Modeling On-line Handwriting Using Pairwise Relational Features

Rudy Sicard, Thierry Artières, Eric Petit

Modeling On-line Handwriting Using Pairwise Relational Features

Rudy Sicard

France Télécom – R&D/
TECH/IDEA/TIPS

Rudy.Sicard@francetelecom.com

Thierry Artières

LIP6, Université Paris 6

Thierry.artieres@lip6.fr

Eric Petit

France Télécom – R&D/
TECH/IDEA/TIPS

Eric.Petit@francetelecom.com

Abstract

We propose a new modeling strategy for on-line handwriting. It relies on the use of local and relational features. It allows implementing a variety of models, including traditional Markovian models. Introducing relational features allows building models that exhibit much robustness to noise, extra strokes, temporal ordering variations etc. It may be used for various tasks such as sequence recognition or partial matching of sequences.

1. Introduction

A number of models have been proposed for sequence processing, recognition and segmentation. In order to make learning tractable these models generally rely on a number of simplifying assumptions, this is the case of one of the most popular models for sequence processing, namely Hidden Markov Models (HMM). A number of extensions of HMM have been proposed to take into account dependencies between observations in a sequence. One may cite among others autoregressive HMMs [14] and trajectory models [12]. These systems allow taking into account local temporal dependencies between observations. Relational features have also been used in the image processing field, e.g. for image segmentation. For instance, Markov Random Fields are a popular technology for integrating relationship features among neighbouring pixels in order to smooth pixel labelling [6].

Recently, conditional models have been proposed to overcome some of the major drawbacks of HMM and more generally of generative models, Conditional Random Fields (CRF) is one of these models [10]. The nature of these conditional models avoids making any restrictive assumption about the input data distribution; no simplifying assumption is required. Although these models have been shown to outperform more traditional generative models like HMM in a few information retrieval tasks such as part-of-speech tagging, they are not so well adapted to real-valued signals recognition tasks such as on-line handwriting recognition. Also, conditional models are leaned in a discriminant way that fits well a classification task but may not be adapted for other tasks of interest in sequence processing and in on-line handwriting in particular.

The goal of this study is to develop efficient models of sequential data for various tasks such as sequence recognition and segmentation but also for more general sequence analysis tasks such as partial recognition, rejection, diagnosis... Diagnosis covers a wide range of applications; it aims at giving an accurate evaluation about the quality of a sequence with respect to a model.

We propose in this paper to develop models that include traditional local features as well as relational features. The idea is to take into account relationships between all the observations in the input sequence. Rather than assuming independence between observations, we consider relational features between all pairs of observations and assume these are independent. This leads to a generative model whose distribution on input sequences is rather close to Random Fields. The definition of relational features and of their probability distribution may lead to various models, traditional models such as HMM are special cases of this modeling scheme.

In the context of handwriting relational features may correspond to position and spatial features. Using spatial information has proven to be useful to improve recognition accuracy [11]. It is often roughly used ([3][11]) except in the case of Asian character recognition where some ad-hoc method have been investigated ([9][19]). The application of our modeling framework to handwriting will allow us to show how spatial information may increase the system's robustness to noisy signals, extra strokes or temporal ordering variations.

We present our modeling strategy in §2 and discuss inference and training algorithms in §3 and §4. Then we present how these models may be used for recognition of complete or partial sequences and report experimental results for on-line handwriting data in §5.

2. Relational modeling for sequences

The probability $p(x/y)$ of a sequence of observations $x = (x_1,...,x_T)$ conditionally to a state sequence (i.e. a segmentation) $y = (y_1,...,y_T)$ may be written as:

$$p(x/y) = \prod_{t=1}^{T} p(x_t / x_1^{t-1}, y) \qquad (1)$$

where x_1^{t-1} stands for $(x_1,...,x_{t-1})$. Distributions such as $p(x_t / x_1^{t-1}, y)$ being difficult to estimate, one traditionally introduces independence assumptions to

simplify inference and learning. For instance, in HMM, one assumes conditional independence so that

$$p(x_t / x_1^{t-1}, y_1^T) = p(x_t / y_t)$$

and $p(y_t / x_1^{t-1}, y_1^{t-1}) = p(y_t / y_{t-1})$. Such assumptions lead to efficient algorithms but fail at taking into account complex and long range dependencies. Attempts have been made for proposing richer models by considering specific temporal local dependencies. A family of such models consists of segmental and trajectory models where one state emits globally a sequence of observations rather than emitting a sequence of successive independent observations [12].

We investigate here another alternative which consists in using as much relational features (describing relations between observations) as possible for approximating $p(x_t / x_1^{t-1}, y_1^T)$. We are interested in approximations expressed as a product of potential functions of the following form:

$$p\left(x_t \middle| x_1^{t-1}, y\right) \approx \frac{1}{Z\left(y, x_1^{t-1}\right)} f(x_t, y_t) \prod_{i=1}^{t-1} g(x_t, x_i, y_t, y_i) \qquad (2)$$

where f and g may be any arbitrary potential functions and $Z\left(y, x_1^{t-1}\right)$ is a normalization factor that ensures the above is a probability. Function f encodes local dependencies between an observation x_t and the corresponding state variable y_t, while function g encodes dependencies between pairs of observations and the corresponding states. The form in Eq. 2 is very close to pairwise Markov Random Fields that have been popularized in the image segmentation and recognition processing field [5]. Pairwise modeling appears as an efficient alternative for estimating complex probabilistic distributions over a set of variables. It is an interesting trade-off between expressive power and tractability. In a way, pairwise modeling allows taking into account dependencies between the predicted variable x_t and multiple observed variables $(x_i)_{i=1,...,t-1}$ through the dependencies of x_t with each one of the observed variables. Our model is more complex than pairwise Markov Random Fields which exploit g function of the form $g(y_t, y_{t'})$; hence observations x_t are handled through f functions only. For instance in image segmentation tasks x_t is a local feature describing a pixel (e.g. grey level) and g functions are used to introduce smoothing constraints on labels of neighbouring pixels $(y_t$ and $y_{t'})$.

The form in Eq. 2 is quite general and exhibits more expressive power than traditional models (e.g. HMM). Using Eq. 1 and 2 the probability of a sequence may be rewritten as:

$$p\left(x_1^T \middle| y\right) \approx \frac{1}{Z(y)} \prod_{t=1}^{T} f(x_t, y_t) \prod_{i=1}^{t-1} g(x_t, x_i, y_t, y_i) \qquad (3)$$

The main difficulty in Eq. 3 lies in the normalization factor $Z(y)$ that may lead to complex and even intractable

algorithm for inference. This term may however be computed in particular cases, for instance if all potential functions are Gaussian functions. In this work we consider normalized potential functions so that the model in Eq. 3 may be formalized as a generative model as follows:

$$p(s, r | y) = \prod_{t=1}^{n^k} p\left(s_t / y_t\right) \prod_{i=1}^{t-1} p\left(r_{t,i} / y_t, y_i\right) \qquad (4)$$

where s and r are two sets of features that are derived from x. s is a sequence of local features and s_t stands for local features corresponding to x_t. Besides r is a matrix $r = \{r_{t,i}\}_{1 \le t, i \le T}$ encoding pairwise relationships, $r_{t,i}$ denotes relational features that characterize the relationship between x_t and x_i. To take an example, one can choose to define $s_t = x_t$ and $r_{t,i} = x_t - x_i$. These sets of features may be viewed as a dual representation of an input sequence x. Figure 1 illustrates the model in Eq. 4 as a Dynamic Bayesian Network for an input sequence of size 4.
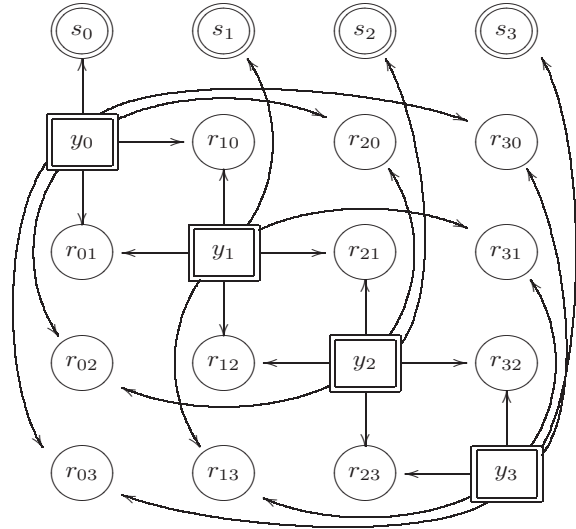

Figure 1. Graphical representation of a Dynamic Pairwise Relational Model. Nodes *y* correspond to labels (states), nodes *s* to local features and nodes *r* to relational features. The model is represented unfolded for an input sequence of length 4.

Various models may be implemented depending on the definition of relational features and on the distribution $p(r_{t,i} / y_t, y_i)$. We describe how to choose features and distributions to build a purely sequential local model equivalent to an HMM, a purely relational model and mixed models. We define this terminology below.

To design an HMM one defines local features as $s=x$ and defines f functions as Gaussian distributions. Relational features are defined and modelled according to:

$$\begin{cases} r_{t,i} = t - i \\ p(r_{t,i} / y_t, y_i) = a_{y_t, y_i} \ if \ r_{t,i} = 1 \end{cases} \quad (5)$$

where $a_{i,j}$ are real values satisfying $\sum_j a_{i,j} = 1$. This model is equivalent to a HMM whose transition probabilities are $a_{i,j}$. It is a *local* model since relational features do not depend on x and it is a *sequential* model since it uses the temporal ordering of observations.

Another interesting model is a purely relational model defined though relational feature $r_{t,i} = x_t - x_i$ whatever t and t', and with a Gaussian distribution $p(r_{t,i} / y_t, y_i)$ computed on these feature vector $r_{t,i}$. The model does not use local features and does not use the temporal ordering of observations. It is a *relational* model that allows the segmentation of an observation sequence to be driven by the relationships between observations rather than by their temporal order. This may be very interesting in some pattern matching problems as we will see in the experimental section.

Of course one can imagine a number of *intermediate* models exploiting both local and relational features with Gaussian f functions on local features $s_t = x_t$ and Gaussian g functions on relational features $r_{t,i} = x_t - x_i$. Many variants may be obtained by defining local and relational features and by using or not the temporal ordering of observations.

3. Inference and segmentation

Segmentation of an input sequence is performed though inference in the Bayesian model expressed in Eq. 4 (cf. Fig. 1). Given an input observation sequence x, segmentation consists in finding the best label sequence y^*, i.e. the one that maximizes $P(y/x)$. It is an inference problem which is NP-hard in our case because of the existence of loops in the model (cf. Fig. 1).

There are a number of algorithms for performing inference in Bayesian networks, such as Belief Revision (BR) and Belief Propagation (BP) to name most popular ones. BR aims at finding the maximum a posteriori solution (MAP) for y^*. It is an exact algorithm for loop-less networks but its behaviour for more complex (i.e. loopy) networks is less appealing [18], e.g. its convergence is not warranted. Concerning BP, although it does not provide explicitly an approximation for y^* and despite it is an exact algorithm for loop-less networks only, BP is known to exhibit still interesting properties for more loopy networks [17]. For this reason, we rather chose to use BP in our work.

Actually, BP aims at calculating marginal distributions for every random variable in the network. It may be used for computing approximation of $p(y|x)$ since the product of the marginal distributions may be shown to be an approximation of $p(y|x)$. Hence, noting

$q_i(y_i)$ the marginal distribution for y_i, $q(y) = \prod_{i=1} q_i(y_i)$ may be shown to be the best approximation of $P(y/x)$ with respect to the Kullback-Leibler divergence criteria $KL[p(y|x)\|q(y)]$ [2]. This algorithm leads to a decoding complexity proportional to the square of the sequence's length times the square of model size (i.e. number of states).

4. Learning

Let first assume that the training set include complete labelling of observation sequence which means the learning set include a set of N observation sequences $X = \{x^1, ..., x^N\}$ and their respective label sequence $Y = \{y^1, ..., y^N\}$ where each label sequence y^k is a complete state sequence with same length (noted n^k) than its corresponding observation sequence x^k. Then:

$$P(\Theta|X,Y) = \frac{P(X|Y,\Theta)P(\Theta|Y)}{P(X|Y)} \quad (6)$$

with:

$$p(X|Y,\Theta) = \prod_{k=1}^{N} \left[p\left(s^k, r^k \middle| y^k, \Theta\right) \right] \quad (7)$$

where (s^k, r^k) is the dual representation of x^k (as discussed in section 2). Besides, using Eq. 3:

$$p(X|Y,\Theta) = \prod_{i=1}^{N} \left[\prod_{t=1}^{n^k} p\left(s_t^k \middle| y_t^k\right) \prod_{j=1}^{t-1} p\left(r_{t,j}^k \middle| y_t^k, y_j^k\right) \right] \quad (8)$$

where s_t^k is the tth term in sequence s^k. In the following we note α_l and $\beta_{l,m}$ the parameters of likelihood functions for local and for relational features. Hence:

$$p\left(s_t \middle| y_t = q_l\right) = p(s_t | \alpha_l) \quad (9)$$

$$p\left(r_{t,j} \middle| y_t = q_l, y_j = q_m\right) = p\left(r_{t,j} \middle| \beta_{l,m}\right) \quad (10)$$

Let further assume that $P(\Theta)$ may be factorized as:

$$p(\Theta) = \prod_{l=1}^{L} p(\alpha_l) \prod_{m=1}^{L} p(\beta_{l,m}) \quad (11)$$

Then any learning method relying on the parameter's posterior probability can be used (e.g. Maximum A Posteriori MAP, Bayes Point Estimation [7], Bayesian Model Averaging [8]). Note that putting all together (Eq. 6 to 11) one may show that the parameters of all potential functions may be learned independently in order to maximize $P(\Theta / X, Y)$.

Actually in the general case the label information of a training observation sequence is limited to the class

label. Then the parameter posterior probability includes a summation over all possible segmentations Y:

$$p(\Theta|X) = \sum_Y p(\Theta|X,Y)p(Y|X) \qquad (12)$$

A few problems arise since first $p(Y|X)$ is unknown, second the summation over Y is intractable and makes model parameters dependent on each others. A solution is to rely on an EM-like algorithm. Following the derivation from Tanner [16] that aims at maximizing the logarithm of the posteriors one may show that parameters can again be estimated independently with any standard method using the parameter posterior computed in the E step. For instance with the MAP criterion parameters β would be chosen as:

$$\hat{\beta}_{l,m} = \arg\max_{\beta_{l,m}}\left[p(\beta_{l,m}) \prod_{i=1}^{N} \prod_{t=1}^{n^k} \prod_{j=1}^{t-1} p\left(r_{t,j}^k / \beta_{l,m}\right)^{q\left(y_t^k=l, y_j^k=m\right)} \right]$$

where $q\left(y_t^k = l, y_j^k = m\right)$ stands for the probability that y_t^k equals l and y_j^k equals m. Note that these terms are computed from marginal distributions that are approximated using the inference algorithm described in section 3.

5. Experiments

As we said earlier, various models may be implemented from our framework, they correspond to different applications. We give a few examples below. All experiments have been performed on the benchmark Unipen database. The samples used in the experiments are digits samples written by 12 writers.

5.1. What does a relational model learn?

Here, to show deeper the interest of modeling relational features, we investigate what has been learned by a relational model for symbol Π. The model has three states and has been learned with three training samples (Fig. 2-bottom). Figure 2-top shows the distribution over relational features. It is a 3x3 matrix whose boxes illustrate the β parameters corresponding to pairs of states (i.e. Gaussian distribution parameters). Hence, the box on the lth row and the mth column illustrates values of parameters $\beta_{l,m} = (\mu_{l,m}, \Sigma_{l,m})$. The mean vector $\mu_{l,m}$ is an average displacement vector between observations in states l and m, it is represented by a straight line starting from the center of the box. Centered at the end of this average displacement vector, an ellipse represents deviation $\Sigma_{l,m}$ around the average displacement. For example in box (1,3) one can see that observations corresponding to the third state are on the right of observations in first state, with high vertical variability and low horizontal variability. Note that self state relation distribution represents the deviation of

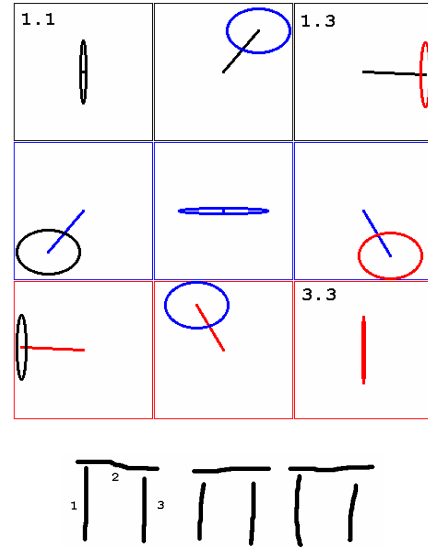observations in the state, hence a null mean displacement vector.

Figure 2. Illustration of distributions over relational features (top) in a three states of a model that has been learned with the three training samples for symbol Π on the right (bottom).

5.2. Handwriting quality and robust segmentation

Evaluating the quality of an input handwriting signal is a difficult problem and may be used for different purposes. First it may be used to design a rejection mechanism in a handwriting recognition engine. One wants to reject parts of an input sequence (e.g. words in a sentence) because of low confidence on the recognition decision or because these parts may correspond to out-of-vocabulary words. Rejection mechanisms are often rough and consist in comparing likelihoods to thresholds. There are situations where more accurate diagnoses are required. For instance it is important to evaluate the quality of handwriting in order to detect potential problems in childhood. Hence, there is today some interest in automating handwriting or hand draw diagnosis tools ([1], [5], [15]). For such tasks, it is necessary to have a smart analysis method for detecting poorly written or drawn part. In order to do so one has to detect parts of letters that are badly written or not written at all, to detect additional strokes etc. Also, one has to identify absolute and relative problems such as when two letters do not have the same height, or when an o is not written clockwise (i.e. in a non standard temporal ordering), or when a dot of an i is far too high or big etc. Such information may be gathered from the segmentation path. Robust segmentation is then required to for some works design automated diagnosis systems able to determine fine and accurate information about the quality of a handwriting signal. Unfortunately, standard models (e.g. HMM) are very sensitive to extra strokes and noisy parts in an input signal. For instance, an extra stroke

usually perturbs the segmentation of remaining strokes of an input signal.

We present here some experiments that show some robustness features of our models. As we show, the use of relational features brings much robustness concerning the segmentation of correct parts of an input signal. In the case of handwriting, the relational model that we described in §2 is interesting in that it allows identifying partial writings of letters as well as unexpected additional strokes, it is also robust against the drawing order and may recognize a letter whatever the temporal order used. Figure 3 illustrates this with an example. A model of letter "*a*" has been learned from a set of training samples that are similar to sample in the left of the figure. On the right, it shows the segmentation obtained for a test sample that has been drawn in reverse order. It may be seen that although the test sample was not drawn as training samples were, the model has been able to correct segment the test sample.
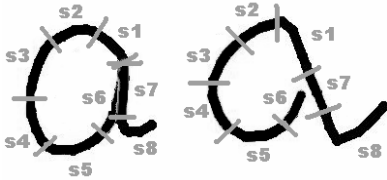

Figure 3. Example of a letter that is correctly segmented by the model (left) where the segmentation of the signal into states are indicated by letters (s1,s2,…). On the right is shown the segmentation of a test sample that has been drawn in an unexpected order. As may be seen the model is able to correctly segment this sample as well.

As suggested by this example, experimental results show that such a relational model performs robust segmentation and is rather insensitive to noisy information such as extra additional strokes, variations in temporal order etc. We investigate here one of these aspects. In order to evaluate sensitivity of the robustness of the segmentation step to perturbations in the ordering of the strokes drawn, we performed experiments on handwriting signals that we artificially corrupted.

Given an input signal consisting of a sequence of a few strokes (separated by pen-up moves), the signal is corrupted by combining a number of three elementary steps that consist in permuting two strokes, reversing the drawing order of a stroke, cutting a stroke in two parts in order to obtain two new strokes. These perturbations introduce a high level of noise in the temporal order of the writings.

We made the noise level vary by corrupting handwriting samples with a varying number of perturbation steps. Hence, we built a test database of samples that have been corrupted with N perturbations of each one of the three elementary steps. For a level N corruption, N "cuts" are first applied, then N "permutations" steps and finally N "reverse" steps. Figure 4 shows the difference between the segmentation

obtained on the original test sample and on the corrupted one, as a function of N. The curve corresponds to the percentage of points for which both segmentations differ. One may see that the first level of perturbation ($N=1$) introduces around 4% error then this rate increases slowly to 9% for $N=10$. Considering the corruption level these results show that relational models are rather insensitive to temporal perturbation.
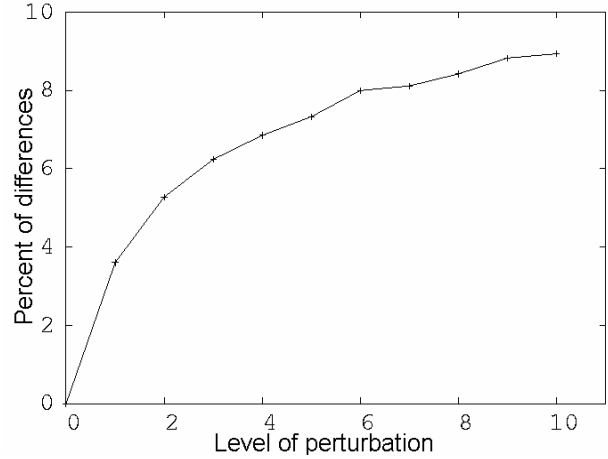

Figure 4. Percent of points for which segmentation of the original and corrupted test sample differs, as a function of the level of perturbation N.

5.3. Recognition

Of course, our models may also be used for character recognition by training a model for each class. However the previous section have shown that our models may score with high likelihood an input sequence which is not complete with respect to the model (e.g. all states are not visited). In order to perform recognition, one has to add a mechanism able to handle this completeness information. This is done by estimating during training the probabilities that each state is observed. This allows computing the probability that a particular segmentation fits well the model. At recognition time, the score of a class is computed as the product of the likelihood computed by the model and of the probability of the correctness of the segmentation.

In a first series of experiments, we compared a HMM system with the purely relational models described in §2. Models have 5 states in both experiments, HMM are classical left-right models. The models are learned with 20 samples per digit. Table 1 shows that relational model significantly outperform HMM and reduce errors rates by about 30%.

In a second series of experiments we investigated the importance of relative features and of local features for recognition. We compared a number of models that use relational and local features by weighting, at recognition time, the local feature likelihoods and the relative features likelihood (Figure 5). When the weight equals 1.0 only local features are used while when the weight is 0.0 only relational features are used. We provide results

for two model topologies, with five and ten states. One may notice first that low recognition rate is achieved when using local features only, and that this performance is highly sensitive to the number of states in the model. By comparison, models using purely relational features perform better and are not much sensitive to the number of states. More importantly, whatever the number of states, it is worth combining local and relative features. Depending on the number of states, adding relational features to local features (that are traditionally used in e.g. HMM systems) allows reducing the error rate by 62% for 5 states models and by 92% for 10 states models.

Table 1. Performance of purely relational models and standard HMMs for isolated on-line digit recognition.

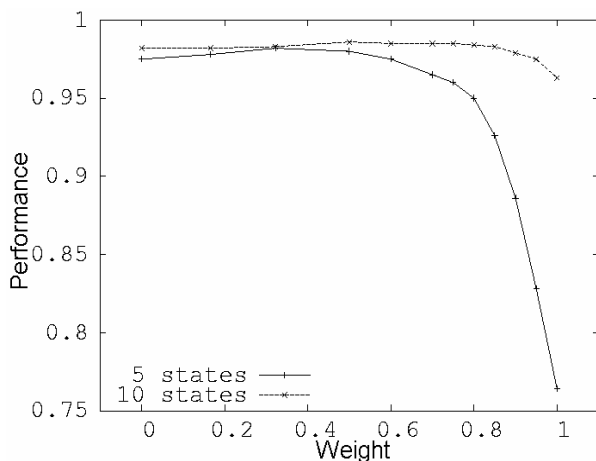| System	Accuracy
HMM	95,3
Relational Model	96,9

Figure 5. Accuracy for purely relational models (weight=0), purely local models (weight=1) and mixed models (weight between 0 and 1).

6. Conclusion

We presented a new modeling framework for on-line handwritten signals. It allows building a variety of models (including traditional Markovian models) that exploit both local and relational features. We detailed inference and learning algorithm for these models and show their intrinsic interest in on-line handwriting processing tasks such as partial matching, diagnosis, sequence recognition. Furthermore, we show how introducing relational features brings much robustness to extra strokes, unusual temporal ordering etc.

7. References

[1] Bara, F., Gentaz, E. & Colé, P. "Early handwriting acquisition and its difficulties". 2005.

[2] Jean-François Cardoso. "Dependence, correlation and Gaussianity in independent component analysis", Journal of Machine Learning Research. Vol. 4, pp 1177-1203, 12/2003.

[3] S.-J. Cho, and J.H. Kim, "Bayesian Network Modeling of Strokes and Their Relationships for On-Line Handwriting Recognition" Proc. Sixth Int'l Conf. Document Analysis and Recognition, pp. 86-90, 2001.

[4] S. Glenat, L. Heutte, T. Paquet, D. Mellier. "Computer-Based Diagnosis of Dyspraxia: the MEDDRAW project", to appear in 12th Conference of the International Graphonomics Society, IGS 2005, Salerno, Italy, June 2005.

[5] Hamstra-Bletz L., DeBie J. and Den Brinker B. "Concise evaluation scale for children handwritting", L.S.Z, 1987.

[6] K. Held and all, "Markov Random Field Segmentation of Brain MR Images". IEEE Trans. on medical imaging,Vol. 16,No. 6, 1997.

[7] Ralf Herbrich, Thore Graepel, and Colin Campbell. "Bayes Point Machines". Journal of Machine Learning Research, 1:245-279, 2001.

[8] Jennifer Hoeting, David Madigan, Adrian Raftery and Chris Volinsky "Bayesian Model Averaging" Statistical Science 14, 382-401, 1999.

[9] Liu, W.-K. Cham, and M.M.Y. Chang, "Stroke Order and Stroke Number Free On-Line Chinese Character Recognition Using Attributed Relational Graph Matching," Proc. 13th Int'l Conf. Pattern Recognition, vol. 3, pp. 259-263, 1996.

[10] Lafferty J., McCallum A., and Pereira F. (2001). Conditional random fields: "Probabilistic models for segmenting and labeling sequence data". International Conf. on Machine Learning, 282–289. Morgan Kaufmann, San Francisco, CA.

[11] Marukatat S., Artières T., Handling spatial information in on-line handwriting recognition, IWFHR 2004.

[12] Ostendorf M., Digalakis V.V., Kimball O.A. (1996). "From HMMs to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition", IEEE Transactions of Speech and Audio Processing, Vol 4, No. 5, pp 360-378.

[13] Pearl J. (1988). "Probabilistic Reasoning in Intelligent Systems : Networks of Plausible Inference". San Francisco, CA : Morgan Kauffmann.

[14] Poritz A. (1982) "Linear predictive hidden markov models and the speech signal," Proc. of ICASSP.

[15] Rosenblum, S., Weiss, P. L., & Parush, S. (2003). "Product and process evaluation of handwriting difficulties". Educational Psychology Review, 15, 41-81.

[16] Tanner M., *Tools for statistical inference,* Springer Verlag, NY, third edition.

[17] Weiss Y. and Freeman W., "Correctness of belief propagation in graphical models of arbitrary topology", Neural Computation.

[18] Weiss Y. and Freeman W., "On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs", IEEE Transactions on Information Theory, Vol. 47, No. 2, Feburary 2001.

[19] J. Zheng, X. Ding, Y. Wu, and Z. Lu, "Spatio-Temporal Unified Model for On-Line Handwritten Chinese Character Recognition". Proc. Fifth ICDAR, pp. 649-652, 1999.