = 1, \dots, 6,$$

то наилучшим называется такое смещение границы δ_i (т.е. $start_i = start + \delta_i$), что $\forall \delta \in [-r, r]: \max_i (diff_{\delta_i}[i, k_i]) \leq \max_i (diff_{\delta_i}[i, k_i])$. Т.е. не существует другого допус-

тимого сдвига в пределах $[-r, r]$, для которого максимальная ошибка в блоках опорного штрих-кода превышает такую же ошибку при сдвиге границы на δ_i .

Для правой границы критерий засчитывается аналогично.

Построение и размытие исходных сигналов

После вычисления границы блоков для каждого из них производится аппроксимация сигнала эталонными значениями. Каждая итерация состоит из следующих шагов.

1. Кодовая последовательность (7 бит) масштабируется по ширине блока и нормируется по диапазону амплитуд реального сигнала:

$$curr'_j[x] = Min + Delta \cdot code_j \left[\frac{(x - Block_start[i]) \cdot 7}{Block_size} \right],$$

$$Delta = Max - Min,$$

где $curr'$ — масштабированный эталонный сигнал, нормированный по амплитудам; Max — максимальная амплитуда; Min — минимальная амплитуда сигнала; $code_j$ — 7-битовая кодовая комбинация для цифры j ; $x \in [Block_start[i], Block_stop[i]]$.

2. Фильтр Гаусса применяется к полученным эталонным данным в диапазоне $[Block_start[i], Block_stop[i]]$:

$$curr_j[x] = \sum_{k=-r}^r (curr'_j[x+k] \cdot G_{m,\sigma}(k)),$$

$$x \in [Block_start[i], Block_stop[i]].$$

3. Вычисляется ошибка данной аппроксимации, исходя из критерия наилучшего варианта (см. ниже).

Критерий поиска наилучшего варианта

Наилучшим для i -го блока считается вариант, дающий минимальное значение средневзвешенной квадратичной ошибки на интервале $(Block_start[i], Block_stop[i])$, т.е. $best_code[i] = k$, где

$$diff[i, k] = \min_j (diff[i, j]), j = 0, \dots, 9,$$

$$diff[i, j] = \sum_{x=Block_start[i]}^{Block_stop[i]} (data[x] - curr_j[x])^2.$$

Таким образом, на i -м блоке вычисляется ошибка $diff[i, j]$ для всех вариантов значения j (цифр от 0 до 9). В качестве наилучшего варианта выбирается значение j , для которого ошибка минимальна.

Эвристическое исправление ошибок

Штрих-код считается прочитанным верно, если контрольная сумма сходится, а значение ошибки для каждой из цифр последовательности не превышает определенного предела:

$$diff[i, k_i] \leq Delta \cdot C \quad (C \text{ — константа, примерно равная } 1, \text{ подбираемая экспериментально}),$$

$$diff[i, k_i] = \min_j (diff[i, j]), j = 0, \dots, 9, i = 1, \dots, 12.$$

Проверка на размер ошибки обеспечивает фильтрацию изображений, на которых был обнаружен «ложный» штрих-код или штрих-код с ошибками в границах. В этом случае средние ошибки в кодовых блоках будут достаточно велики.

Накапливаемые данные для эвристики

Для оценки эвристики используется таблица средневзвешенных квадратичных ошибок для допустимых кодовых комбинаций в блоках. Для каждого допустимого варианта из таблицы вычисляются относительные отклонения от лучшего текущего варианта:

$$mindiff[i] = \min_i (diff[i, j]), j = 0, \dots, 9,$$

$$ratio[i, j] = \frac{(diff[i, j] - mindiff[i])}{mindiff[i]}, i = 1, \dots, 12, j = 0, \dots, 9.$$

Таким образом, $ratio[i, j] \geq 0$ для любого j , $ratio[i, j] = 0$ для варианта с минимальной ошибкой, и чем выше ошибка, тем выше относительная ошибка.

Накопление таких данных об относительной ошибке позволяет накапливать результаты аппроксимации по нескольким изображениям (2 и более). В этом случае $ratio[i, j]$, полученное для аппроксимации текущего реального сигнала, добавляется к сумме значений, накопленных на предыдущих итерациях.

По минимальным значениям $ratio[i, j]$ строится первичный вариант баркода, т.е. формируется такой цифровой код $\{c_1, \dots, c_{12}\}$, для которого

$$ratio[i, c_i] = \min_j (ratio[i, j]), j = 0, \dots, 9.$$

Если полученный код успешно прошел проверку контрольной суммы, этот код является искомым результатом. Если же он не прошел эту проверку, производится поиск ближайшей корректной последовательности. Для этого необходимо найти такой код $\{c'_1, \dots, c'_{12}\}$, что для него сходится контрольная сумма (т.е. он корректен), и этот код является ближайшим к $\{c_1, \dots, c_{12}\}$ корректным кодом. Т.е.

$$\exists \{c'_1, \dots, c'_{12}\} \text{ — корр.},$$

$$\forall \text{ корр. } \{c'_1, \dots, c'_{12}\} : \sum_{i=1}^{12} ratio[i, c'_i] \leq \sum_{i=1}^{12} ratio[i, c_i].$$

Таким образом, задача сводится к поиску такой корректировки опорного кода, которая обладает верной контрольной суммой и минимальным совокупным отклонением.

Выводы

Предложенный метод был апробирован на реальных данных и показал устойчивую и уверенную работу в условиях сильной расфокусировки изображений, а также их малого разрешения. Скорость работы и малая ресурсоемкость метода позволяют использовать его в системах реального времени.

-
1. Adair Jeffrey. Locating, Tracking, and Interpreting Ean-13 Bar Code Waveforms in a Two-Dimensional Video Stream Jeffrey —

www.acm.org/src/subpages/gf_entries_06/JeffreyAdair_src_gf06.pdf

2. Wittman Todd. Lost in the Supermarket: Decoding Blurry Barcodes — www.math.ucla.edu/~wittman/thesis/barcodes_siam.pdf
3. Liyanage J.P. Efficient Decoding of Blurred, Pitched, and Scratched Barcode Images — www.cs.ucf.edu/~janaka/projects/barcode/barcode_paper.pdf
4. Tekin Ender, Coughlan James. A Bayesian Algorithm for Reading 1D Barcodes — www.ski.org/rerc/Coughlan_lab/General/.../barcodes-TekinCoughlan.pdf
5. Chai Douglas, Hock Florian. Locating and Decoding EAN-13 Barcodes from Images Captured by Digital Cameras — www.icics.org/2005/download/P0840.pdf