A Language-Independent, Open-Vocabulary System Based on HMMs for Recognition of Ultra Low Resolution Words

Farshideh Einsele
University of Fribourg
Department of Informatics
Bd. de Perolles 90
1700 Fribourg, Switzerland
farshideh.einsele@unifr.ch

Rolf Ingold
University of Fribourg
Department of Informatics
Bd. de Perolles 90
1700 Fribourg, Switzerland
rolf.ingold@unifr.ch

Jean Hennebert
HES-SO
Institut Informatique de gestion
TECHNO-pôle 3
3960 Sierre, Switzerland
jean.hennebert@hevs.ch

ABSTRACT

In this paper, we introduce and evaluate a system capable of recognizing ultra low resolution words extracted from images such as those frequently embedded on web pages. The design of the system has been driven by the following constraints. First, the system has to recognize small font sizes where anti-aliasing and resampling procedures have been applied. Such procedures add noise on the patterns and complicate any a priori segmentation of the characters. Second, the system has to be able to recognize any words in an open vocabulary setting, potentially mixing different languages. Finally, the training procedure must be automatic, i.e. without requesting to extract, segment and label manually a large set of data. These constraints led us to an architecture based on ergodic HMMs where states are associated to the characters. We also introduce several improvements of the performance increasing the order of the emission probability estimators and including minimum and maximum duration constraints on the character models. The proposed system is evaluated on different font sizes and families, showing good robustness for sizes down to 6 points.

Categories and Subject Descriptors

12. [**DE**]: Document Engineering

Keywords

Screen-rendered text recognition, HMMs, web document analysis

1. INTRODUCTION

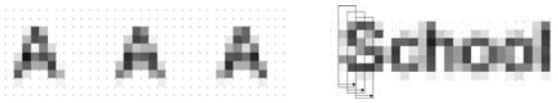
There is no doubt about it: the world wide web has been established as the most ultimative information provider nowadays. Consequently the investigations in the area of text indexation and information retrieval of web pages has been excessively increased during the recent years. Search engine crawlers have made significant progresses in indexing the HTML plain text. A considerable amount of research has been performed in the area of web image indexation. These approaches are either based on image shape analysis and classification [17] or on analyzing the HTML-text associated to these images [8].

More specifically in the image indexation problem, web pages often contain images with embedded text with important semantical value for information retrieval and indexation [2]. One can distinguish between two main categories of text found in web images. The first one corresponds to text visible on scenes shot by cameras. The related area of research which is called "camera based text recognition" tackles this problem [9]. The second category corresponds to bitmap images that are processed using dedicated software such as Photoshop or Fireworks. Such images are generated by web designers to create for example banners, menus, headers, logos etc. The work that we present in this paper focuses on the recognition of text belonging to this second category. However, we believe that the principles of the approach could also be generalized to camera-based text-recognition.

Such text embedded in web images is often anti-aliased with small font sizes($<$ 12 pts) and has *ultra* low resolution (between 72 and 90 dpi). An existing approach is to use classical OCR software. However, OCRs are generally built to treat high-resolution ($>$150 dpi) bi-level images acquired from scanned documents and are therefore not designed to recognize such text. Several works have been proposing various image enhancement algorithms to transform the text image into a quality that is supported by the OCR [10] [1] [14]. While these works have reported noticeable results, the image enhancement is sometimes limited due to very low resolution, to anti-aliasing or to non-homogenous text.

Instead of pre-processing the images to feed a classical OCR system, our approach is to use a recognizer specifically trained to recognize such inputs. Our motivation is indeed to reach better accuracy using a recognizer that is specifically trained on inputs including the specificities of such images. To achieve this, we based our system on Hidden Markov Models(HMMs) that are versatile and powerful statistical tools used in various applications such as in the field of cursive handwriting [11] or automatic speech recognition [15]. In our previous works, we have been first conducting a preliminary study on isolated characters [4]. The focus was on the understanding of the variabilities of text in web images and on identifying a reliable feature extrac-

SAC'08 March 16-20, 2008, Fortaleza, Ceará, Brazil



(a) Character level (b) Word level

Figure 1: (a) Example of anti-aliased, down sampled character 'A' with different grid alignments. (b) Low resolution version of word 'School' and illustration of the sliding window used for the feature extraction.

tion and local character scoring. We have shown that a feature extraction based on moments computation and on multivariate Gaussian density functions leads to robust results. Then, in [5], we extended the approach to recognize full words. Our decision was to use HMMs that present the interesting property to solve the character segmentation and word recognition at the same time. In this approach, one left-right HMM is built for each word where characters are associated to one or more HMM state. A large HMM can be finally built considering the vocabulary taken from a dictionary of 60'000 words, making each word-level HMMs competing against each other. While giving very satisfactory results, this approach has two drawbacks. First, the recognition is limited to the words available in the dictionary. Some inputs were, indeed, not available in the dictionary, such as inflected forms or proper names, and therefore were not recognized. Second, the memory and cpu usage was still pretty high even when performing several optimization of the HMM topology. A porting of this system on low-end devices such as PDAs would have been difficult to realize.

To overcome these drawbacks, we are proposing here to use an ergodic topology for the HMM, where all character models are connected to each other. With such a system, the vocabulary size is potentially unlimited while keeping low the usage of system resources. More specifically, we use an ergodic topology based on minimum and maximum constraints which have been obtained automatically from training process. We first measure the impact of number of Gaussian mixtures on a specific font.Then we evaluate the performance of the system when changing font sizes from 6 to 10 points and different font groups.

The remainder of this paper is organized as follows: In Section 2 we list the specifities of text in web images. In section 3 we describe the system used both for training and test. In section 4 we show the evaluation results and finally we draw conclusions and discuss our future work.

2. SPECIFITIES OF ULTRA LOW RESOLUTION, ANTI-ALIASED TEXT

The type of inputs our system is treating is illustrated on Fig. 1. These inputs present specificities at the character and at the word level.

Character level: (1) the character has an *ultra* low resolution, usually smaller than 100 dpi with small point sizes frequently between 6 and 12 points, (2) the character has artefacts due to anti-aliasing filters and (3) the same characters can have multiple representations due to the position of the sampling grid.

Word level: As can be observed, there are no spacings available to segment characters within the word. Therefore the well-known pre-segmentation methods [13] used in classical OCR systems can not be applied anymore in this case. Furthermore, the anti-aliasing noise on both borders of adjacent characters is also source of variability.

3. SYSTEM DESCRIPTION

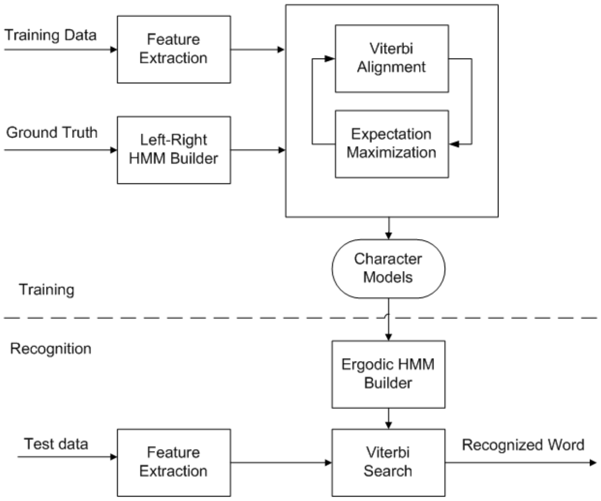


Figure 2: Block diagram of training and recognition

We don't address in this work the problem of text detection. Furthermore, we assume that words can be accurately segmented using classical segmentation algorithms. In other words, we are making the assumption that our system receives as input an image including a single word that then need to be recognized by our system.

Our system is based on HMMs that are statistical models able to compute the likelihood of an observation sequence given a set of states having transitions between them [15]. Each state is usually associated to a given pattern and a so-called emission probability density function (pdf) is used to model features extracted from this pattern. In our approach, HMM states are associated to character images. The transitions between states are weighted with transition probabilities. According to this and to the usual symplifying assumptions done with HMMs, the likelihood of a model can be computed as the sum of the product of emission and transition probabilities along all the possible paths.

As illustrated on Fig. 2, our system has two parts: the training part and the recognition part. The training part aims at computing character models by recomposing word-level HMMs based on simple left-right topology iteratively analysing a large training set of word images. At recognition time, we use an Ergodic HMM topology where each character model are connected to each other. More details about the different blocks composing our system are given below.

3.1 Feature extraction

HMMs model ordered sequences of features that are function of a single independent variable. We decided here to compute a left-right ordered sequence of features by sliding an analysis window on top of the word. Therefore the independent variable is, in our case, the x-axis. As *few pixels* are available for each character, we decided to use the *first and second order central moments*. We have observed in our previous studies [4] [3] that such features are fairly discriminant for the recognition of low-resolution characters embedded in images. As illustrated on Fig. 1(b), we used a 2 pixels length window shifted 1 pixel right. In each analysis window, we compute a feature vector of 8 components including the 6 first and second order central moments, the sum of gray pixel values and an additional feature computed from the difference between the baseline and the y coordinate of the gravity center of each analysis window. This last feature is actually optimistically computed as the baseline is here assumed to be correctly estimated.

3.2 Character Models Training

The training method is performed directly on words for which a simple left-right HMM is recomposed by gluing together the corresponding character sub-models. At training time, a character is modeled with one state where a self-loop transition allows to remain in this model as long as the sliding window is on top of the character. As introduced in [6], we also use an inter-character model '#' to capture the anti-aliasing noise between the adjacent characters. This model is here treated in the same way as another character model. According to our tests this noisy zone spans 1 to 3 pixels dependent of font family, size and shape. Fig. 3 show the topology of an HMM recomposed at training time for the word 'cat'. The emission probability of each state is computed using continuous Gaussian mixtures (see for example [15]). The training of the model parameters is performed using an iterative process as follows:

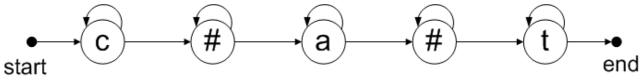


Figure 3: HMM topology for training

1. **Viterbi alignment**. According to the values of the parameters of the character models, we compute for the whole training set the corresponding alignement between states and feature vectors using the Viterbi algorithm.

2. **Parameter re-estimation**. From the segmentations obtained earlier, we compute new values of the parameters of the character models, i.e. mean vectors, covariance matrices, mixture weigths and self-loop probability associated to each states. This re-estimation is also an iterative process that is performed using the classical expectation maximisation (EM) process.

Steps 1 and 2 are iteratively repeated until convergence is reached, typically after some iterations. The initial alignment is obtained performing a linear segmentation, assuming that all characters have equal lengths. This approach has proven to be efficient provided that the quantity of training samples is large enough. Additionally, we have made the assumption that the components of the feature vector are uncorrelated. This presents the advantage to let the covariance matrix be diagonal and to be more computationally efficient. We have measured that this assumption is actually not critical in terms of accuracy [6].

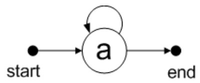


Figure 4: One-state character model

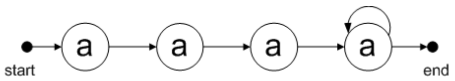


Figure 5: Character model with min duration constraint

3.3 Recognition With Ergodic Topology

At recognition time, we are proposing here to build an HMM with an ergodic topology at the character level. The topology includes all transitions from one character to the other, therefore allowing to recognize potentially any words in an open-vocabulary approach. As alternative to the ergodic approach, a topology could also be proposed where a large set of words is used to build a large HMMs where all words are competing in parallel [5]. While such an approach allows a more precise modelling of a set of words, it has the disadvantage of limiting the recognition capability to a given vocabulary. In other related recognition domains where the vocabulary of the input is potentially very large, ergodic topologies have also been proposed. We can refer to [7] where an ergodic HMM system is presented to recognize handwritten street names, to [16] for automatic language identification and to [12] for speaker verification.

Our ergodic topology is illustrated on Fig. 7. As seen on this Figure, the characters are all accessible in parallel and a transition is looping back to all characters from the inter character model '#'. In this Figure, the states represented by black dots are non-emitting states classicaly used to glue sub-HMMs together. Each character sub-model can take different topology as illustrated on Fig.4-6. The first topology on Fig. 4 is corresponding to the one used at training time. In our previous work [6], we have experienced that the use of minimum duration constraints as expressed in the second topology on Fig. 5 delivers better results as it basically impeach the decoding procedure to leave too early a state giving low local scores. For this work, the minimum duration values were inferred from the bounding boxes of each isolated characters. In this paper, we are introducing an extension of this topology that is illustrated on Fig. 6. The values of each transitions leading to the end of the model are corresponding to the probabilities $p_i(n)$ of observing at least n feature vectors in a given character model i. These values are computed during training by inspecting the Viterbi forced alignment on each word. This new duration model, while introducing similar minimum duration constraints as in model Fig. 5, introduces also a maximum duration constraint, expressing the fact that characters have limited width.

For a given test image, we use the Viterbi criterion to determine the best path in this ergodic topology. This path actually defines the recognized sequence of characters composing the word. The Viterbi decoder is also configured to prune out the less probable paths along the recognition process to keep the memory and cpu usage in reasonable ranges.

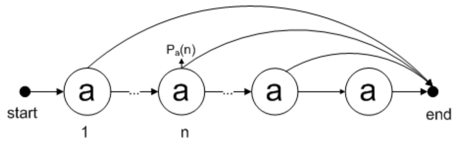


Figure 6: Character model with min-max duration constraint

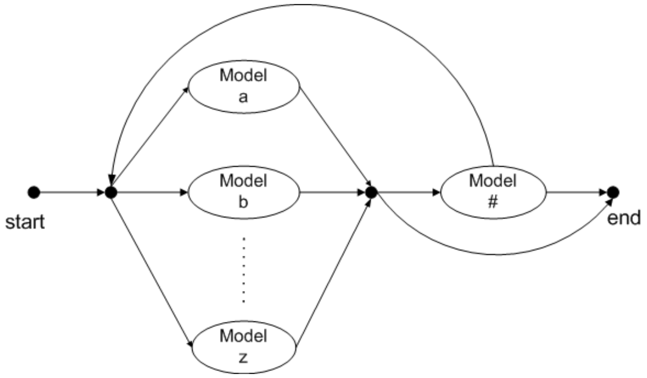


Figure 7: Ergodic model for testing

The transition probabilities going from the submodel '#' back to each character is actually corresponding to the case of equiprobable character sequences, for any pair of characters. More realistic transition values could be computed by estimating character bigram frequencies from a real life dictionary. Such a configuration would present the advantage to give lower scores to less probable character sequences, at the cost of making the system dependent to a given language. We leave this investigation for future work.

4. EXPERIMENTAL RESULTS

For training, a set of nearly 8000 word images is generated by selecting words from a large dictionary. The selection procedure guarantees that each of the 26 characters are represented at least 400 times in the training set. First the images are produced in high resolutions using the `java.awt.Font` class and are then resampled using Photoshop to our target resolution. Anti-aliasing filters are automatically applied by Photoshop. For testing, an independent set of 3000 unseen word images is generated with the same procedure. Our results are all reported in terms of word recognition rates.

We also focus on single font recognition in this work, which means that each font is modeled and tested independently. Using the topology of Fig. 6 for all experiments, we have investigated the following factors:

1. **Model order**. We investigated the impact of using more complex models by increasing the number of Gaussian mixtures used to compute the emission probabilities. Word recognition results were obtained using the Sans Serif Verdana font, 10 points. As illustrated on Fig. 8, increasing the model order allows to reach better performance thanks to a more precise modelling. No significant gain is observed for models over 64 Gaussians.

2. **Font size**. Using the optimum number of Gaussian mixtures obtained from previous experiments and without modifying the system architecture, we computed the recognition performance for different font size going from 10 to 6 points. The objective is here to measure the impact of smaller font sizes on the system performance. Table 1 summarizes our results. As expected, reducing the font size has a negative impact on the recognition performance. Nevertheless, fairly good performance around 90% can still be obtained even for the very small size of 6 points.

3. **Font families**. Still keeping the full system constant, we investigated the stability of our recognizer on two font families Serif and Sans Serif. From Table 1, we can observe that performances are similar between these families when considering larger sizes (8 to 10 points). On the other side, Sans Serif shows better performance for small font sizes (6 and 7 points). A potential explanation of this behaviour can be found in the Serif artefacts that are more numerous and that add more anti-aliasing noise when lower resolutions are considered. Also, the Sans Serif Verdana font we used for our tests has been specifically designed to offer good lisibility at low resolution.

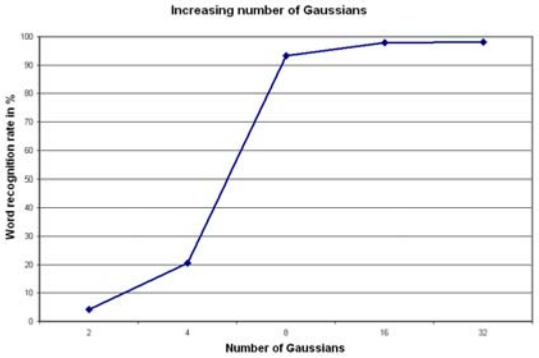


Figure 8: Evolution of word recognition rate increasing the number of Gaussian Mixtures.

Table 1: Recognition rate for different font families and sizes.

	10pts	8pts	7pts	6pts
Sans Serif	96.63%	95.21%	92.91%	92.91%
Serif	96.40%	96.48%	89.01%	88.98%

5. CONCLUSIONS AND FUTURE WORK

We have introduced and evaluated a recognition system capable of recognizing ultra low resolution words extracted from images. The system includes a features extraction module based on sliding windows on which moments are computed. The modelling part is based on HMMs where each state is associated to a specific character. At training time, the character models are trained automatically by recomposing simple left-right word level HMMs. At recognition time, an ergodic topology is built where all character sub-models are allowed to be followed by another character model. Minimum and maximum duration constraints are also introduced for the character models, simply altering the original topology of each model. The proposed system has been evaluated on different font sizes and families. From this evaluation, we can conclude on the following advantages of our approach:

- The HMM is able to solve at the same time the segmentation and the recognition of the characters, then avoiding the use of character segmentation procedures that do not apply well on low resolution anti-aliased character sequences.
- The ergodic architecture allows to recognize any words making the system able to work in an open vocabulary manner, potentially on any language supported by this set of characters.
- The robustness of the training convergence of HMMs allows for a fully automated training where the information on character position is not requested.
- The design of the system based on sliding windows, HMMs and multi-Gaussian models allow to apply the same architecture potentially on any font familly and sizes.

Potential future works could go in the direction of including linguistic constraints in the system architecture. One possibility would be to measure character bigram frequencies from a dictionary and to use these values for the transition probabilities between character sub-models. A generalization of this approach could also be performed computing n-gram character sequences. Another possibility could be to keep the n-best recognition hypothesis as output of the ergodic model and to prune out the hypothesis that are unprobable looking in a dictionary.

6. REFERENCES

[1] A. Antonacopoulos and D. Karatzas. Text extraction from web images based on a split-and-merge segmentation method using color perception. In *Proc. of ICPR*, Cambridge, UK, August 23-26, 2004.

[2] A. Antonacopoulos, D. Karatzas, and J.O. Lopetz. Accessing textual information embedded in internet images. In *Proc. of Electronic Imaging, Internet Imaging II*, San Jose, California, USA, Jan. 2001.

[3] F. Einsele, J. Hennebert, and R. Ingold. Towards identification of very low resolution, anti-aliased characters. In *Proc. of IEEE conference of ISSPA*, Sharjah, UAE, Feb 12 - 17 2007.

[4] F. Einsele and R. Ingold. A study of the variability of very low resolution characters and the feasibility of their discrimination using geometrical features. In *Proc. of World Academy of Science, Engineering and Technology vol. 6*, pages 213–217, Istanbul, Turkey, June 24 - 26 2005.

[5] F. Einsele, R. Ingold, and J. Hennebert. Recognition of low resolution word images using hmms. In *Advances in Soft Computing 45*, pages 429–437, Springer Verlag, 2007.

[6] F. Einsele, R. Ingold, and J. Hennebert. Using hmms to recognize ultra low resolution anti-aliased words. In *LNCS no. 4815*, Springer Verlag, Dec. 2007.

[7] M. A. El-Yacoubi, M. Gilloux, and J. M. Bertille. A statistical approach for phrase location and recognition within a text line: an application to street name recognition. In *IEEE Transactions on PAMI, vol. 24, no. 2*, 2002.

[8] Z. Gong, L. Hou, and C. Wa Cheang. Web image indexing by using associated texts. In *Knowledge and information systems, 2006, vol. 10, no2, ISSN 0219-1377*, pages 243–264, Faculty of Science and Technology, University of Macau, MACAO, 2006.

[9] J. Liang, D. Doermann, and H. Li. Camera-based analysis of text and documents: a survey. In *International Journal on Document Analysis and Recognition, vol. 7*, pages 84–104, 2005.

[10] D. Lopresti and J. Zhou. Locating and recognizing text in www images. In *Information Retrieval, Volume 2, Numbers 2-3*, pages 177–206, Springer Verlag, Heidelberg Germany, May 2000.

[11] U.V. Marti and H. Bunke. Using a statistical language model to improve the performance of a hmm-based cursive handwriting recognition system. In *Int. Journal of Pattern Rec. and Art. intelligence, 15*, pages 65–90, 2001.

[12] Y. Miyazawa, J.-I. Takami, S. Sagayama, and S. Matsunaga. An all-phoneme ergodic hmm for unsupervised speaker verification. In *Proc. of ICASSP*, pages I–249–252, 1994.

[13] G. Nagy. Twenty years of document image analysis in pami. *IEEE Transactions on PAMI*, 22:38–62, Jan 2000.

[14] S.J. Perantonis, B. Gatos, and V. Maragos. A novel web image processing algorithm for text area identification that helps commercial ocr engines to improve their web recognition accuracy. In *Proc. of WDA*, Edinburgh, United Kingdom, August 3, 2003.

[15] L. Rabiner and B. Juang. *Fundamentals Of Speech Recognition.* Prentice Hall, 1993.

[16] V. Ramasubramanian S. A. Santoshkumar. Automatic langauage identification using ergodic hmm. In *Proc. of ICASSP*, pages 455–468, 2005.

[17] S. Santini. Multimodel search in collections of images and text. In *Journal of Electronic Imaging, Volume 11, Issue 4*, pages 455–468, 2002.

7. BIOGRAPHIES

Farshideh Einsele received her diploma of electrical engineering in 1989 from ETHZ Switzerland. She then has worked in several companies and since 2000 has been lecturer in Unversities of Applied Sciences. She is currently approaching the end phase of her PhD in CS in the University of Fribourg, Switzreland.

Jean Hennebert received his diploma of electrical engineering in 1993 from the Polytech Mons, Belgium. In 1998 he received his PhD in CS from EPFL Switzerland. He then worked for 6 years in different companies. He joined as a researcher and lecturer in 2004 the DIVA group of the CS department of the University of Fribourg, Switzerland.

Rolf Ingold received his diploma of mathematical engineering in 1983 and his PhD in Sciences in 1989 both from EPFL Switzerland. Since 1989 he works as a professor in the DIVA group of the CS department of the University of Fribourg, Switzerland.