Open access • Proceedings Article • DOI:10.1109/ACPR.2013.17

Robust Angle Invariant 1D Barcode Detection — Source link

Alessandro Zamberletti, Ignazio Gallo, Simone Albertini

Institutions: Applied Science Private University

Published on: 05 Nov 2013 - Asian Conference on Pattern Recognition

Topics: Barcode, Object detection, Mobile device and Mobile computing

Robust Angle Invariant 1D Barcode Detection

Alessandro Zamberletti Ignazio Gallo Simone Albertini
Department of Theoretical and Applied Science
University of Insubria
Via Mazzini 5, 21100, Varese, Italy
ignazio.gallo@uninsubria.it

Abstract—**Barcode reading mobile applications that identify products from pictures taken using mobile devices are widely used by customers to perform online price comparisons or to access reviews written by others. Most of the currently available barcode reading approaches focus on decoding degraded barcodes and treat the underlying barcode detection task as a side problem that can be addressed using appropriate object detection methods. However, the majority of modern mobile devices do not meet the minimum working requirements of complex general purpose object detection algorithms and most of the efficient specifically designed barcode detection algorithms require user interaction to work properly. In this paper, we present a novel method for barcode detection in camera captured images based on a supervised machine learning algorithm that identifies one-dimensional barcodes in the two-dimensional Hough Transform space. Our model is angle invariant, requires no user interaction and can be executed on a modern mobile device. It achieves excellent results for two standard one-dimensional barcode datasets: WWU Muenster Barcode Database and ArTe-Lab 1D Medium Barcode Dataset. Moreover, we prove that it is possible to enhance the overall performance of a state-of-the-art barcode reading algorithm by combining it with our detection method.**

I. Introduction

In the last few years, online shopping has grown constantly, and so have the number of customers that buy online using their smartphones or tablets. Those devices often integrate high quality cameras; as such, many researchers and companies focused on solving the problem of identifying products shown in camera captured images on the basis of their visual features [1]. However, the task of recognizing both the brand and the model of a product in a real world image has yet to be efficiently solved; this is mostly due to the large number of issues that need to be addressed when using camera captured images, such as poor light conditions or occlusions. An easier way to approach the object identification task in the field of e-commerce lies in exploiting the barcodes that nowadays appear on almost every item in the market: since each barcode univocally identifies a product, it is possible to precisely recognize an object just by detecting and decoding its barcode [2]. While both the detection and the decoding tasks have already been exhaustively faced for two-dimensional barcodes (e.g., Quick Read codes) [3]–[5], the same does not hold for one-dimensional (1D) barcodes, even though Universal Product Codes (UPC) and European Article Numbers (EAN) are widely diffused all over the world.

The task of reading 1D barcodes from camera captured images has been approached in different ways [6]–[12]. Most of the currently available barcode reading mobile applications read the gray intensity profile of a line in the processed image, thus they usually require the user to place the barcode in a specific position within the camera screen [12]. Some industrial approaches obtain excellent results using hardware implementations of their barcode reading softwares [13] but they usually exploit some prior knowledge related to the specific domain, e.g., the dimension and the position in which a barcode may appear inside the processed image. Other works propose different techniques of decoding 1D barcodes to deal with camera related issues, such as poor light conditions or lack of focus [9]–[11].

Overall, most of the works presented in literature mainly address the barcode decoding phase and treat the underlying barcode detection task as a side problem. Nonetheless, we argue that the task of detecting multiple arbitrary rotated barcodes in real world images is crucial to reduce the amount of user interaction involved in the subsequent decoding process. Moreover, real time angle invariant barcode detection algorithms may be exploited by automated systems to identify products without defining placement or ordering constraints. Obviously, 1D barcodes may be effectively detected by general purpose object detection methods, such as the one presented by Lempitsky *et al.* [14], however this is not an optimal solution since most of the interesting applications of barcode reading algorithms lie in the mobile field and the majority of currently available mobile devices do not meet the minimum working requirements of those object detection approaches.

In this paper, we propose an angle invariant method for barcode detection in camera captured images based on the properties of the Hough Transform [15]. A properly trained supervised machine learning model identifies the rotation angle of every barcode in real world images by analyzing their Hough Transform spaces and a subsequent phase detects the bounding boxes surrounding those barcodes. We prove that our method can obtain excellent results for three different 1D barcode datasets and that it is also effective in detecting barcodes that are twisted, partially occluded or illegible due to reflections. The main novelties of our approach are that: (i) it detects the exact positions and rotation angles of 1D barcodes without exploiting prior knowledge, thus it does not require any user interaction, (ii) the bounding boxes identified by our model can be exploited by a subsequent decoding phase to efficiently read barcodes without wasting any time searching for them in the processed image. We publicly release the

Fig. 1. Examples showing the barcode bounding boxes detected by the proposed method for arbitrary rotated 1D barcodes appearing in camera captured images; the detected bounding boxes are correct even when the barcodes are twisted, occluded or partially illegible due to reflections.

source code used in our experiments as it can be used by most of the barcode reading algorithms presented in literature and we present a new dataset specifically designed to evaluate the performances of angle invariant 1D barcode detection methods[1].

II. Related Works

A. Hough Transform

The classical Hough Transform [15] is a feature extraction technique commonly used in image processing and computer vision for the detection of regular shapes such as lines, circles or ellipses. The Hough Transform for lines detection adopts a voting procedure to identify the set of linear shapes L in a given image I. The normal form equation of a generic line $l \in L$ in I can be defined as follows:

$$\rho = x cos\theta + y sin\theta \tag{1}$$

where $\rho \geq 0$ is the distance of l from the origin of I and $\theta \in [0, 2\pi)$ is the angle of l with the normal. Let the two-dimensional Hough Transform space H be the (ρ, θ) plane, for an arbitrary point $(x_i, y_i) \in I$, Eq. (1) corresponds to a sinusoid in H. If two points $(x_0, y_0), (x_1, y_1) \in I$ belong to the same line l, their corresponding sinusoids intersect in a point $(\rho_l, \theta_l) \in H$. The same holds true for all the points of l. Note that the coordinates of the point $(\rho_l, \theta_l) \in H$ correspond to the main parameters of l, therefore it is possible to detect the set of linear shapes L by identifying the points of intersection in the Hough Transform space H of I.

In a discrete implementation, the Hough Transform algorithm uses a two-dimensional array A, called *accumulator*, to represent the plane H. In its first step, the algorithm executes an edge detection algorithm on I. Let I_e be the edge map computed for I, for each pixel $p \in I_e$ the Hough Transform algorithm determines if p corresponds to an edge in I; if so, for every line l_p (in the discrete space defined by A) that may pass through p, the algorithm increases the value of the element in A that corresponds to the main parameters of l_p. Finally, the linear shapes in I are identified by applying a local threshold operator to A to detect its peaks.

B. Barcode detection

The barcode detection task consists in locating the barcodes that appear in a given image; the output of a barcode detection algorithm should consist of a set of bounding boxes surrounding those barcodes.

[1]http://artelab.dista.uninsubria.it/download/

This task has been faced using many different techniques, for example: (i) in [12], [16] scan lines are drawn over the image to detect the exact position of a barcode, (ii) Basaran *et al.* [17] rely on the properties of the Canny Edge Detector [18] to identify edges corresponding to barcodes, (iii) Gallo and Manduchi [9] assume that the regions in the image characterized by weak horizontal gradients and strong vertical gradients correspond to barcodes. In order for the cited models to operate effectively, the barcodes that appear in the processed images need to satisfy a set of constraints, e.g., none of the cited models can detect arbitrary rotated barcodes.

C. Barcode decoding

The barcode decoding task consists in exploiting the information provided by a barcode detection algorithm to read the barcodes that appear in a given image.

As for barcode detection, the decoding task has been faced in literature in many different ways: (i) Gallo and Manduchi [9] exploit deformable templates to efficiently read extremely blurred barcodes, (ii) in [8], [12], [16] the authors adopt different thresholding techniques to decode the bars of 1D barcodes, (iii) Zamberletti *et al.* [10] use a supervised neural network to improve the performance of the Zebra Crossing (ZXing) [19] algorithm, (iv) Muñiz *et al.* [20] decode 1D barcodes by exploiting the accumulator array of the Hough Transform algorithm. Based on the results provided by the authors, the algorithm proposed by Gallo and Manduchi [9] proves to be significantly more robust than the others when applied to camera captured images. The methods presented in [9], [10], [19] are able to read 1D barcodes efficiently on common mobile devices.

III. Proposed Model

A detailed description of the proposed method is given in the following paragraphs. Given an image I, we apply the Canny Edge Detector [18] to I to obtain its edge map I_e. Note that this step can be computed efficiently even on a mobile device, as proved by many available mobile implementations. Once the edge map has been determined, we compute the Hough Transform of I_e in the two-dimensional Hough Transform space H. Finally, we detect the rotation angle θ of the barcodes that appear in I as described in Section III-A and we determine their bounding boxes by analyzing a neural generated accumulator matrix as in Section III-B. In Fig. 2 we show the pipeline of our algorithm for an image of the *Rotated Barcode Database* presented in Section IV-A.

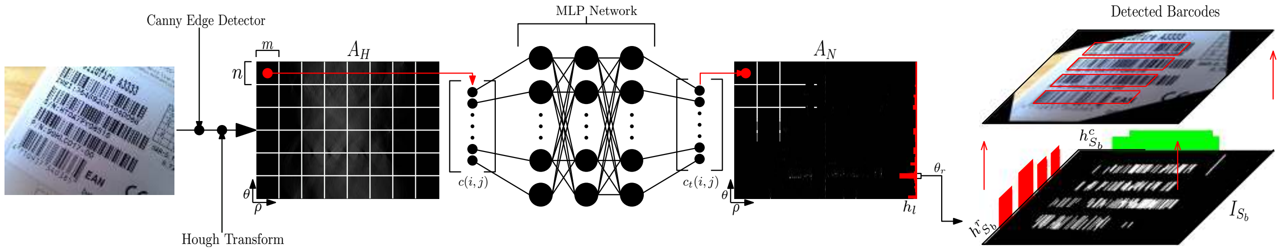

Fig. 2. A visual overview of the steps performed by the proposed model to detect the bounding boxes of the barcodes that appear in a given image.

A. Angle detection

Let A_H be the accumulator matrix for the two-dimensional Hough Space H, a regular grid of cells is superimposed over A_H; the height and the width of each cell are defined as n and m respectively. Every cell c of the grid is given as input to a Multi Layer Perceptron (MLP) [21] neural network that produces a new cell c_t having the same size of c. Let $c(i, j)$ be the value of the element of c in position (i, j), with $0 \leq i \leq n$ and $0 \leq j \leq m$; the value assigned by the MLP network to its corresponding element $c_t(i, j)$ is defined as follows:

$$c_t(i,j) = \begin{cases} 1 & \text{if } c(i,j) \text{ denotes a barcode bar in } I. \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

where an element $c(i, j)$ denotes a barcode bar in I if the line defined by Eq. (1) for $(\rho_{c(i,j)}, \theta_{c(i,j)})$ corresponds to a barcode bar in I.

As stated by Eq. (2), the goal of the neural network is to assign an high intensity value to an element of c_t if it corresponds to an element of c that denotes a barcode bar in the original image I. We train the neural model using a set of training patterns generated from the test set of the given barcode dataset. Let I_t be a training image in the current test set, a training pattern is composed of a pair (in, out) in which: (i) in is the vector representation of a cell extracted from the accumulator in the two-dimensional Hough Transform space H of I_t, (ii) out is the vector representation of a cell in which the elements of in that denote barcode bars in I_t are assigned 1 as intensity value, the others are assigned 0. Once all the cells defined for A_H have been processed by the neural model, we combine them together to generate a new matrix A_N in which the elements having high intensity values represent potential barcode bars.

The main feature of a 1D barcode is that its bars are parallel; for this reason, we define the likelihood l_r of a barcode appearing in I rotated by the angle associated with a row r in A_N as the sum of all the elements of r. This process is repeated for all the rows of A_H to obtain an histogram h_l in which each bin b_r represents the likelihood that the elements of the row r denote the bars of a barcode in I. An example of such histogram is presented in Fig. 2. Let b_r be a bin in h_l and max_{h_l} be the maximum value in h_l, if $b_r = max_{h_l}$ then we assume that some of the elements of r denote the bars of a barcode in I. Let θ_r be the rotation angle specified by r, without further operations we could perform a set of scan lines [12], [16] rotated by θ_r to decode the barcode associated with r. This is an expensive operation since those scan lines should be performed over all the lines of I whose rotation angle is θ_r. However, it is possible to reduce the number of scan lines required to decode the barcode by identifying its bounding box as described in Section III-B.

B. Bounding box detection

Given A_N, we obtain the rotation angle θ_b of every barcode b in I as described in Section III-A. After that, we determine the set S of all the segments in I by applying the same technique of Matasyz *et al.* [22] to A_N. For each barcode b, we define $S_b \subseteq S$ as the set of segments whose rotation angles differ by at most $\pm 5°$ from θ_b and we create a binary image I_{S_b} in which the intensity value assigned to the pixels of the segments of S_b is 1, the others are assigned 0; the image I_{S_b} is then rotated so that the majority of its segments are parallel to the vertical. Similarly to Section III-A, we define two histograms $h^r_{S_b}$ and $h^c_{S_b}$ that describe the intensity profile of the rows and the columns of I_{S_b} respectively, as shown in Fig. 2. More specifically, each bin of those histograms is computed as the sum of the elements of a row/column in I_{S_b}. Finally, we apply a smoothing filter to each histogram to remove low value bins corresponding to isolated non-barcode segments and we determine the bounding box of the barcode b as the intersection area between the rows and the columns associated with the remaining non-zero bins in $h^r_{S_b}$ and $h^c_{S_b}$ respectively. All the operations previously described can be performed in parallel for each detected barcode.

C. Discussion

The computational complexity of the proposed model strictly depends on the size of the accumulator A_H. Note that, due to the aspect ratio of a 1D barcode, it is possible to successfully decode a barcode using a scan line if the rotation angle of the scan line differs by at most $\pm 30°$ from the one of the barcode [9]. This feature enable us to obtain good results even when a single row in A_H is associated with multiple consecutive rotation angles. The neural network is also affected by the parameters n and m; in fact, as proved in Section IV-C, the capability of the MLP network to detect twisted barcodes depends on those two parameters. This is due to the fact that the bars of a twisted barcode (e.g., a barcode printed on an irregular object) are not parallel, therefore some of the points generated in A_H for such bars lie on different

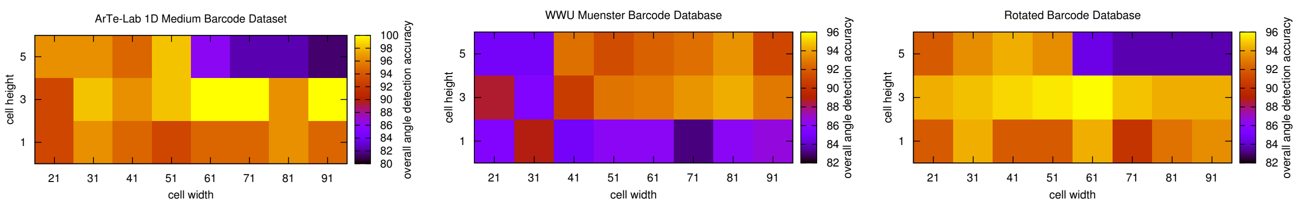

Fig. 3. Overall angle detection accuracy achieved by our model for the three 1D barcode datasets presented in Section IV-A while varying the cell size.

subsequent rows. If we increase n, each cell provided as input to the MLP network covers multiple subsequent rows of A_H and this enables the neural model to successfully detect multiple rows patterns. The computational complexity of the *bounding box detection* phase depends on the size of the original image I; in our experiments we always rescale the input image to a 640x480 pixels resolution without losing overall detection accuracy.

IV. Experiments

A. Datasets

In this section we present the datasets used to measure the performance of the proposed model. We employ two standard 1D barcode datasets: *ArTe-Lab 1D Medium Barcode Dataset* [10] and *WWU Muenster Barcode Database* [12]. We also build an additional publicly available dataset, called *Rotated Barcode Database*, specifically designed to evaluate the performances of angle invariant 1D barcode detection algorithms. Since our method involves a supervised machine learning algorithm, we split each dataset into a training and a test sets. In details, for each dataset, we randomly select 66% of its images as training set and the remaining 33% as test set. In order to evaluate the accuracy of the bounding box detection phase, described in Section III-C, we define the figure-ground segmentation masks for all the images of the previously cited datasets.

ArTe-Lab 1D Medium Barcode Dataset. It consists of 215 1D barcode images acquired using a Nokia 5800 mobile phone. This dataset is not specifically designed to evaluate the performances of angle invariant algorithms; as such, the barcodes that appear in the images are rotated by at most $\pm 30°$ from the vertical. Each image contains at most one non-blurred EAN barcode. In our experiments, we do not employ the extended version of the dataset because the proposed method is not specifically designed to deal with unfocused images.

WWU Muenster Barcode Database. It consists of 1055 1D barcode images acquired using a Nokia N95 mobile phone. As for the *ArTe-Lab 1D Medium Barcode Dataset*, this dataset has not been specifically designed for angle invariant detection algorithms, for this reason most of the barcodes that appear in the images are not rotated from the vertical. Each image contains at most one non-blurred EAN or UPC-A barcode.

Rotated Barcode Database. It consists of 368 1D barcode images acquired using multiple smartphones; all the images are scaled to a 640x480 pixels resolution. This dataset is specifically designed to evaluate the performances of angle invariant barcode detection algorithms; as such, the barcodes shown in the images are rotated by arbitrary angles. Each image may contain multiple EAN and UPC barcodes. Moreover, the barcodes may appear twisted, illegible due to reflections or partially occluded. The dataset is publicly available for download and use.

B. Evaluation metrics

We measure the performances of the two main phases of the proposed model using the *overall angle detection accuracy* for the angle detection phase of Section III-A and the *overall bounding box detection accuracy* for the bounding box detection phase of Section III-C.

Overall angle detection accuracy. Given a dataset D, the overall angle detection accuracy achieved by the proposed model for D is computed as follows:

$$OA_D^{\theta} = \frac{tp}{tp + fn + fp} \tag{3}$$

where tp is the number of barcode rotation angles successfully detected in the test set of D, $tp + fn$ is the total number of 1D barcodes that appear in the images of the test set of D and fp is the number of objects wrongly identified by the MLP network as barcodes. The rotation angle detected for a barcode b is considered correct if it differs by at most $\pm 10°$ from the true rotation angle θ_b.

Overall bounding box detection accuracy. Given a dataset D, the overall bounding box detection accuracy OA^{bb} is calculated by redefining tp in Eq. (3) as the number of barcode bounding boxes correctly detected. Let bb_b be the bounding box for a barcode b, a detected bounding box d_b is considered correct for b if $\frac{bb_b \cap d_b}{bb_b \cup d_b} \geq 0.5$.

C. Results

In this section we discuss the results obtained by the model presented in Section III for the three datasets of Section IV-A. In all the experiments we adopt an MLP network composed by a single hidden layer whose size is equal to $n \cdot m$. We extract 150 background and 50 foreground training patterns from each image of the given test set and we exploit them to train the neural model using the resilient backpropagation algorithm. We define the accumulator A_H as a matrix with 180 rows and $\sqrt{2} \cdot max(h, w)$ columns, where h and w are the height and the width of I respectively.

TABLE I
OVERALL BOUNDING BOX DETECTION ACCURACY.

Dataset	OA^{bb}
ArTe-Lab 1D Dataset [10]	0.86
Muenster BarcodeDB [12]	0.83
Rotated Barcode Database	0.84

TABLE II
OVERALL BARCODE READING ACCURACY WHILE VARYING THE DETECTION METHOD.

Dataset	Barcode Reading Algorithm	
	ZXing [19]	Our ∪ ZXing [19]
ArTe-Lab 1D Dataset [10]	0.82	0.85
Muenster BarcodeDB [12]	0.73	0.81
Rotated Barcode Database	0.61	0.82

In our first experiment we analyze the angle detection phase described in Section III-A; the results we obtain are shown in Fig. 3. It is possible to observe that, as stated in Section III-C, the parameters n and m affect the overall angle detection accuracy. The best value for m is 3; lower values do not allow the MLP network to detect twisted barcodes while higher values introduce too much noise in the patterns processed by the MLP network. Overall, we achieve excellent angle detection performances: if we set $n = 3$ and $m = 61$, we obtain a 100% OA^{θ} for the simple *ArTe-Lab 1D Medium Barcode Dataset* and an average of 95.5% OA^{θ} for the other two datasets. In this configuration, the time required to process an image is roughly 200 ms on a mobile device. Next, we evaluate the overall bounding box detection accuracy obtained by the bounding box detection phase for the same three datasets; the results are presented in Table I. Unfortunately, we cannot provide any comparison with other barcode detection algorithms as they do not usually detect region of interests within the processed images; in our experience, the only method that performs a similar detection process is the one in [9], however we cannot test it because its source code is not currently available. The bounding box detection accuracies we obtain are close to 85% OA^{bb} for all the datasets analyzed, this is a good result considering the fact that our method does not impose constraints and requires no user interaction. The completion time of the bounding box detection phase is 70 ms per image. We perform a final experiment to prove that it is possible to improve the performance of an existing barcode reading algorithm by replacing its detection algorithm with our method. We chose the ZXing [19] algorithm because its source code is available for download; we evaluate the perfomances of both the original algorithm and our modified version using the same metric of Zamberletti *et al.* [10]. From the results presented in Table II, we observe that the algorithm obtained by combining ZXing with our detection method achieves better overall perfomance than the original algorithm for all the three datasets, especially for our *Rotated Barcode Database*.

V. CONCLUSION

We have presented a method for detecting one-dimensional barcodes in camera captured images that is angle invariant and requires no user interaction. We proved the effectivness of the proposed model using three EAN/UPC datasets. The obtained results show that our method is able to precisely detect both the rotation angles and the bounding boxes of one-dimensional barcodes even when such barcodes are partially occluded, twisted or illegible due to reflections. The time required by our approach to process an entire image is roughly 270 ms on a modern mobile device; this is an excellent results because, as shown in our experiments, it is possible to obtain a robust one-dimensional barcode reading algorithm simply by combining our approach with a fast scan line decoding algorithm that processes only the detected bounding boxes.

REFERENCES

[1] X. Shen, Z. Lin, J. Brandt, and Y. Wu, “Mobile product image search by automatic query object extraction,” in *Proc. ECCV’12*, 2012, pp. 114–127.
[2] T. Pavlidis, J. Swartz, and Y. Wang, “Fundamentals of bar code information theory,” *Computer*, vol. 23, no. 4, pp. 74–86, 1990.
[3] Y. Kato, D. Deguchi, T. Takahashi, I. Ide, and H. Murase, “Low resolution QR-code recognition by applying super-resolution using the property of QR-codes,” in *Proc. ICDAR’11*, 2011, pp. 992–996.
[4] H. A-Lin, F. Yuan, and G. Ying, “QR code image detection using run-length coding,” in *Proc. ICCSNT’11*, 2011, pp. 2130–2134.
[5] I. Szentandrási, A. Herout, and M. Dubská, “Fast detection and recognition of QR codes in high-resolution images,” in *Proc. SCCG’12*, 2012, pp. 129–136.
[6] K. Wang, Y. Zhou, and H. Wang, “1D bar code reading on camera phones,” *IJIG*, vol. 7, no. 3, pp. 529–550, 2007.
[7] C. Zhang, J. Wang, S. Han, M. Yi, and Z. Zhang, “Automatic real-time barcode localization in complex scenes,” in *Proc. ICIP’06*, 2006, pp. 497–500.
[8] D. Chai and F. Hock, “Locating and decoding EAN-13 barcodes from images captured by digital cameras,” in *Proc. ICICS’05*, 2005, pp. 1595–1599.
[9] O. Gallo and R. Manduchi, “Reading 1-D barcodes with mobile phones using deformable templates,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1834–1843, 2011.
[10] A. Zamberletti, I. Gallo, M. Carullo, and E. Binaghi, “Neural image restoration for decoding 1-D barcodes using common camera phones,” in *Proc. VISAPP’10*, 2010, pp. 5–11.
[11] L. Chen, “A directed graphical model for linear barcode scanning from blurred images,” in *Proc. ACCV’12*, 2012, pp. 524–535.
[12] S. Wachenfeld, S. Terlunen, and X. Jiang, “Robust 1-D barcode recognition on camera phones and mobile product information display,” *Mobile Multimedia Processing*, vol. 5960, pp. 53–69, 2010.
[13] CipherLab. [Online]. Available: http://www.cipherlab.com.au/
[14] V. Lempitsky, A. Vedaldi, and A. Zisserman, “A pylon model for semantic segmentation,” in *Proc. NIPS’11*, 2011, pp. 1485–1493.
[15] R. O. Duda and P. E. Hart, “Use of the hough transformation to detect lines and curves in pictures,” *Comm. of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.
[16] R. Adelman, M. Langheinric, and C. Floerkemeier, “A toolkit for barcode recognition and resolving on camera phones,” in *Proc. MEIS’06*, 2006.
[17] E. Basaran, O. Uluçay, and S. Ertürk, “Reading barcodes using digital cameras,” in *Proc. IMS’06*, 2006, pp. 835–843.
[18] J. Canny, “A computational approach to edge detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, 1986.
[19] Zebra Crossing. [Online]. Available: http://code.google.com/p/zxing/
[20] R. Muñiz, L. Junco, and A. Otero, “A robust software barcode reader using the Hough Transform,” in *Proc. IISA’99*, 1999, pp. 313–319.
[21] C. M. Bishop, *Neural Networks for Pattern Recognition*, 1995.
[22] J. Matasyz, C. Galambosy, and J. Kittler, “Progressive probabilistic Hough Transform for line detection,” in *Proc. CVPR’99*, 1999.