

An Image-based Postal Barcode Decoder with Missing Bar Correction

Peng-Hua Wang* and Jia-Wei Ciou†

*Department of Communication Engineering, National Taipei University, New Taipei City, Taiwan, R.O.C.
E-mail: phwang@mail.ntpu.edu.tw, Tel: +886-2-26748565

†Graduate Industrial Collaboration R&D Master Program of Information and Communication Technology,
National Taipei University, New Taipei City, Taiwan, R.O.C.
E-mail: david3867572@gmail.com, Tel: +886-2-26748565

Abstract—In this paper, we propose an image-based postal barcode decoding algorithm. The proposed algorithm is capable of correcting missing bars in a symbol. Postal barcodes in general belong to bar length modulated symbology. Usually, postal barcodes use a check digit to detect errors. It is possible for some postal code symbologies not only to detect errors, but also to correct them by using the uniqueness of the pattern of a digit. In this paper, we present a general method of correcting errors caused by missing bars. In order to validate our method, a POSTNET (Postal Numeric Encoding Techniques) decoder is implemented. Experimental results show that the proposed algorithm can correct one missing bar in a digit, and at most ten missing bars in a POSTNET symbol.

I. INTRODUCTION

Barcodes are widely used for automatic data entry, in industrial, commercial, and personal applications. There are roughly three types of barcode symbologies: linear, stacked, and matrix symbologies [1]. In a linear barcode, information is encoded into width, height, or position of bars and spaces. Any row carries the same information. The matrix barcodes encode the information into positions of data cells in a region. In the stacked symbologies, several linear codes are used. These linear codes are typically width modulated.

Most postal barcodes belong to linear symbologies. A postal code may use one or two position modulated rows to carry information, and an additional clock row, called the tracker, for measuring the positions of every bar. Modern symbologies have the property of “self-clocking”. Extra clock row is not needed. However, in some old symbologies, the clock row may increase the reliability of barcode reading, or simplify the decoding algorithm.

Typically, postal barcode symbologies define error checking method for error detection. For example, POSTNET (Postal Numeric Encoding Techniques) symbology [1] uses a check digit to detect errors. The sum of all digits values including the check digit is a multiple of 10. By using the error checking mechanism, we can reject the wrong barcodes caused by poor image quality. In a common decoding flow, images are captured and decoded until at least one symbol passes the error checking. Then the decoder claims a decoding success. However, if any bar is missing, the decoder can only reject the symbol, no matter how good the image quality is.

There is large amount of literature on every aspects of barcodes. In [1], a comprehensive and detailed introduction to barcodes and symbologies is provided. Fundamental concepts on barcodes, mainly for linear codes, can be found in [2]. Concepts on matrix symbologies are provided in [3]. Methods of processing linear codes can be found in [4], [5], [6]. In [7], [8], algorithms for reading matrix codes are proposed.

In this paper, we propose a method of decoding postal symbology with missing bars. The key concept of decoding is the uniqueness of the bar pattern of a digit. The pattern of a digit is a group of bars. The pattern is defined in the specification of symbology. If we can find one bar from the remaining bars in a group, then we can correct a missing bar in a digit. To show the feasibility of our method, we build real-time decoders that can decode missing bars. In this paper, we apply our method to a POSTNET decoder as our first implementation.

The rest of the paper is organized as follows. In Section 2, we introduce the proposed method in details. In Section 3, we introduce the specification of POSTNET symbology, including its structure and encoding rule. In Section 4, we briefly describe our implementation and experiment results. Section 5 concludes the paper.

II. PROPOSED METHOD

In this section, we will explain the proposed method in details. Fig. 1 shows the flow of our algorithm. The algorithm is designed to correct missing bars in symbols containing ascender and tracker. Figs. 2 and 4 show the definition of the ascender and tracker. However, our algorithm can be applied to symbols having not only ascender and tracker, but also descender [1]. Modification will be outlined in the following subsections.

A. Input Image

In general, an image may contain not only the barcode symbol but also other objects such as texts or figures. It is possible that several barcode symbols are presented in the same image. Therefore, in general some preprocessing task is needed in order to segment the target symbol for decoding. In this paper, we focus on the improvement of the decoding process, and assume that the input image contains only one

normalized barcode symbol. A normalized barcode symbol means that its bars are in vertical direction.

The input image is further assumed to be two-valued. The black pixels are objects, that is, the bars, and the white pixels are background. If the image captured by a camera is a color one, conversion from the color image to a gray-scaled one is performed [9]. The gray-scaled image is then converted into black-and-white image by thresholding. Let $g(m, n)$ be the gray-scaled $M \times N$ image and $b(m, n)$ be the corresponding black-and-white image for $1 \leq m \leq M$, $1 \leq n \leq N$, where m is the index of row and n is the index of column. Thresholding can be performed globally by

$$b(m, n) = \begin{cases} 1, & g(m, n) \geq T; \\ 0, & g(m, n) < T, \end{cases}$$

where T is the global threshold. Refer to [9] for methods of thresholding.

B. Locate the Ascender and Tracker

The first step in our algorithm is to locate the ascender and tracker. This step is indicated by the “*Locate A/T*” block in Fig. 1. Typically, a normalized postal barcode symbol is composed of two tracks, the ascender and tracker, such as POSTNET barcode [1], or of three tracks, the ascender, tracker and descender, such as Royal Mail 4-state code [1]. There are one or more rows in each track. The task of this step is to segment these tracks and extract the information contained in each track.

We propose a simple approach to searching for a “representative” row in the ascender, tracker or descender. A representative row is defined as following. Let $p(m)$, $1 \leq m \leq M$ be the number of black pixels in the m -th row, i.e.,

$$p(m) = \sum_{n=1}^N (1 - b(m, n)).$$

Then the k -th row is representative if

$$p(k-1) = p(k) = p(k+1).$$

If several rows are representative, satisfying

$$k_1 < \dots < k_{r-1} < k_r < \dots < k_{s-1} < k_s < \dots,$$

and

$$p(k_1) = \dots = p(k_{r-1}) < p(k_r) = \dots = p(k_{s-1}),$$

then we take k_1 -th row as the representative row for ascender, and k_r -th row as the representative row for tracker. If, in addition,

$$p(k_r) = \dots = p(k_{s-1}) > p(k_s) = p(k_{s+1}) = \dots,$$

then k_s -th row is representative for descender. Therefore, we can extract barcode information from the row $b(k_1, n)$ (ascender), the row $b(k_r, n)$ (tracker), and the row $b(k_s, n)$ (descender) for $1 \leq n \leq N$. Fig. 2 shows the waveform of an ideal $p(m)$. There are two flat regions, in which the first one is the ascender, and the second one is the tracker.

C. Analyze Space Widths

In this subsection, we will explain the task of the “*Analyze S/W*” block in Fig. 1, where “*S/W*” stands for space width. In the previous step, we have find one row $b(k_1, n)$ in ascender, one row $b(k_r, n)$ in tracker, and one row $b(k_s, n)$ in descender, if it exists. In the following discussion, for convenience, we denote $a(n) \triangleq b(k_1, n)$, $t(n) \triangleq b(k_r, n)$, and $d(n) \triangleq b(k_s, n)$ for ascender, tracker, and descender, respectively. For a symbol without any bar missing, we can extract data by firstly locating the black pixels in $t(m)$, then looking up the corresponding values in $a(m)$ and $d(m)$. These values can be used for decoding. However, if bars are missing, these values can not be used for decoding because some data are lost.

We propose to locate the missing bars and then find the missing values. Since missing bars result in abnormally wide space, it is reasonable to analyze the space width. Our approach is to measure the space width in $t(m)$, then calculate its average. Specifically, if $e_0 < e_1 < e_2 < e_3 < \dots$ satisfies

$$t(e_{2i-1} - 1) = 0, t(e_{2i-1}) = 1, t(e_{2i} - 1) = 1, t(e_{2i}) = 0, \dots$$

for $i = 1, 2, \dots, I$. That is, there is a rising edge at e_{2i-1} , and a falling edge at e_{2i} . The average space width can be computed by

$$\bar{w}_s = \frac{1}{I} \sum_{i=1}^I (e_{2i} - e_{2i-1})$$

Fig. 3 shows the waveform of the representative row in the tracker. A bar is missing between e_5 and e_6 in which e'_5 and e'_6 indicate the edges of the missing bar.

D. Detect Missing Bars

Now, we can locate the position of the missing bar based on \bar{w}_s . This step is indicated by the “*Detect M/B*” block in Fig. 1, where “*M/B*” stands for missing bar. Our method is to compare the space length with \bar{w}_s . If a space length $e_{2i} - e_{2i-1} > \bar{w}_s$, then there exists a missing bar between e_{2i} and e_{2i-1} .

E. Fill Missing Bars

After locating the missing bar in the tracker, it is possible to find the missing values in the ascender or descender. The step is indicated by the “*Fill M/B*” block in Fig. 1. Because the step may depend on the symbology, we use POSTNET code for demonstrating. Summary of POSTNET specification is given in the next section. The key to find the value of missing bars is the uniqueness of patterns. There are five bars in the pattern of a digit, two are long and three are short. In other words, we can group successive five bars for a digit in the tracker. Then there should be two bars and three spaces in the corresponding positions of ascender. If one bar is missing, we can find the missing value from the remaining bars and space in ascender.

Specifically, let $\hat{a}(i)$ be the information carried in the ascender. $\hat{a}(i) = 1$ if i -th bar is long and $\hat{a}(i) = 0$ if i -th bar is short. From Fig. 3, we know that i -th bar is located

between e_{2i-2} and e_{2i-1} . If no bar is missing between e_{2i-2} and e_{2i-1} , then $\hat{a}(i)$ can be estimated by

$$\hat{a}(i) = \begin{cases} 1, & a(m_i) = 0; \\ 0, & a(m_i) = 1, \end{cases}$$

where $m_i = (e_{2i-2} + e_{2i-1})/2$. That is, m_i is the middle point of e_{2i-2} and e_{2i-1} , rounding to the nearest integer if necessary.

If we know that there is a missing bar between e_{2i-2} and e_{2i-1} , then we can not find the value of $\hat{a}(i)$ from $a(m_i)$ since we do not know if this bar is long or short. However, from the group of five bars that the i -th bar is located, we can find the value of $\hat{a}(i)$. Specifically, let $\{\hat{a}(i'), \hat{a}(i'+1), \hat{a}(i'+2), \hat{a}(i'+3), \hat{a}(i'+4)\}$ form a group of five values corresponding to a digit. If any one is unknown, we can still find its value since in this group there are two ones and three zeros.

F. Decode

After finding the values in the ascender and in the descender, we can decode the barcode information according to its specification. In the next section, we will give the specification of POSTNET code which is used for our implementation.

III. POSTNET SPECIFICATION

POSTNET (Postal Numeric Encoding Techniques) is a symbology that encodes numeric digits only. The symbol can encode 5, 6, or 9 digits. Each digit consists of five bars, including two long bars and three short ones. In other words, POSTNET code can be regarded as a bar length modulated code. Table I lists all the ten patterns in which “1” represents a long bar and “0” represents a short one. Let bits b_4, b_3, b_2, b_1, b_0 form a pattern from left to right. The digit value v can be calculated by

$$v = 7b_4 + 4b_3 + 2b_2 + b_1 \mod 11.$$

b_0 is a check bit in order to keep two “1”s in the five bits.

Fig. 4 shows the symbol structure of a five-digit POSTNET code. The symbol begins in the start bar S_1 , it is a long bar, and end in the stop bar S_2 , it is also a long bar. The patterns of 5 data digits D_1, D_2, D_3, D_4 and D_5 , the check digit C , are printed between the start bar and the stop bar. The values of digits and the check digit satisfy

$$D_1 + D_2 + D_3 + D_4 + D_5 + C = 10K$$

where K is an integer. Refer to [1] for more details.

IV. EXPERIMENTAL RESULTS

We implement the proposed algorithm for correcting missing bars in a POSTNET symbol. The program is compiled in C and executed on a personal computer with Window 7 system. Experimental result shows that the proposed algorithm can correct and decode the symbols that have one missing bar in one digit. That is, for k -digit symbol, the proposed algorithm can correct up to $k+1$ missing bars, k for the digits, and one for the check digit.

Fig. 5 shows a set of five-digit POSTNET samples with missing bars. The data is 53498 and the check digit is 1. In Fig. 5(a), one bar is missing in the pattern of digit 5. In Fig. 5(b), two bars are missing, one in the pattern of digit 8, and another in digit 5. In Fig. 5(c), three bars are missing, in digits of 3, 4 and 8. In Fig. 5(d), each pattern of 5, 3, 4, 9 has one missing bar. Finally, in Fig. 5(e), the pattern of every digit has one missing bar. Our method can successfully decode these images. Actually, we have applied the proposed method to decode 5-, 6- and 9-digit POSTNET symbols. The symbols are successfully decoded if at most one bar is missing is a digit.

V. DISCUSSIONS AND CONCLUSIONS

In this paper, we propose a method of decoding postal barcodes with missing bars. The method is applied to decode the POSTNET code. Experimental results validates the proposed method. Our method is originated in the uniqueness of length modulated codes. For POSTNET codes, there are two long bars and three short bars in the pattern of any digit. Therefore, we can find any one from the remaining four. The concept can be applied to other symbology.

The proposed algorithm can be used for correcting several bars missing in a symbol. The following conditions are not dealt with by the proposed algorithm:

- 1) start bar or stop bar is missing,
- 2) two bars are missing in one digit, and
- 3) two adjacent bars are missing.

Currently, an extra bar length checking routine (not presented in Fig. 1) can report a wrong code and stop decoding when any one condition above occurs. However, it is possible to decode symbols in these conditions, by carefully analyzing the uniqueness of length modulated codes, and by more advanced algorithm of space width analysis.

REFERENCES

- [1] R. C. Palmer, *The Bar Code Book*, 5th ed. Trafford Publishing, 2007.
- [2] T. Pavlidis, J. Swartz, and Y. P. Wang, “Fundamentals of bar code information theory,” *IEEE Computer*, vol. 23, no. 4, pp. 74–86, Apr. 1990.
- [3] —, “Information encoding with two-dimensional bar codes,” *IEEE Computer*, vol. 25, no. 6, pp. 18–28, June 1992.
- [4] E. Joseph and T. Pavlidis, “Bar code waveform recognition using peak locations,” vol. 16, no. 6, pp. 630–640, June 1994.
- [5] D. Chai and F. Hock, “Locating and decoding ean-13 barcodes from images captured by digital cameras,” in *Fifth International Conference on Information, Communications and Signal Processing (ICICS)*, 2005, pp. 1595–1599.
- [6] R. Shams and P. Sadeghi, “Bar code recognition in highly distorted and low resolution images,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, 2007, pp. 1–737.
- [7] T.-J. Chin, H. Goh, and N.-M. Tan, “Exact integral images at generic angles for 2d barcode detection,” in *19th International Conference on Pattern Recognition (ICPR)*, 2008, pp. 1–4.
- [8] N. Normand and C. Viard-Gaudin, “A two-dimensional bar code reader,” in *Proceedings of the 12th IAPR International Conference on Pattern Recognition*. IEEE, 1994, pp. 201–203.
- [9] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed. Pearson Education Inc, 2010.

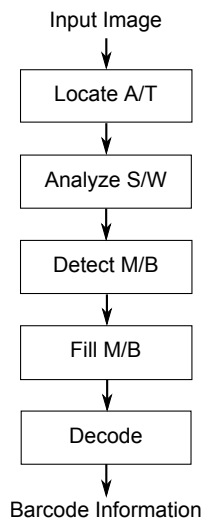


Fig. 1. Block diagram of the proposed algorithm

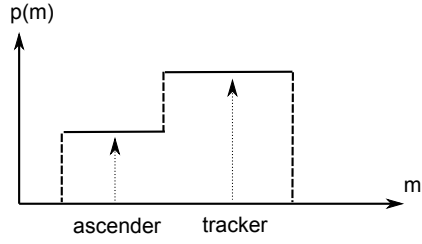


Fig. 2. Number of black pixels in each row of a normalized POSTNET image. The first flat region is the ascender, and the second one is the tracker.

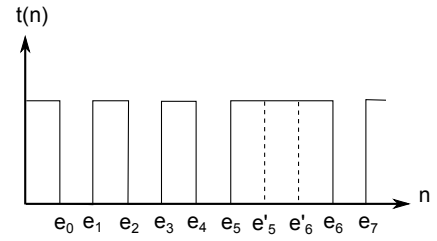


Fig. 3. The waveform of the representative row in the tracker. There is a bar missing between e_5 and e_6 . e'_5 and e'_6 are the edges of the missing bar.

TABLE I
ENCODING RULE OF THE POSTNET SYMOLOGY.

Data	Pattern
0	11000
1	00011
2	00101
3	00110
4	01001
5	01010
6	01100
7	10001
8	10010
9	10100

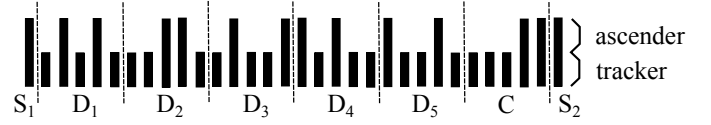


Fig. 4. Structure of the POSTNET symbol.

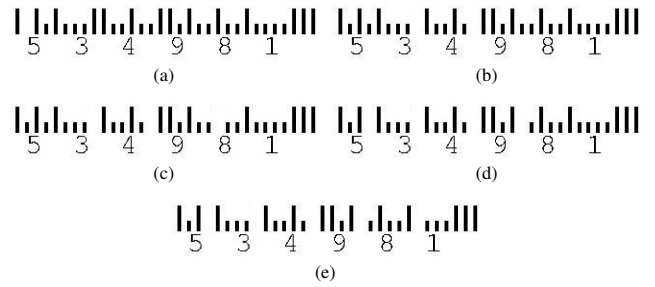


Fig. 5. Sample test images. (a) 1 bar missing. (b) 2 bars missing. (c) 3 bars missing. (d) 4 bars missing. (e) 5 bars missing.