IDIAP RESEARCH REPORT

Robust Video Text Segmentation and Recognition with Multiple Hypotheses

Jean-Marc Odobez and Datong Chen
IDIAP, Swizerland
{odobez, chen}@idiap.ch

IDIAP–RR 02-18

Apr. 2002

Institut Dalle Molle
d'Intelligence Artificielle
Perceptive • CP 592 •
Martigny • Valais • Suisse

téléphone +41−27−721 77 11
télécopieur +41−27−721 77 12
adr.él. secretariat@idiap.ch
internet http://www.idiap.ch

Rapport de recherche de l'IDIAP 02-18

Robust Video Text Segmentation and Recognition with Multiple Hypotheses

Jean-Marc Odobez and Datong Chen
IDIAP, Swizerland
{odobez, chen}@idiap.ch

Apr. 2002

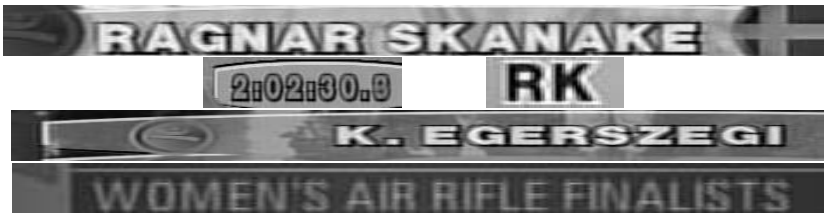

FIG. 1 – *Examples of located text in video*

method for segmenting and recognizing text embedded in video and images is proposed in this paper. In the method, multiple segmentation of the same text region is performed, thus producing multiple hypotheses of binary text images. The segmentation algorithm is stated as a statistical labeling and is based on a markov random field (MRF) model of the label map. Background regions in each hypothesis are then removed by performing a connected component analysis and by enforcing a more stringent constraint (called GCC) on the text characters grayscale values using a robust 1D-Median operator. Each text image hypothesis is then processed by an optical character recognition (OCR) software. The final result is then selected from the set of output strings. Results show that both the use of multiple hypotheses and the GCC significantly improve the results.

1 Introduction

Text recognition in video and images is now recognized as a key component in the development of advanced video and image annotation and retrieval systems. Text characters contained in video are of low resolution, of any grayscale value (not always white) and embedded in complex background even when the whole text string is well located. Thus, applying conventional OCR technology directly leads to poor recognition rate. Therfore, a large amount of work on text segmentation from complex background has been published in recent years. Lienhart [?] and Bunke [?] clustered text pixels from images using a common image segmentation or color clustering algorithm. Although these methods can somehow avoid the text location work, they are very sensitive to noise and character size. Most top down text segmentation methods are performed after text string is located in images. These methods assume that the grayscale distribution is bimodal and that characters correspond a priori to either the white part or the black one, but without providing a way of choosing the right one on-line. Great efforts are thus devoted to perform better binarization, combining global and local thresholding [?], or M-estimation [?], or simple smoothing [?]. However, these methods are unable to filter out background regions with similar grayscale value to characters. Text enhancement methods, if the character grayscale value is known, can help the binarization process [?]. However, without estimation of scales, the designed filters can not enhance character stroke with varying width [?]. In video, multi-frame enhancement [?] also can reduce the number of background regions, but only when text and background have different movements.

In this paper, we present a multi-hypotheses method based on MRF and on grayscale consistency constraint (GCC) to improve both the segmentation and recognition of embedded text with unknown grayscale value. We modelize the gray level distribution in the text images as a mixture of gaussians, and then assign each pixel to one of the gaussian layer. The assignment is based on prior of the contextual information, which is modeled by a MRF with online estimated coefficients. Each layer is considered as a candidate binary text image and is further processed. A GCC based connected component analysis module is first applied on each candidate to remove the background regions which have different shape or greyscale level than the text characters regions. The result is then forwarded to the OCR system as one segmentation hypothesis. By varying the number of gaussians, multiple hypotheses are provided to the OCR and the final result is selected from the set of outputs, leading to an improvement of the system's performances.

The next section presents in more detail our method. Commented results are given in Section 3 and Section 4 concludes the paper.

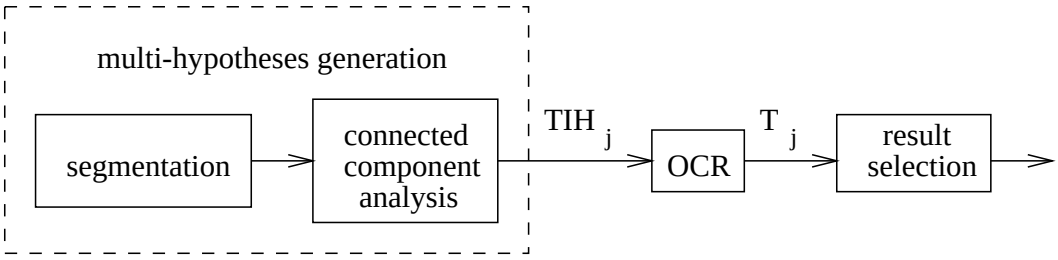

FIG. 2 – *Text recognition scheme*

2 Description of the method

Candidate text regions are first located using the method presented in [?]. In this method, text-like textures are first detected by integrating horizontal and vertical edges, and further segmented into single line text candidates using baseline location. A support vector machine is then used to identify text regions from the candidates.
The text images provided by the text location step are in rectangles as those presented in figure 1. As we mentioned before, OCR software can not be applied directly. Indeed, experience shows that OCR performances are quite unstable, as already mentioned by others [?], and significantly rely on the segmentation quality, in the sense that errors made in the segmentation are directly forwarded to the OCR. In our case, we propose a softer scheme (see figure 2) in which multiple text layer candidates are provided to the OCR, delaying the hard decision, if any, after the OCR step. Our algorithm works as follows: first, text image hypotheses TIH_j are generated, relying on a segmentation step followed by a connected component analysis; then hypotheses are processed by the OCR and the result is selected from the output strings $(T_j)_j$.

2.1 The Gibbsian expectation maximization (GEM) segmentation method

Let S denote the set of sites s (pixels), and o the observation field $o = \{o_s, s \in S\}$, where o_s corresponds to the grayscale value at site s. We model the image intensities in terms of the combination of K simple random processes, also referred to as layers. Each simple process is expected to represent regions of the image having similar gray levels, one of them being text. Thus, the segmentation consists in the mapping of pixels to processes. It is stated as a statistical labeling problem, where the goal is to find the label field $e = \{e_s, 1 \leq e_s \leq K, s \in S\}$ that best accounts for the observations, according to a given criterion. We perform a Maximum a-posteriori (MAP) optimization. Furthermore, to take into account the spatial correlation between assignment of pixels to layers, we model the label field as a markov random field (MRF). Due to the equivalence between MRF and Gibbs distribution $(\mathrm{p}(e) = \frac{1}{Z(V)} e^{-U_1^V(e)})$ [?], the MAP optimization is equivalent to the minimization[1] of an energy function $U(e,o) = U_1^V(e) + U_2^\varphi(e,o)$ where U_1 is equal to:

$$\sum_{s \in S} V_{11}(e_s) + \sum_{<s,t> \in \mathcal{C}_{hv}} V_{12}^{hv}(e_s, e_t) + \sum_{<s,t> \in \mathcal{C}_{diag}} V_{12}^{d}(e_s, e_t) \tag{1}$$

and $\mathcal{C}_{hv}$(resp. $\mathcal{C}_{diag}$) denotes the set of two neighbor pixels in the horizontal/vertical (resp. diagonal) direction. The V are the (local) interaction potentials which expresses the prior on the labels. The energy term U_2 expresses the adequacy between observations and the labels. It is given by:

$$U_2^\varphi(e,o) = \sum_{s \in S} (-\ln \mathrm{p}_{e_s}(o_s)) \qquad (\varphi = (m_i, \sigma_i)_i) \tag{2}$$

where the likelihood of a gray value o at a site s given a particular label i, is modeled by a gaussian, i.e. $\mathrm{p}_i(o_s) = \mathcal{N}(m_i, \sigma_i)$. The use of off-line learned potential V requires to know online the correspondence between learned labels and current ones.[2] Instead, we propose the following algorithm to estimate all

1. with respect to e
2. Remind that text may be black or white or gray.

the parameters Θ=(μ_i,σ_i,V) using an Expectation-Minimization procedure [?]. Recall that the E step involves the computation of:

$$\mathbb{E}\left[\ln \mathrm{p}^{\Theta}_{eo}|o,\Theta^n\right] = \sum_e \ln\left(\mathrm{p}^{\Theta}_{o|e}(e,o)\mathrm{p}_e(e)\right)\ \mathrm{p}^{\Theta^n}_{e|o}(e,o) \tag{3}$$

which is then maximized over Θ. Two problems arise here. Firstly, this expectation on $\mathrm{p}^{\Theta^n}_{e|o}$ can not be computed explicitly nor directly. Instead, this law will be sampled using Monte Carlo methods with a Gibbs sampler, and the expectation will be approximated along the obtained Markov chain. Secondly, the joint log-likelihood probability p^{Θ}_{eo} is not completely known, because of the presence of the uncomputable normalizing constant $Z(V)$ in the expression of $\mathrm{p}(e)$. To avoid this difficulty, we replace $\mathrm{p}(e)$ by its pseudo-likelihood $\mathrm{p_S}(e)$ [?] defined from the conditional probabilities:

$$\mathrm{p_S}^V(e) \doteq \prod_{s\in S} \mathrm{p}(e_s|e_{\mathcal{G}_s}) \tag{4}$$

where $e_{\mathcal{G}s}$ represents the label in neighborhood of s. Using this new criterion, the maximization of the expectation (3) can be executed, providing new estimates of (μ_i,σ_i) and of V. The reader is referred to [?] for more details on the procedure that we have adapted to our need.

2.2 GCC based connected component analysis

After segmentation, for each label, a binary text image hypothesis is generated by assuming that this label corresponds to text and all other labels corresponds to background. In each hypothesis, the text regions are refined by using a simple connected component analysis. We keep only connected components that satisfy the following constraints: size is bigger than 5 pixels; width/height ratio is between 4.5 and 0.1; the width of the connected component is less than 2.1 of the height of the whole text region. The above step is especially useful in eliminating rather large and elongated non-text regions. However, it is unable to eliminate background regions with a gray value slightly different from that of characters but still belongs to the text layer/class. We thus developped another module to enforce a more stringent gray consistency among the connected components.

For a given layer, after the connected component analysis step, we estimate in a robust fashion the gray level mean m^* and standard deviation st^* of the set of pixels S_r belonging to the remaining regions. More precisely, we first compute:

$$m_1 = \underset{m}{\operatorname{argmin}}\, Med_{s\in S_r}|o_s - m| \ ,\ r_1 = Med_{s\in S_r}|o_s - m_1|$$
$$\text{and}\ \ st_1 = 1.48 \times (1 + \frac{5}{|S_r| - 1}) \times r_1$$

where Med denotes a standard median operator. Valide pixels are then identified by checking that their gray value is within a given range of m_1, that is: $\frac{|o_s - m_1|}{st_1} < k$ with k a given ratio (we use a value of 2). The Then the mean m^* and standard deviation st^* are reestimated from the valid pixels only using the usual formula. Finally, a connected component is eliminated from the layer if more than 50% of its pixels have a gray level value different than the majority of pixels, that is, verify: $\frac{|o_s - m^*|}{st^*} > k$. Figure 3 illustrates the result of this step.

2.3 Text recognition

The basic recognition of a text string from a single binary text image is performed using an OCR software. Indeed, in our method, we provide multiple text image hypotheses to the OCR and select the final result from the set of output strings based on a string confidence evaluation.

2.3.1 The initial hypotheses generation:

In the GEM segmentation method, if we have K different labels, we get K hypotheses. The right choice of K is another important and difficult issue. One general way to address the problem consists in checking whether the increase in model complexity really provides a better fit of the data. This can

methods	K	Ext.	CRR	Prec.	WRR
Otsu	2	**9009**	**88.4%**	**93.8%**	**89.3%**
GEM*	2	**9081**	**88.7%**	**93.4%**	**90.8%**
GEM	2	**9149**	**90.4%**	**94.5%**	**93.0%**
GEM*	3	9249	89.9%	92.9%	84.9%
GEM	3	9196	90.1%	93.7%	86.6%
GEM*	4	9128	88.8%	93.0%	85.2%
GEM	4	9172	89.0%	92.9%	85.9%
GEM*	4/3/2	9432	92.5%	93.8%	91.7%
GEM	4/3/2	**9460**	**92.7%**	**93.7%**	**93.1%**

TAB. 1 – *Recognition results in extracted characters (Ext.), character recognition rate (CRR), precision (Prec.) and word recognition rate (WRR). GEM* are the algorithms without using GCC.*

be done for instance by using the minimizing description length criterion. However, this information theoretic approach may not be appropriate for qualifying a good text segmentation. Therefore, we use a more conservative approach, by varying K from 2 to 4, generating in this way nine text image hypotheses TIH_j.

2.3.2 Result selection and confidence value :

Each text image candidate TIH_j is processed by the OCR software[3] thus providing a string T_j. The final string result is the string T_j that provides the largest confidence value CV which is defined as:

$$CV\left(T\right) = \sum_{i=0}^{l_T-1} f\left(T\left[i\right]\right) + \sum_{i=1}^{l_T-1} g\left(T\left[i-1\right], T\left[i\right]\right).$$

where, l_T is the length of string T, $g(x,y)$ is a simple bi-gram language model defined as:

$$g(x,y) = \begin{cases} -4 & if\ x = lower\ case, y = upper\ case \\ 0 & \text{otherwise} \end{cases}$$

$$\text{and } f(x) = \begin{cases} 4 & if\ x = upper\ case \\ 2 & if\ x = lower\ case \\ 0 & if\ x = i,I,l \\ -1 & \text{otherwise} \end{cases}$$

Some characters (i, I, l) and lower case characters are given lower weights because background noise is more often recognized as these characters.

3 Experiments

The whole scheme was tested on text regions located and extracted from one hour of sports video provided by the BBC, using the algorithm presented in [**?**]. It correctly located 98.7% text regions while providing 0.38% false alarms. We randomly selected 1208 images from the extracted text regions, mainly by time sub-sampling, providing a database of 9562 characters or 867 words. Figure 1 shows some examples. Text characters are embedded in complex background with JPEG compression noise, and the grayscale value of characters is not always the highest.
Performances are evaluated using character recognition rates (CRR) and precision rates (Prec) that are computed on a ground truth basis as:

$$CRR = \frac{N_r}{N} \quad and \quad CRR = \frac{N_e}{N}$$

3. we use an open OCR toolkit (OpenRTK) from Expervision

FIG. 3 – *Applying grayscale consistency: a) original image b) text layer (GEM K=3) c) same as b), after the connected component analysis d) text layer (Otsu), e) same as d), after the connected component analysis, f) same as c), after the gray level consistency step.*

N is the true total number of characters, N_r is the number of correctly recognized characters and N_e is the total number of extracted characters. Additionally, we compute the word recognition rate to get an idea of the coherency of character recognition within one solution. For each text image, we count the words from the ground truth of that image that appear in the string result. Results are listed in table 1.
Considering the binarization case (K=2), we can see that the GEM algorithm provides slightly better results than the Otsu algorithm. It is due to the GEM adaptability, which, by learning the local spatial properties of the grayscale distribution, is noise adaptive and is able to better avoid over segmentation. However, the gain is not so significant and can be explained by the fact that the GEM algorithm mainly improves the shape of the characters. This improvement is somehow cancelled out by the OCR which is indeed able to cope with shape noise. However, the use of the GCC improves the results a lot. It can be explained similarly, by the fact that background regions that are not removed greatly affects the OCR performances. The GCC step succeeds well in this task by keeping the character components and removing the background regions.
Similarly, the last rows of table 1 list the results obtained by generating 9 hypotheses (using K=2 to 4). They show that even with our simple confidence criteria, and without the GCC, the improvement is very significant, with a reduction of 35% with respect to the standard Otsu algorithm of the character error rate. Together with the GCC, the reduction in both character and word error rate with respect to the Otsu algorithm is about 36%.
The word recognition can be improved by keeping the best two or best three strings with respect to the confidence level. We obtain a 97.5% word recognition rate using the highest two hypotheses and a 97.9% using three hypotheses in GEM, which significantly improve the result of 93.1% obtained when keeping only the best hypothesis. This can yield better text searching results by offering more precise keywords in image and video indexing and retrieval system. Also, this means that there is room for improvement using a better selection scheme.

4 Conclusion

In this paper, we proposed a method for segmenting and recognizing embedded text of any grayscale value in image and video based on MRF, gray scale constraints and a multiple hypotheses scheme. Experiments have shown that the use of these multiple hypotheses and/or of the gray scale constraint significantly improve the results with respect to Otsu algorithm.