License Plate Character Segmentation Using Hidden Markov Chains

Vojtěch Franc and Václav Hlaváč

Center for Machine Perception, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University, {xfrancv, hlavac}@cmp.felk.cvut.cz http://cmp.felk.cvut.cz

Abstract. We propose a method for segmentation of a line of characters in a noisy low resolution image of a car license plate. The Hidden Markov Chains are used to model a stochastic relation between an input image and a corresponding character segmentation. The segmentation problem is expressed as the maximum a posteriori estimation from a set of admissible segmentations. The proposed method exploits a specific prior knowledge available for the application at hand. Namely, the number of characters is known and its is also known that the characters can be segmented to sectors with equal but unknown width. The efficient algorithm for estimation based on dynamic programming is derived. The proposed method was successfully tested on data from a real life license plate recognition system.

1 Introduction

The segmentation of a line of characters is an important problem emerging in the license plate recognition (LPR) systems. The objective is to partition image into segments with isolated characters which serve as an input of the Optical Character Recognition (OCR) system. The problem is challenging due to noise in the image, low resolution, space marks, illumination changes, shadows and other artifacts present in real images (see Figure 1). Despite a large effort dedicated to the LPR systems a robust method for character segmentation remains to be an open problem.

Segmentation techniques of machine printed characters have been studied for a long time [2]. Common approaches to character segmentation specially designed for the LPR systems are based on the projection method [3], the Hough transform [7] and intensity thresholding with connected component analysis [5].

We approach the segmentation problem differently by using machine learning methods. In particular, we apply the Hidden Markov Chain (HMC) statistical model and the maximum a posteriori (MAP) estimation. The statistical model is learned from a training set of examples endowed with a ground truth segmentation provided by a user. The aim is to derive a method which mimics the user's segmentation and which exploits all the prior knowledge specific for

W. Kropatsch, R. Sablatnig, and A. Hanbury (Eds.): DAGM 2005, LNCS 3663, pp. 385–392, 2005. © Springer-Verlag Berlin Heidelberg 2005

the application at hand. Namely, we know the number of characters in the license plate in the Czech Republic and also that the characters can be segmented into sectors with the same but unknown width. We started from a segmentation method proposed in [4] which can be applied to find the maximum a posteriori set of beginnings of a specified number of characters. We extended this method in such sense that it exploits the knowledge about the equal width of the characters. We will demonstrate the positive effect of the exploited prior knowledge in the experiment performed on real life data. The proposed method is able to segment characters correctly even in images of a very poor quality. The error rate 3.3% was achieved on the testing set with 1000 examples captured by a real LPR system.



Fig. 1. Example of a car license plate and its ground truth segmentation provided by an user

The paper is organized as follows. The problem is defined in Section 2. Section 3 outlines two related approaches. The proposed segmentation method is described in Section 4. Section 5 gives experimental evaluation on real life data and Section 6 concludes the paper.

2 Task Definition

An input image is a sequence $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathcal{X} \subset \mathbb{R}^{d \times n}$ of n column vectors each of which has d entries. The entries of each column vector are intensity values coded by real numbers from the interval $\langle 0, 255 \rangle$. The input image contains a line of $m \in \mathcal{N}$ characters (alphabets and numerals) which are assumed to be segmentable into sectors of equal width $w \in \mathcal{N}$. The number of characters mis known while their position in the image and their width w is unknown. A character position is an index i of an image column x_i corresponding to the beginning of the character. A segmentation of the input image \boldsymbol{x} is a pair (\mathcal{I}, w) where the set of indices $\mathcal{I} = \{i_1, \ldots, i_m\} \subset \{1, \ldots, n\}$ contains beginnings of mcharacters and w is their width. It is further assumed that $0 < i_1, i_q + w < i_{q+1},$ $\forall q \in \{1, \ldots, m-1\}$ and $i_m + w < n$, i.e., characters cannot overlap each other. The user provides a training set $\mathcal{T} = \{(\boldsymbol{x}^1, \mathcal{I}^1, w^1), \ldots, (\boldsymbol{x}^l, \mathcal{I}^l, w^l)\}$ of examples which contains triplets composed of an image \boldsymbol{x}^j and a segmentation (\mathcal{I}^j, w^j) . The aim is to find a strategy which for a given image \boldsymbol{x} returns the segmentation (\mathcal{I}, w) as would be produced by the user.

2.1 Statistical Model

We decided to use the Hidden Markov Chain (HMC) model to describe a relation between the image \boldsymbol{x} and its corresponding segmentation (\mathcal{I}, w) . To this end, we express the segmentation (\mathcal{I}, w) as a sequence of labels $\boldsymbol{y} = (y_0, \ldots, y_n) \in \mathcal{Y} = \mathcal{L}^{n+1}$. We assume that the label y_i can attain value from a set $\mathcal{L} = \{0, 1, \ldots, w_{max}\}$ where w_{max} is the maximal possible character width. The symbol $\boldsymbol{y}(\mathcal{I}, w)$ denotes a sequence of labels (y_0, y_1, \ldots, y_n) assigned to the segmentation (\mathcal{I}, w) such that:

$$y_{i} = \begin{cases} 1 & \text{if } i \in \mathcal{I}, \\ y_{i-1} + 1 & \text{if } y_{i-1} > 0 \text{ and } y_{i-1} + 1 \le w, \\ 0 & \text{otherwise.} \end{cases}$$
(1)

The first label y_0 which has no corresponding image column is always set to 0. It plays no role in the segmentation but its presence eases the formal analysis. The label $y_i = 0$ means that the column x_i does not belong to any character. The label $y_i > 0$ means that x_i is the y_i -th column of a character. The upper part of Figure 2 shows an example of an input image \boldsymbol{x} with two characters m = 2 of the width w = 4 and the corresponding labelling \boldsymbol{y} .

The formula (1) allows to represent the training set \mathcal{T} by a new training set of sequences $\mathcal{T}_{\mathcal{X}\mathcal{Y}} = \{(\boldsymbol{x}^1, \boldsymbol{y}^1), \dots, (\boldsymbol{x}^l, \boldsymbol{y}^l)\}$ without loss of any information. A pair of sequences $(\boldsymbol{x}, \boldsymbol{y})$ is assumed to be a realization of a random process which is described by a joint probability distribution function

$$p(\boldsymbol{x}, \boldsymbol{y}) = p(y_0) \prod_{i=1}^{n} p(x_i | y_i) p(y_i | y_{i-1}) .$$
(2)

The formula (2) defines a distribution of the HMC model. It is assumed that the HMC model is determined by a discrete distribution $p(y_0)$, a discrete conditional distribution $p(y_i|y_{i-1})$ and a set of $|\mathcal{L}|$ multivariate distributions $p(x_i|y_i), y_i \in \mathcal{L}$. All the distributions do not depend on the position *i* in the sequence. The bottom part of Figure 2 shows the HMC model and its states aligned with the corresponding image columns.

The parameters of the distributions were estimated from the training set $\mathcal{T}_{\mathcal{XY}}$. The discrete distributions $p(y_0)$ and $p(y_i|y_{i-1})$ were estimated using the maximum-likelihood principle (e.g. [4]). The Parzen window density estimator [1] with isotropic Gaussian kernel was used to model the distributions $p(x_i|y_i)$, $y_i \in \mathcal{L}$. The kernel centers were found by the k-means algorithm. The kernel width and the number of kernels were determined using the cross-validation.

2.2 Maximum a Posteriori Estimation of Segmentation

The main aim is to find a most probable segmentation (\mathcal{I}^*, w^*) given an input image \boldsymbol{x} . Each segmentation (\mathcal{I}, w) can be equivalently expressed as a labelling $\boldsymbol{y}(\mathcal{I}, w)$ its construction is described by (1). On the other hand, not every labeling $\boldsymbol{y} \in \mathcal{Y}$ corresponds to a valid segmentation (\mathcal{I}, w) . The symbol $\mathcal{Y}_F = \{\boldsymbol{y}(\mathcal{I}, w): \forall \mathcal{I}, \forall w\} \subset \mathcal{Y}$ denotes a set of all admissible labelings for which corresponding segmentations exist. The HMC model (2) can be used to evaluate a probability $p(\boldsymbol{x}, \boldsymbol{y})$ for given image \boldsymbol{x} and its labeling \boldsymbol{y} . Thus the estimation of the most probable segmentation (\mathcal{I}^*, w^*) can be seen as the search for the most



Fig. 2. Illustrative example of a license plate image aligned with the Hidden Markov Chain model

probable labelling $\boldsymbol{y}(\mathcal{I}^*, w^*)$ which belongs to the set of admissible labelings \mathcal{Y}_F . We define the MAP estimation problem as the following optimization task

$$(\mathcal{I}^*, w^*) = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}_F} p(\boldsymbol{y} | \boldsymbol{x}) = \operatorname*{argmax}_{\mathcal{I}} \max_{w} p(\boldsymbol{y}(\mathcal{I}, w) | \boldsymbol{x}) .$$
(3)

It is seen that the task (3) cannot be solved directly by the Viterbi algorithm [6] which computes the MAP estimation of a sequence of labels but it searches through all possible sequences \mathcal{Y} . We started off by a segmentation method proposed in [4] which allows to find the most probable set of indices \mathcal{I} , however, the character width w is not regarded. The method is outlined in Section 3. We used the idea of the method to derive an algorithm which regards the character width w and solves precisely the task (3). The proposed algorithm will be described in Section 4.

3 Related Methods for Maximum a Posteriori Estimation

In this section we mention two related methods applicable for estimation of the sequence of labels and the segmentation. These methods are the Viterbi algorithm and the method for MAP estimation of segmentation described in [4]. We used both the methods in experimental comparison against the proposed approach (c.f. Section 5).

The first related methods is the Viterbi algorithm [6] which computes the maximum a posteriori (MAP) estimation of the sequence of labels

$$\boldsymbol{y}^* = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}} p(\boldsymbol{x}, \boldsymbol{y}) , \qquad (4)$$

The symbol $x \in \mathcal{X}$ is the observed sequence and $y \in \mathcal{Y}$ is the unknown sequence of labels. Notice that the MAP sequence of labels y^* is selected from the set of all possible sequences \mathcal{Y} . Therefore, it can happen that the optimal sequence

 \boldsymbol{y}^* does not belong to the set of admissible sequences \mathcal{Y}_F . The task (4) can be solved efficiently by the dynamic programming. The computational complexity of the Viterbi algorithm scales with $O(n \cdot |\mathcal{L}|^2)$.

The second related method is the segmentation algorithm described in [4] which solves the following task. Let a set of indices $\mathcal{I} = \{i_1, i_2, \ldots, i_m\} \subset \{0, \ldots, n\}$ define a segmentation. The segmentation \mathcal{I} defines a set of admissible sequences of labels $\mathcal{Y}(\mathcal{I})$. An admissible sequence $\mathbf{y} = (y_0, y_1, \ldots, y_n) \in \mathcal{Y}(\mathcal{I})$ satisfies $y_i = \sigma$, $i \in \mathcal{I}$ and $y_i \neq \sigma$, $i \notin \mathcal{I}$. In our application $\sigma = 1$, i.e., the set \mathcal{I} contains beginnings of characters (indices of first columns). The probability that the true segmentation is \mathcal{I} equals to the probability that the sequence \mathbf{y} belongs to the set $\mathcal{Y}(\mathcal{I})$. The MAP estimation of the segmentation \mathcal{I} is defined as

$$\mathcal{I}^* = \underset{\mathcal{I}}{\operatorname{argmax}} \sum_{\boldsymbol{y} \in \mathcal{Y}(\mathcal{I})} p(\boldsymbol{y}|\boldsymbol{x}) = \underset{\mathcal{I}}{\operatorname{argmax}} \sum_{\boldsymbol{y} \in \mathcal{Y}(\mathcal{I})} p(\boldsymbol{x}, \boldsymbol{y}) .$$
(5)

It is shown in [4] how to transform the task (5) into another one which is efficiently solvable by dynamic programming. The overall complexity of the method scales with $O(n^2 \cdot |\mathcal{L}|^2)$. The proposed extension of this algorithm which considers also the character width is a subject of the Section 4.

4 Proposed Method for Segmentation

Let a pair (\mathcal{I}, w) define a segmentation where $\mathcal{I} = \{i_1, i_2, \ldots, i_m\} \subset \{0, \ldots, n\}$ is the set of indices and $w \in \mathcal{N}$ is character width. The indices \mathcal{I} denote the beginnings of characters thus it must hold that $0 < i_1, i_q + w < i_{q+1}, \forall q \in$ $\{1, \ldots, m-1\}$ and $i_m + w < n$, i.e., all the characters are completely visible and they must not overlap. A segmentation (\mathcal{I}, w) can be expressed using (1) as a sequence of labels

$$\boldsymbol{y}(\mathcal{I},w) = (0, a_1^{i_1-1}, b_{i_1}^{i_1+w-1}, a_{i_1+w}^{i_2-1}, b_{i_2}^{i_2+w-1}, \dots, b_{i_m}^{i_m+w-1}, a_{i_m+w}^n).$$
(6)

The subsequences $a_i^j = (0, 0, \ldots, 0)$ have all the labels equal to 0 and their width j-i+1 varies from 0 (empty sequence) to n-mw, i.e., these subsequence corresponds to the gaps between the characters. The subsequences b_i^{i+w-1} have the identical form equal to $(1, 2, \ldots, w)$, i.e., these subsequences correspond to the characters. We define the MAP estimation of the segmentation (\mathcal{I}, w) as

$$(\mathcal{I}^*, w^*) = \underset{\mathcal{I}}{\operatorname{argmax}} \max_{w} p(\boldsymbol{y}(\mathcal{I}, w) | \boldsymbol{x}) = \underset{\mathcal{I}}{\operatorname{argmax}} \max_{w} p(\boldsymbol{x}, \boldsymbol{y}(\mathcal{I}, w)) .$$
(7)

We will show that the optimization task (7) can be efficiently solved by dynamic programming.

Assuming the HMC model (2) allows to write

$$p(\boldsymbol{x}, \boldsymbol{y}(\mathcal{I}, w)) = \Phi(i_1) \left(\prod_{q=1}^{m-1} \Phi(i_q, i_{q+1}) \right) \Phi(i_m) ,$$

where

$$\begin{split} \varPhi(i_1) &= p(y_0) \ p(x_1^{i_1-1}, y_1^{i_1-1} | y_0) \ p(x_{i_1}^{i_1+w-1}, y_{i_1}^{i_1+w-1} | y_{i_1-1}) \\ &= p(y_0) \prod_{q=1}^{i_1-1} p(x_q | y_q) \ p(y_q | y_{q-1}) \prod_{r=1}^{i_1+w-1} p(x_r | y_r) \ p(y_r | y_{r-1}) \\ \varPhi(i,j) &= p(x_{i+w}^{j-1}, y_{i+w}^{j-1} | y_{i+w-1}) \ p(x_j^{j+w-1}, y_j^{j+w-1} | y_{j-1}) \\ &= \prod_{q=i+1}^{j-1} p(x_q | y_q) \ p(y_q | y_{q-1}) \prod_{r=j}^{j+w-1} p(x_r | y_r) \ p(y_r | y_{r-1}) \\ \varPhi(i_m) &= p(x_{i_m+w}^n, y_{i_m+w}^n) = \prod_{q=i_m}^n p(x_q | y_q) \ p(y_q | y_{q-1}) \end{split}$$

The numbers Φ can be evaluated for a given image \boldsymbol{x} and a sequence of labels $\boldsymbol{y}(\mathcal{I}, w)$ given by (6). If $w \in \{1, \ldots, w_{max}\}$ is fixed then the most probable set of indices $\mathcal{I}^*(w)$ can be expressed as

$$\mathcal{I}^{*}(w) = \underset{i_{1}}{\operatorname{argmax}} \underset{i_{2}}{\operatorname{argmax}} \cdots \underset{i_{m}}{\operatorname{max}} \left(\Phi(i_{1}) \left(\prod_{q=1}^{m-1} \Phi(i_{q}, i_{q+1}) \right) \Phi(i_{m}) \right)$$
$$= \underset{i_{1}}{\operatorname{argmax}} \underset{i_{2}}{\operatorname{argmax}} \cdots \underset{i_{m}}{\operatorname{max}} \left(\log \Phi(i_{1}) + \sum_{q=1}^{m-1} \log \Phi(i_{q}, i_{q+1}) + \log \Phi(i_{m}) \right).$$
(8)

The indices must satisfy $0 < i_1, i_q + w < i_{q+1}, \forall q \in \{1, \ldots, m-1\}$ and $i_m + w < n$. It is easy to see that the task (8) can be already solved by dynamic programming. Finally, the desired MAP segmentation (\mathcal{I}^*, w^*) defined by (7) is computed as

$$(\mathcal{I}^*, w^*) = \operatorname*{argmax}_{w \in \{1, \dots, w_{max}\}} \mathcal{I}^*(w) .$$

The computation of the numbers Φ requires at most $O(n^2)$ operations. Dynamic programming which requires $O(n \cdot m)$ operations has to be performed $w_{max} = |\mathcal{L}| - 1$ times. Therefore the overall computation complexity is $O(n^2 \cdot |\mathcal{L}|)$.

5 Experiments

We tested the proposed method on data captured from a real recognition system for license plates used in the Czech Republic. We compared the following three methods for segmentation:

- **MAP-SEQ.** The segmentation is computed from the MAP estimation of the sequence of labels obtained by the standard Viterbi algorithm (c.f. Section 3). Having the sequence y^* estimated the beginnings \mathcal{I} and widths $\{w_1, \ldots, w_m\}$ are extracted.
- **MAP-I.** The MAP estimation of set of m indices \mathcal{I}^* is computed disregarding the character width (c.f. Section 3). To allow for comparison with other methods the character width was estimated as

$$w^* = \operatorname*{argmax}_{w \in \{1, \dots, w_{max}\}} p(y(\mathcal{I}^*, w), \boldsymbol{x}) ,$$

where \mathcal{I}^* was fixed to the previously compute MAP estimate.

MAP-Iw. The most probable segmentation (\mathcal{I}^*, w^*) computed by the proposed method (c.f. Section 4).

We used l = 1570 examples of images for the estimation of the HMC model. The number of 1000 examples was used for testing. The images were normalized to height d = 14 using the nearest neighbor rescaling. No other preprocessing was used. The number of image columns varied around n = 190. The license plates contained m = 7 characters.

Every tested segmentation method returned for each input image \boldsymbol{x} a pair (\mathcal{I}, w) . Having the segmentation (\mathcal{I}, w) the character centers $\{s_1, \ldots, s_m\}$ can be computed as $s_q = i_q + \frac{w-1}{2}$, $\forall q \in \{1, \ldots, m\}$. The ground truth centers $\{s_1^*, \ldots, s_m^*\}$ can be determined likewise. The image was recognized to be correctly segmented if the estimated centers differed from the grounds truth centers by 20% of the ground truth character width w^* at most, i.e.,

$$\left| \left(i_q + \frac{w-1}{2} \right) - \left(i_q^* + \frac{w^*-1}{2} \right) \right| < 0.2w^*, \quad \forall q \in \{1, \dots, m\},$$

must hold, where (\mathcal{I}^*, w^*) is the ground truth segmentation.

We also measured the number of individual overlooked characters (false negative) and the number of individual false detected (false positives) characters. The total number of characters is m = 7 times higher then the number of images. The correctly segmented character is again such one its estimated center differs from the ground truth center by 20% of the ground truth character width w^* at most. Notice, that the false positives rate equals to the false negatives rate for the MAP-I and MAP-Iw methods because they are guaranteed to return the correct number of segmented characters unlike the MAP-SEQ method.

	Overlooked	False detected	Incorrect
Method	characters	characters	segmentations
MAP-SEQ	7.8%	3.7%	37.0%
MAP-I	0.9%	0.9%	6.1%
MAP-Iw	0.5%	0.5%	3.3%

Table 1. Comparison of the segmentation algorithms

Table 1 summarizes the results obtained. It is clear that the segmentation error decreases as the methods use more and more prior knowledge of the problem at hand. The MAP-SEQ gives the worst results. The MAP-I improves the segmentation considerably as it employs knowledge of the number of characters in the image. The proposed method reduces the error further as it fits best to the problem at hand, i.e., it employs knowledge of the number of characters as well as the fact that the characters are of the same width.

6 Conclusions

We proposed the method for segmentation of a line of characters in an image of a car license plate. The method uses the Hidden Markov Chains (HMC) for modeling the stochastic relation between an input image and its corresponding character segmentation. The segmentation is expressed as the maximum a posteriori estimation problem. The problem leads to the maximum a posteriori (MAP) estimation of a sequence of labels from a set of admissible sequences. An efficient algorithm to solve the problem based on dynamic programming was proposed. The method exploits a specific prior knowledge available for the application which helps to reduce the segmentation error considerably. The proposed method is able to segment characters correctly even in images of very poor quality. The method achieved an error rate 3.3% estimated on data captured by a real LPR system.

Acknowledgments

The authors were supported by the Austrian Ministry of Education under project CONEX GZ 45.535 and by the EC project IST-004176 COSPAL. The first author was support also by MSM 6840770013. The second author was also supported by GACR 102/03/0440.

References

- R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern classification*. John Wiley & Sons, 2nd edition, 2001.
- Y. Lu. Machine printed characters segmentation an overview. *Pattern Recognition*, 28(1):67–80, 1995.
- 3. F. Martin and D. Borges. Automatic car plate recognition using a partial segmentation algorithm. In *IASTED conference on Signal Processing, Pattern Recognition,* and Applications (SPPRA), 2003.
- M.I. Schlesinger and V. Hlaváč. Ten lectures on statistical and structural pattern recognition. Kluwer Academic Publishers, 2002.
- V. Shapiro and G. Gluhchev. Multinational license plate recognition system: Segmentation and classification. In *Proceedings of the 17th International Conference* on *Pattern Recognition*, volume 4, pages 352–355, 2004.
- A.J. Viterbi. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transations Information Theory*, 13:260–269, 1967.
- Y. Zhang and C. Zhang. A new character segmentation algorithm for license plate. In *IEEE Intelligent Vehicles Symposium (IV2003). Columbus, USA*, June 2003.