Original Article
International Journal of Fuzzy Logic and Intelligent Systems
Vol. 14, No. 1, March 2014, pp. 8-16
http://dx.doi.org/10.5391/IJFIS.2014.14.1.8

ISSN(Print) 1598-2645
ISSN(Online) 2093-744X

Robust Video-Based Barcode Recognition via Online Sequential Filtering

Minyoung Kim
Department of Electronics & IT Media Engineering, Seoul National University of Science & Technology, Seoul, Korea

ljfis

Abstract

We consider the visual barcode recognition problem in a noisy video data setup. Unlike most existing single-frame recognizers that require considerable user effort to acquire clean, motionless and blur-free barcode signals, we eliminate such extra human efforts by proposing a robust video-based barcode recognition algorithm. We deal with a sequence of noisy blurred barcode image frames by posing it as an online filtering problem. In the proposed dynamic recognition model, at each frame we infer the blur level of the frame as well as the digit class label. In contrast to a frame-by-frame based approach with heuristic majority voting scheme, the class labels and frame-wise noise levels are propagated along the frame sequences in our model, and hence we exploit all cues from noisy frames that are potentially useful for predicting the barcode label in a probabilistically reasonable sense. We also suggest a visual barcode tracking approach that efficiently localizes barcode areas in video frames. The effectiveness of the proposed approaches is demonstrated empirically on both synthetic and real data setup.

Keywords: Hidden Markov models, Online sequential filtering, Barcode recognition

1. Introduction

In recent years, the barcodes have become ubiquitous in many diverse areas such as logistics, post services, warehouses, libraries, and stores. The success of barcodes is mainly attributed to easily machine-read representation of data that aautomates and accelerates the tedious task of identifying/managing a large amount of products/items.

In barcodes, data are typically represented as images of certain unique and repeating one-dimensional (1D) or two-dimensional (2D) patterns. The task of retrieving data from these signal patterns is referred to as *barcode recognition*, and designing a fast, accurate, and robust recognizers is of significant importance. Although traditional optical laser barcode readers are highly accurate and reliable, these devices are often very expensive and not very mobile.

Because of the availability of inexpensive camera devices (e.g., in mobiles or smartphones), image-based barcode recognition has become considerably attractive. Recently considerable research work has been conducted on this subject. Here we briefly list some recent image-based barcode localization/recognition methods. In [1] block-wise angle distributions were considered for localizing barcode areas. A region-based analysis was employed in [2] whereas discrete cosine transformation was exploited in [3]. Image processing techniques including morphology, filtering, and template matching were serially applied in [4]. The computational

Received: Sep. 28, 2013
Revised : Mar. 10, 2014
Accepted: Mar. 11, 2014

Correspondence to: Minyoung Kim
(mikim@seoultech.ac.kr)

efficiency was recently considered by focusing on both turning points and inclination angles [5].

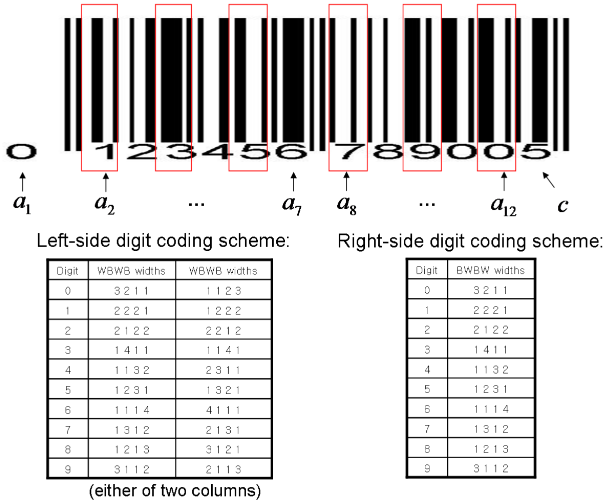
However, most existing image-based barcode recognition systems assume a *single* barcode image as input, and require it to be relatively high quality with little blur or noise for accurate recognition. The drawbacks, also shared by optical readers, are clearly the extra human effort needed to capture a good-quality still picture and consequently the delayed reading time.

In this paper, we consider a completely different yet realistic setup: the input to the recognizer is a *video* of a barcode (i.e., a sequence of image frames) where the frames are assumed to be considerably blurred and noisy. Hence with this setup, we eliminate the extra effort needed to acquire motionless, clean and blur-free image shots. Our main motivation is that although it is ambiguous and difficult to recognize barcodes from each single, possibly corrupted frame, we can leverage the information or cues extracted from entire frames to make a considerably accurate decision about the barcode labels.

Perhaps the most straightforward approach for recognizing barcodes in video frames is a key-frame selector, that basically selects the so-called key frames with least blur/noise, and then applying off-the-shelf single-image recognizer in series. The potential conflict of differently predicted codes across the key frames can be addressed by majority voting. However, such approaches can raise several technical issues: i) how would you judge rigorously which frames should be preferrably selected and ii) which voting schemes are the most appropriate with theoretical underpinnings? Moreover, the approach may fail to exploit many important frames that contain certain discriminative hints about the codes.

Instead, we tackle the problem in a more principled manner. We consider a temporal model for a sequence of barcode image frames, similar to the popular hidden Markov model (HMM). The HMMs were previously applied to visual pattern recognition problems [6–10]. In our model, we introduce hidden state variables that encode the blur/noise levels of the corresponding frames, and impose smooth dynamics over those hidden states. The observation process, contingent on the barcode label class, is modeled as a Gaussian density centered at the image/signal pattern of the representing class with the perturbation parameter (i.e., variance) determined from the hidden state noise level.

Within this model, we perform (online) sequential filtering to infer the most probable barcode label. As this inference procedure essentially propagates the belief on the class label as well as the frame noise levels along the video frames, we can exploit all cues from noisy frames that are useful for predicting the barcode label in a probabilistically reasonable sense. Thus the issues and drawbacks of existing approaches can be addressed in a principled manner.



Left-side digit coding scheme:

Digit	WBWB widths	WBWB widths
0	3 2 1 1	1 1 2 3
1	2 2 2 1	1 2 2 2
2	2 1 2 2	2 2 1 2
3	1 4 1 1	1 1 4 1
4	1 1 3 2	2 3 1 1
5	1 2 3 1	1 3 2 1
6	1 1 1 4	4 1 1 1
7	1 3 1 2	2 1 3 1
8	1 2 1 3	3 1 2 1
9	3 1 1 2	2 1 1 3

(either of two columns)

Right-side digit coding scheme:

Digit	BWBW widths
0	3 2 1 1
1	2 2 2 1
2	2 1 2 2
3	1 4 1 1
4	1 1 3 2
5	1 2 3 1
6	1 1 1 4
7	1 3 1 2
8	1 2 1 3
9	3 1 1 2

Figure 1. (Top) Example EAN-13 barcode image that encodes 12-digit barcodes ($a_{1..12}$) with the checksum digit c. (Bottom) Encoding scheme for each digit.

The paper is organized as follows. We briefly review barcode specifications (focusing on 1D EAN13 format) and describe conventional procedures for barcode recognition in Section 2. In our video-based data setup, the task of barcode localization (detection) must be performed for a series of frames, and we pose it as a tracking problem solved by a new adaptive visual tracking method in Section 3. Then in Section 4, we describe the proposed online sequence filtering approach for barcode recognition, The experimental results follow in Section 5.

2. One-Dimensional Barcodes

Although 2D barcode technologies are now emerging, in this paper we deal with 1D barcodes for simplicity. As it is straightforward to extend it to 2D signals, we have left it for future work. In 1D barcodes, information is encoded as a set of digits where each digit is represented as parallel black and white lines of different widths. Although there are several different types of coding schemes, we focus on the widely-used EAN-13 format. The detailed specifications of EAN-13 barcodes are described here.

An example EAN-13 barcode image is shown in the top panel of Figure 1, where the black and white stripes encode 12-digit barcodes ($a_{1..12}$) with the checksum digit c as also shown at the bottom. In fact, the barcode stripes encodes 12 digits of $a_{2..12}$ and c while the first code a_1 can be easily retrieved from the checksum equation, namely $c = S_1 - S_0$ where $S_0 = a_1 + a_3 + \cdots + a_{11} + 3 * (a_2 + a_4 + \cdots + a_{12})$ and S_1 is the nearest 10's multiple not less than S_0. The barcode stripe image is composed of: 1) the left-most `BWB` (boundary indicator), 2) 6 `WBWB` patterns (one for each of the left 6 digits), 3) the middle `WBWBW` (left/right separator), 4) 6 `BWBW` patterns (one for each of the right 6 digits), and 5) the right-most `BWB` (right boundary).

How each digit is encoded is determined by the widths of the alternating 4 B/W bar stripes (red boxes in the figure). The specific digit encoding scheme is summarized in the bottom panel, for instance, the digit 7 in the right part is encoded by `BWBW` with widths proportional to $(1, 3, 1, 2)$. Note that each digit in the left part can be encoded by either of two different width patterns (e.g., digit 1 can be depicted as `WBWB` with either $(2, 2, 2, 1)$ or $(1, 2, 2, 2)$ widths).

Hence the barcode recognition mainly estimates the bar widths accurately from the image/signal. Of course, in an image-based setup, one first needs to localize the barcode region, a task often referred to as *barcode detection*. Barcode recognition commonly uses the tightly cropped barcode image from a barcode detector, then to decode barcode signals. In our video-based framework, however, the barcode detection has to be performed for each and every frame in real time. This can be posed as an *object tracking* problem. So we propose a fast adaptive visual tracking method in the next section.

3. Barcode Tracking

In this section we deal with the visual barcode tracking problem. Although one can invoke a barcode detector for each frame from scratch, one crucial benefit of tracking is computational speed-up that is critical for real-time applications. A similar idea has arisen in the field of computer vision, known as the object tracking problem. Numerous object tracking approaches have been proposed, and we briefly list a few important ones here. The main issue is modeling the target representation: a view-based low-dimensional appearance model [11], contour models [12], 3D models [13], mixture models [14], and kernel representations [15, 16], are the most popular among others.

Motivated by these computer vision approaches, we suggest a fairly reasonable and efficient barcode tracking method here. Tracking can be seen as an online estimation problem, where given a sequence of image frames up to current time t, denoted by $F_0, \ldots, F_t$, we estimate the location and the pose of the target object (the barcode in this case) in the frame F_t. One way to formalize the problem is the online Bayesian formulation [12], where we estimate $P(u_t|F_t)$ at each time $t = 1, 2, \ldots$. Here u_t is the tracking state specifying the two end points (with respect to the coordinate system of the current frame F_t) whose line segment tightly covers the target barcode area. Typically, we define $u_t = [a_x, a_y, b_x, b_y]^\top$ where the first (last) two indicates starting (end) position. It is straightforward to read the barcode signal $\mathbf{z}_t$ from u_t and F_t by a simple algebraic transformation $\mathbf{z}_t = \omega(u_t, F_t)$.

We consider a temporal probabilistic model for tracking states and target appearances. We setup a Gaussian-noise Markov chain over the tracking states, and let each state be related to an appearance model that measures the goodness of track (how much $\mathbf{z}_t = \omega(u_t, F_t)$ looks like a barcode). Formally, we set the state dynamics as $P(u_t|u_{t-1}) = \mathcal{N}(u_t; u_{t-1}, V_0)$ with some small covariance V_0 for white-noise modeling. The essential part is the appearance model, and we consider a generic energy model,

$$P(F_t|u_t) \propto \exp(-E(\mathbf{z}_t; \theta)/\sigma_0^2), \text{ where } \mathbf{z}_t = \omega(u_t, F_t), \quad (1)$$

where $E(\mathbf{z}_t; \theta)$ is the energy function that assigns a lower (higher) value when $\mathbf{z}_t$ looks more (less) like the target model θ. The online Bayesian tracking can then be written as the following recursion.

$$P(u_t|F_{0\ldots t}) \propto \\ P(F_t|u_t) \cdot \int P(u_t|u_{t-1}) \cdot P(u_{t-1}|F_{0\ldots t-1})\, du_{t-1}, \quad (2)$$

with initial track $P(u_0|F_0)$ can be set as a delta function determined from any conventional barcode detector. The recursion (2) can be typically solved by the sampling-based method (e.g., particle filtering [12]).

Since the target barcode appearance can change over time (mainly due to changes in illumination and camera angles/distances or little hand shaking), it is a good strategy to adaptively change the target model. Among various possible adaptive models, we simply use the previous track $\mathbf{z}_{t-1}$ for the purpose of computational efficiency. Specifically, we define the energy model as $-E(\mathbf{z}_t; \theta) = ||\mathbf{z}_t - \mathbf{z}_{t-1}||^2$. The number of samples (particles) trades off the tracking accuracy against the tracking time, and we fix it as 100 which performs favorably fast with accurate results.

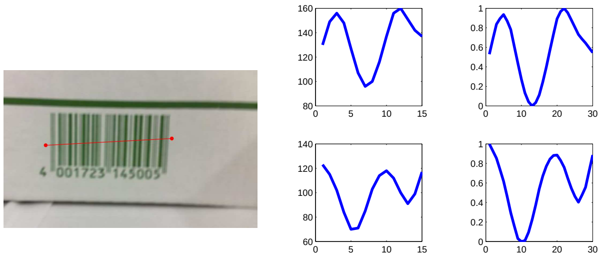


Figure 2. (Left) Endpoints (red) of the line that tightly covers the tracked barcode. (Middle) Unnormalized intensity values for digit 1 and 7 in the left part. (Right) Normalized signals to 1-30 duration and scaled to a range of 0-1.

4. Barcode Recognition

Once tracking is performed for the current frame, we have the tracked end points that tightly cover the barcode (an example is shown in the left panel of Figure 2). We read the gray-scale intensity values along the line segment, which are regarded as the barcode signal to analyze. Based on the prior knowledge of the relative lengths of the left/right-most boundary stripes as well as the middle separator, we can roughly segment the tracked barcode signal into 12 sub-segments, each of which corresponds to each digit's 4 B/W stripes pattern. This can be done simply by equal division of the line segment.

As a result, we get 12 digit patterns, one for each digit. We show, for instance, the intensity patterns of the two digits 1 and 7 on the left part in the middle panel of Figure2. Then for each digit, we normalize its intensity pattern with respect to: 1) a fixed value x-axis length (say, 30) and 2) intensity normalization such that scales up/down the intensity values to range from 0 to 1. The former can be done by interpolation/extrapolation, while we use a simple linear scaling for the latter. The right panel of Figure 2 depicts the normalized signals for the example digits. In this way, the digit signals have the same length and scale across different digits, and hence they can be contrasted meaningfully with each other. We denote by $\mathbf{x}_t^k$ the digit pattern signal of the k-th ($k = 1, \ldots, 12$) digit at frame t.

From the tracking and the feature extraction procedure, we obtain a sequence of digit pattern signals, $\mathbf{X} = \mathbf{x}_1\mathbf{x}_2 \cdots \mathbf{x}_T$, one for each digit, for T video frames. Here we drop the superscript index k for notational simplicity. We then classify the sequence as a digit class label $y \in \{0, 1, 2, \ldots, 9\}$. This sequence classification task is performed individually and independently for each of 12 digits in the barcode.

Instead of heuristic approach such as key-point selection, we tackle the problem in a more principled manner. We consider a temporal model augmented with the class variable y, which essentially builds class-wise HMM. The graphical model representation of our model is depicted in Figure 3.

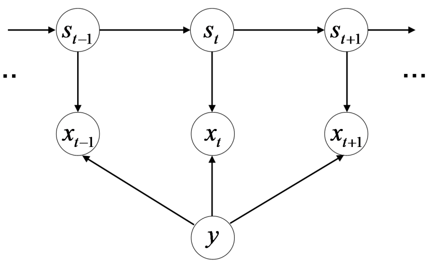


Figure 3. Model of barcode signal sequence recognition.

		s_t		
		1	2	3
s_{t-1}	1	0.7	0.2	0.1
	2	0.2	0.6	0.2
	3	0.1	0.2	0.7

Figure 4. Transition probabilities for hidden state variables.

In our model, we introduce the hidden state variables s_t (associated with each $\mathbf{x}_t$), where s_t encodes the blur/noise level of the observation signal $\mathbf{x}_t$. Specifically, we consider three different noise levels, i.e., $s_t \in \{1, 2, 3\}$, indicating that $s_t = 1$ for relatively clear signal, $s_t = 3$ for severe noise, and $s_t = 2$ in between. To reflect the motion/scene smoothness property in real videos, we also impose a smooth dynamics over the adjacent hidden state variables. In particular, we use a simple table shown in Figure 4 for the transition probabilities $P(s_t|s_{t-1})$. We basically placed higher (lower) values to diagonal (off-diagonal) entries to enforce smooth noise level transition.

The crucial part is the observation modeling, where we employ a Gaussian density model whose mean and variance is determined by the digit class y and the current noise level s_t. More specifically, we set the mean of the Gaussian as the signal code (shown in the bottom tables of Figure 1) corresponding to the digit class y, where the WBWB width codes are duration/intensity-normalized similar to that of the observation signal $\mathbf{x}_t$. The perturbation parameter (i.e., variance) of the isotropic Gaussian is determined from the hidden state noise level s_t. In our experiments, we fix the variances as $\{\sigma_1^2 = 0.5, \sigma_2^2 = 1, \sigma_3^2 = 4\}$ corresponding to $s_t = 1, 2, 3$, respectively. Note that for the digits in the left part, since there are two possible width codes, we model them as mixtures of two Gaussians with equal proportions and the same variance. In summary, the observation model for $\mathbf{x}_t$ can be written as follows:

$$P(\mathbf{x}_t|s_t, y) = \tag{3}$$
$$\begin{cases} \mathcal{N}(\mathbf{m}_y^R, \sigma_{s_t}^2 \mathbf{I}) & \text{(right part)} \\ \frac{1}{2}\mathcal{N}(\mathbf{m}_y^{L_1}, \sigma_{s_t}^2 \mathbf{I}) + \frac{1}{2}\mathcal{N}(\mathbf{m}_y^{L_2}, \sigma_{s_t}^2 \mathbf{I}) & \text{(left part)} \end{cases},$$

where $\mathbf{m}_y^R$ is the normalized signal code vector for the digit y in the right table, and $\mathbf{m}_y^{L_d}$ for the digit y in the column d of the left table ($d = 1, 2$). Also, $\mathbf{I}$ is the identity matrix. The class prior $P(y)$ which we set simply as uniform distribution (i.e., $\frac{1}{10}$).

Within this model, we perform online sequential filtering to infer the most probable barcode label y for a given sequence $\mathbf{X} = \mathbf{x}_1 \cdots \mathbf{x}_T$. That is, the most probable digit label y up to time t can be obtained from:

$$y^* = \arg\max_y P(y|\mathbf{x}_1 \cdots \mathbf{x}_t), \tag{4}$$

where the posterior of y can be computed recursively:

$$P(y|\mathbf{x}_1 \cdots \mathbf{x}_t) \propto P(y|\mathbf{x}_1 \cdots \mathbf{x}_{t-1}) \cdot \tag{5}$$
$$\sum_{s_{t-1}, s_t} P(\mathbf{x}_t|s_t, y) P(s_t|s_{t-1}) P(s_{t-1}|\mathbf{x}_1 \cdots \mathbf{x}_{t-1}, y).$$

The last quantity, namely $P(s_t|\mathbf{x}_1 \cdots \mathbf{x}_t, y)$ is the well-known forward inference for the HMM corresponding to the class y (detailed derivations cab be found in [17]). In our model, it is also possible to infer the noise level at time t, i.e., $P(s_t|\mathbf{x}_1 \cdots \mathbf{x}_t)$, from a similar recursion:

$$P(s_t|\mathbf{x}_1 \cdots \mathbf{x}_t) = \sum_y P(s_t|\mathbf{x}_1 \cdots \mathbf{x}_t, y) P(y|\mathbf{x}_1 \cdots \mathbf{x}_t). \tag{6}$$

In essence, the inference procedures in our model consider the propagating beliefs on the class label as well as the frame-wise noise levels of the video frame sequence. In turn, we are able to exploit all the cues from noisy signal frames potentially useful for predicting the barcode label in a probabilistically reasonable sense. Thus the drawbacks of existing heuristic approaches such as key-frame selection can be addressed in a principled way.

5. Evaluation

In this section we empirically demonstrate the effectiveness of the proposed video-based barcode recognition approach.

5.1 Synthetically Blurred Video Data

To illustrate the benefits of the proposed method, we consider a simple synthetic experiment that simulates the noise/blur that can be observed in real video acquisition. First, a clean unblurred barcode image is prepared as shown in the top-left corner of Figure 5. We then apply Gaussian filters (i.e., $G(x, y) = \exp(-\frac{x^2+y^2}{2\sigma^2})$ with different scales σ. The resulting noisy barcode video frames (of length 8) are shown in the top panel of Figure 5. One can visually check that the frame at $t = 1$ is corrupted the most severely, while the frame at $t = 5$ has the lowest noise level, and so on.

As described in the previous section, we split each barcode frame into 12 areas, one for each digit, normalize regions, and obtain sequences of observation signal vectors $\mathbf{x}_1 \cdots \mathbf{x}_T$. We show the observation vectors for three digits (the first, the third in the left part, and the sixth in the right) at the top rows in three bottom panels of Figure 5.

The online class filtering (i.e., $P(y|\mathbf{x}_1 \cdots \mathbf{x}_t)$) as well as state (noise level) filtering (i.e., $P(s_t|\mathbf{x}_1 \cdots \mathbf{x}_t)$) is performed for each frame t. The results are depicted in second and third rows in each panel. For the 3rd/left digit case, the class prediction at the early stages was highly uncertain and incorrect (i.e., $y = 3$ instead of true 1), however, the model corrects itself to the right class when it observes the cleaner frame ($t = 5$). Although further uncertainty was introduced after that, the final class prediction remains correct.

In addition, in the middle panel (for the 6th/right digit case), as the model observes more frames, the belief to the correct class, that is, $P(y = 7|\mathbf{x}_1 \cdots \mathbf{x}_t)$, increased, indicating that our online filtering approach can leverage and accumulate potentially useful cues from noisy data effectively. The state filtering information is also useful. For instance, looking at the changes of $P(s_t|\mathbf{x}_1 \cdots \mathbf{x}_t)$ along t in the 1st/left digit case, the noise level prediction visually appears to be accurate.

5.2 Real Barcode Videos

We next apply our online temporal filtering approach to the real barcode video recognition task. We collect 10 barcode videos by taking pictures of real product labels using a smartphone camera. The video frames are of relatively low quality with considerable illumination variations and pose changes, most of which are defocused and blurred. Some of the sample frames are shown in Figure 6. The videos are around 20 frames in length on average.

After running the visual barcode tracker using a conventional

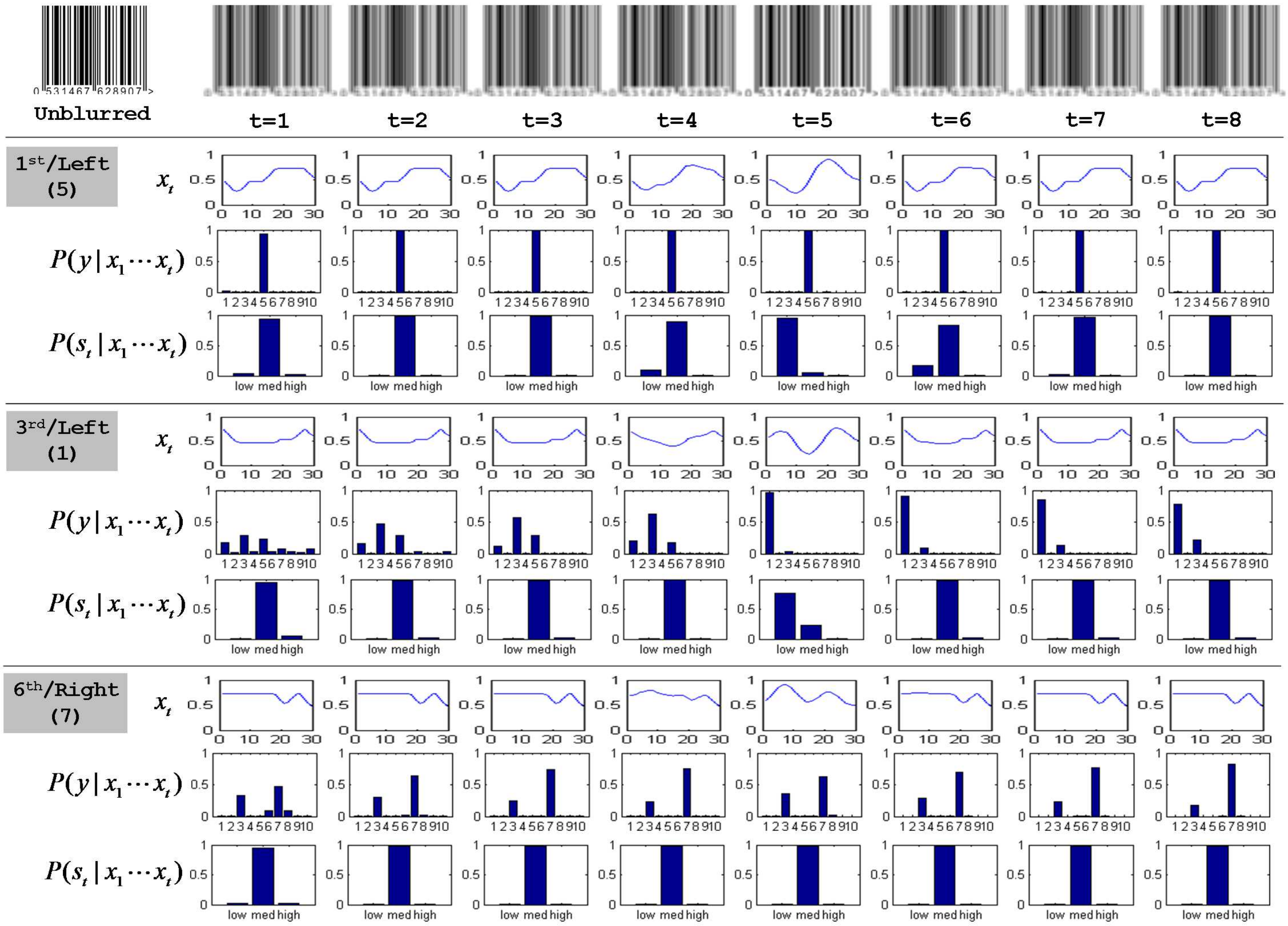


Figure 5. (Top) Unblurred clean barcode image with the sequence of noisy video frames generated from it using Gaussian blur with different scales. (Bottom) Each of three panels depicts the sequences of normalized signal vectors, online class filtering, and state filtering results for three digits (the first, the third in the left part, and the sixth in the right) whose true digit classes are shown in the parentheses.

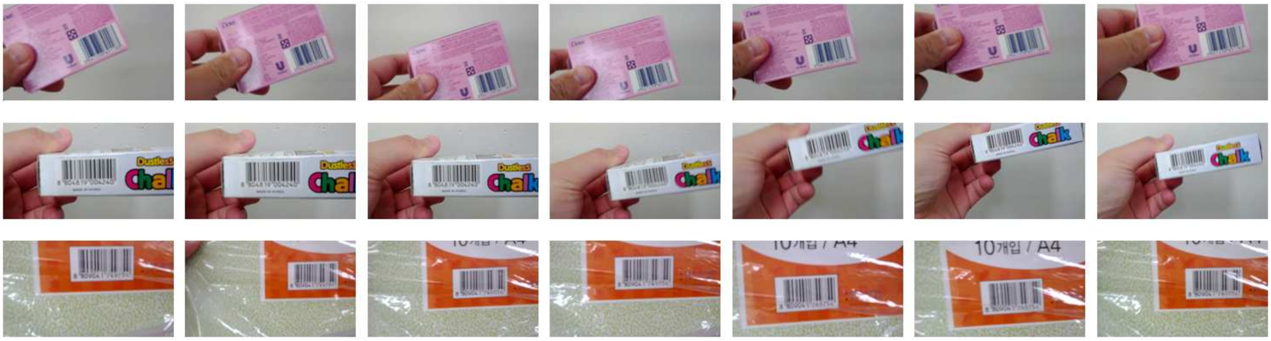


Figure 6. Sample frames from barcode videos.

detector, we obtain the cropped barcode areas. We simply split the area into 12 signals for digits according to the relative boundary/digit width proportions. Then the proposed online filtering is applied to each digit signal sequence individually. For a performance comparison, we also test with a baseline method, a fairly standard frame-by-frame based barcode recognizer, where the overall decoding for a whole video can be done by majority voting.

In Table 1 we summarize the recognition accuracies of the proposed approach and the baseline method over all videos. As there are 12 digits for each video, we report the proportions of the accurately predicted digits out of 12. The proposed ap-

Table 1. Barcode recognition accuracy on real videos

Video#	Frame-by-frame (%)	Proposed method (%)
1	25.00	75.00
2	33.33	75.00
3	0.00	58.33
4	25.00	100.00
5	91.67	100.00
6	33.33	58.33
7	25.00	83.33
8	41.67	66.67
9	58.33	75.00
10	33.33	91.67
Average	36.67	78.33

proach significantly improves the baseline method with nearly 80% accuracy on average, which signifies the impact of the principled information accumulation via sequential filtering. On the other hand, the heuristic majority voting scheme fails to decode the correct digits most of the time, mainly due to its independent treatment of the video frames with potentially different noise/blur levels.

5.2.1 More Severe Pose/Illumination Changes

We next test the proposed barcode tracking/recognition approaches on the video frames with more severe illumination and pose variations. We have collected additional seven videos (some sample video frames illustrated in Figure 7), where there are significant shadows, local light reflections, in conjunction with deformed objects (barcode on plastic or aluminum foil bag containers) and out-of-plane rotations.[1]

The recognition results are summarized in Table 2. The results indicate that the proposed sequential filtering approach is viable even for the severe appearance conditions. In particular, the partial lighting variations (e.g., shadows) can be effectively handled by the intensity normalization in the measurement processing, while the pose changes (to some degree) can also be effectively handled by the noisy emission modeling in HMM.

Table 2. Barcode recognition accuracy on real videos with severe illumination and pose variations.

Video#	Frame-by-frame (%)	Proposed method (%)
11	33.33	83.33
12	41.67	66.67
13	8.33	50.00
14	16.67	91.67
15	25.00	66.67
16	33.33	58.33
17	33.33	75.00
Average	27.38	70.24

6. Conclusion

In this paper we proposed a novel video-based barcode recognition algorithm. Unlike single-frame recognizers the proposed method eliminates the extra human effort needed to acquire clean, blur-free image frames by directly dealing with a sequence of noisy blurred barcode image frames as an online filtering problem. Compared to a frame-by-frame based approach with heuristic majority voting scheme, the belief propagation of the class label and frame-wise noise levels in our model can exploit all cues from noisy frames. The proposed approach was empirically shown to be effective for accurate prediction of barcode labels, achieving significant improvement over conventional single-frame based approaches. Although in practice, the current algorithm needs frames of higher quality to achieve 100% accuracy, the results show that the proposed approach can significantly improve the recognition accuracy for blurred and unfocused video data.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

Acknowledgments

This study was supported by Seoul National University of Science & Technology.

[1] It is worth noting that the in-plane rotations were already dealt with in the previous experiments. However, the out-of-plane rotations, if significant changes, may not be properly handled by the proposed approach since we use a simple equal division to extract each digit code. So, we collect the videos with mild changes in out-of-plane pose changes.

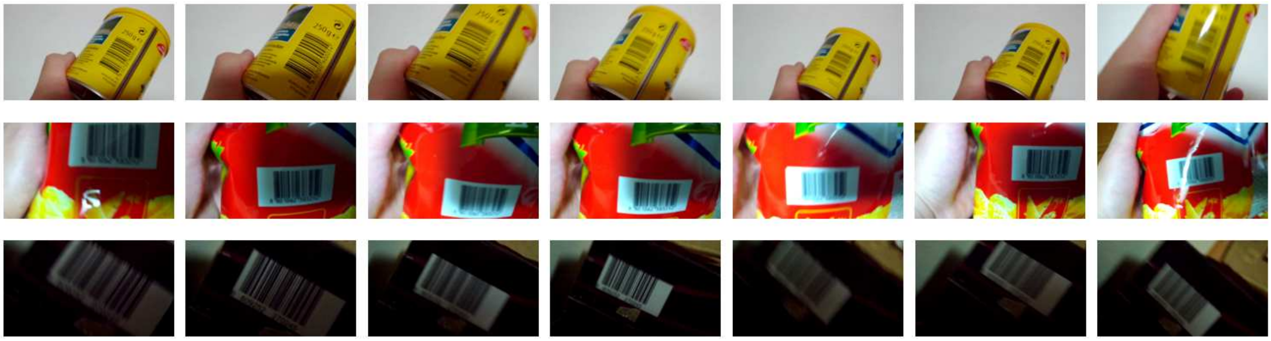


Figure 7. Sample frames from additional barcode videos with severe illumination and pose changes.

References

[1] 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*, Bangkok, Thailand, December 6-9, 2005, pp. 1595-1599. http://dx.doi.org/10.1109/ICICS.2005.1689328

[2] C. Zhang, J. Wang, S. Han, M. Yi, and Z. Zhang, “Automatic real-time barcode localization in complex scenes,” in *IEEE International Conference on Image Processing*, Atlanta, GA, October 8-11, 2006, pp. 497-500. http://dx.doi.org/10.1109/ICIP.2006.312435

[3] A. Tropf and D. Chai, “Locating 1-D bar codes in DCT-domain,” ,” in *Proceedings of the IEEE International Conference on Acoustics*, Speech and Signal Processing, Toulouse, France, May 14-19, 2006, pp. II-741-II-744. http://dx.doi.org/10.1109/ICASSP.2006.1660449

[4] J. Yao, Y. F. Fan, and S. G. Pan, “Segmentation of bar code image with complex background based on template matching,” *Journal of Hehai University Changzhou*, vol. 17, no. 4, pp. 24-27, Apr. 2003.

[5] X. S. Wu, L. Z. Qiao, and J. Deng, “A new method for bar code localization and recognition,” in *2nd International Congress on Image and Signal Processing*, Tianjin, China, October 17-19, 2009, pp. 1-6. http://dx.doi.org/10.1109/CISP.2009.5304402

[6] A. R. Kim and S. Y. Rhee, “Recognition of natural hand gesture by using HMM,” *Journal of Korean Institute of Intelligent Systems*, vol. 22, no. 5, pp. 639-645, Oct. 2012. http://dx.doi.org/10.5391/JKIIS.2012.22.5.639

[7] K. E. Ko, S. M. Park, J. Y. Kim, and K. B. Sim, “HMM-based intent recognition system using 3D image reconstruction data,” *Journal of Korean Institute of Intelligent Systems*, vol. 22, no. 2, pp. 135-140, Apr. 2012. http://dx.doi.org/10.5391/JKIIS.2012.22.2.135

[8] K. A. Lee, D. J. Lee, J. H. Park, and M. G. Chun, “Face recognition using wavelet coefficients and hidden Markov model,” *Journal of Korean Institute of Intelligent Systems*, vol. 13, no. 6, pp. 673-678, Dec. 2003.

[9] I. C. Kim, “Recognition of 3D hand gestures using partially tuned composite hidden Markov models,” *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 4, no. 2, pp. 236-240, Sep. 2004.

[10] M. G. Song, T. T. Pham, S. H. Min, J. Y. Kim, S. Y. Na, and S. T. Hwang, “Robust feature extraction based on image-based approach for visual speech recognition,” *Journal of Korean Institute of Intelligent Systems*, vol. 20, no. 3, pp. 348-355, Jun. 2010.

[11] M. Black and A. Jepson, “EigenTracking: robust matching and tracking of articulated objects using a view-based representation,” *International Journal of Computer Vision*, vol. 26, no. 1, pp. 63-84, Jan. 1998. http://dx.doi.org/10.1023/A:1007939232436

[12] M. Isard and A. Blake, “Contour tracking by stochastic propagation of conditional density,” in *Computer Vision ECCV ’96, Lecture Notes in Computer Science Vol. 1064*, B. Buxton and R. Cipolla, Eds. Heidelberg, Berlin: Springer, 1996, pp. 343-356. http://dx.doi.org/10.1007/BFb0015549

[13] M. La Cascia, S. Sclaroff, and V. Athitsos, “Fast, reliable head tracking under varying illumination: an approach based on registration of texture-mapped 3D mod-

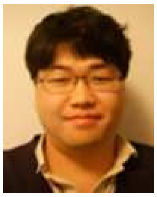
els," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 4, pp. 322-336, Apr. 2000. http://dx.doi.org/10.1109/34.845375

[14] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1296-1311, Oct. 2003. http://dx.doi.org/10.1109/TPAMI.2003.1233903

[15] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564-577, May 2003. http://dx.doi.org/10.1109/TPAMI.2003.1195991

[16] G. D. Hager, M. Dewan, and C. V. Stewart, "Multiple kernel tracking with SSD," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, June 27-July 2, 2004, pp. I-790-I-797. http://dx.doi.org/10.1109/CVPR.2004.1315112

[17] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, Feb. 1989. http://dx.doi.org/10.1109/5.18626



Minyoung Kim received his BS and MS degrees both in Computer Science and Engineering from Seoul National University, South Korea. He earned a PhD degree in Computer Science from Rutgers University in 2008. From 2009 to 2010 he was a postdoctoral researcher at the Robotics Institute of Carnegie Mellon University. He is currently an Assistant Professor in the Department of Electronics and IT Media Engineering at Seoul National University of Science and Technology in Korea. His primary research interest is machine learning and computer vision. His research focus includes graphical models, motion estimation/tracking, discriminative models/learning, kernel methods, and dimensionality reduction.