# TANDEM HMM WITH CONVOLUTIONAL NEURAL NETWORK FOR HANDWRITTEN WORD RECOGNITION

*Théodore Bluche* [ab], *Hermann Ney* [bc], *Christopher Kermorvant* [a]

[a]A2iA SA, France
[b]LIMSI CNRS, Spoken Language Processing Group
[c]RWTH Aachen University, Human Language Technology and Pattern Recognition

## ABSTRACT

In this paper, we investigate the combination of hidden Markov models and convolutional neural networks for handwritten word recognition. The convolutional neural networks have been successfully applied to various computer vision tasks, including handwritten character recognition. In this work, we show that they can replace Gaussian mixtures to compute emission probabilities in hidden Markov models (hybrid combination), or serve as feature extractor for a standard Gaussian HMM system (tandem combination). The proposed systems outperform a basic HMM based on either decorrelated pixels or handcrafted features. We validated the approach on two publicly available databases, and we report up to 60% (Rimes) and 35% (IAM) relative improvement compared to a Gaussian HMM based on pixel values. The final systems give comparable results to recurrent neural networks, which are the best systems since 2009.

***Index Terms***— Handwriting recognition, Hidden Markov Model, Convolutional Neural Network

## 1. INTRODUCTION

Handwritten text recognition consists of transforming an image into text. The difficulty of the task is manifold. The high variability of handwriting styles should be reduced by an efficient preprocessing of the image. Segmenting the words into characters is difficult because of the cursive nature of handwriting. Hidden Markov models (HMM) of characters, concatenated to form word models (hence performing an implicit segmentation of words during recognition) conveniently address this problem, and deal with the variability of character lengths. In this approach, a sliding window is scanned horizontally over the image to extract a sequence of observation vectors from the two-dimensional image. Selecting relevant features to extract from the pixels is also a non-trivial issue. Applying for example a PCA on the pixel values of the extracted frames provides an easy solution to this problem, but relies on a good preprocessing. Higher level approaches may consist of handcrafting features (e.g. [1]), which takes time, or to let a system learn them directly [2].

Neural networks, and especially deep neural networks, learn intermediate representations of their inputs, that are useful for a subsequent classification task. Their combination with HMMs improved the performance of both speech recognition [3, 4], and handwriting recognition [5, 6, 2].

With their particular structure, including the notion of receptive fields via weights sharing and distortion invariance with pooling operations, convolutional neural networks (ConvNN) [7] handle conveniently two-dimensional structures such as images, and can incorporate many hidden layers - hence many intermediate representations - while keeping the total number of free parameters relatively small. They have been successfully applied to computer vision problems, particularly to isolated character recognition [8, 9, 7] and handwriting recognition [10, 11].

We compare standard Gaussian HMMs (GHMMs) based on pixel values and on handcrafted features, with the combination of a ConvNN and an HMM in both hybrid [12] and tandem [13] modes. We report significant improvements in isolated word recognition on two publicly available databases, achieving similar performance as LSTM-RNN, which gave the best results on Rimes database since the ICDAR competition in 2009. Many parameters remain to be adjusted, leaving room for further improvements.

The remaining of this paper is organized as follows. Section 2 presents the relation of this work to earlier studies. Section 3 describes our systems. We discuss our experiments and results in Section 4, and conclude with perspectives in Section 5.

## 2. RELATION TO PRIOR WORK

Hammerla et al. [2] automatically extract features from a bottleneck autoencoder. They train a GHMM with these features to obtain frame labels, and further optimize the autoencoder with regularized non-linear Neighbourhood Component Analysis. The results are comparable to those obtained using heuristic features, but not better. In [5], a multi-layer perceptron (MLP) extracts posterior features from consecutive windows of pixels. The authors compare a hybrid and

a tandem approach and show that they perform better than the GHMM baseline, the tandem approach giving the best results. Similarly, Doestch uses a Long Short-Term Memory Recurrent Neural Network [6] and reports improvements in both hybrid and tandem combination.

This work takes advantage of the power of convolutional neural networks on the one hand, and of the combination of neural networks and HMMs on the other hand. It is motivated by the significant improvements reported with these approaches in the literature. It differs from [11, 14] because we do not try to train the graph (or HMM) along with the neural network, but we rather follow the common approach [12, 4] of training them separately. Thus we can explore the tandem combination, for which we report better results. Unlike [14, 10], we focus on off-line handwriting recognition. Unlike [11], we report results on public databases.

## 3. SYSTEM DESCRIPTION

### 3.1. Overview of the training

The aim of the procedure is to train a neural network to predict the HMM state $s$ corresponding to a frame $x$, i.e. $p(s|x)$. These state posterior probabilities can be rescaled by the state priors $p(s)$ (or $p(s)^\alpha$, where $\alpha$ is a tunable parameter (Table 1)) to obtain pseudo-likelihoods. Such network can replace the Gaussian mixtures emission model of a standard GHMM (hybrid scheme [12]). The likelihoods can also serve as features for a new GHMM (tandem scheme [13]). We assign the frame labels of the training dataset with a bootstrapping procedure, which consists in training a standard GHMM and recording the forced alignments.

Hybrid NN/HMM models typically use several consecutive frames as inputs for the neural network, the corresponding target being the label of the central frame. Here we adopt a different approach, where the neural network sees a wider frame of 39 pixels, while the bootstraping feature-GHMM sees only 9 pixel-wide frames. To ensure a coherent labeling, the (big) pixel frame is centered on the smaller frame. Thus, we give the network more context, while keeping the two dimensional structure of the input and getting rid of too much redundant information. The network is trained using the bootstrapping labels. It is then associated with the HMM in hybrid mode to re-align the training data, and the HMM transition model is updated using the alignment statistics. The obtained alignment become the new labels for a second pass of network training.

| **Scaling factor** | 0.0 | 0.1 | 0.5 | 1.0 |
|---|---|---|---|---|
| **WER** | 14.7% | 14.3% | 12.5% | 14.3% |

**Table 1**. Effect of state priors scaling factor $\alpha$ on WER on Rimes development set with the hybrid model.

### 3.2. Pre-processing

We first correct the slant in the image with a projection-based method [15]. Each image is cropped to the bounding box of the word (found after an adaptive thresholding binarization). Then the contrast is enhanced (5% of darkest pixels are mapped to black, 70% lightest to white, and a linear interpolation in between). Finally, we add 20 white pixels on left and right to model the empty context. Pixel-based systems require fixed size frames: in these cases we rescale the image to a fixed height of 72px, with different scaling factors applied to three different zones (ascenders, descenders and core region). We find these zones with the method described in [16].

### 3.3. Feature extraction

The handcrafted features comprise 26 geometrical and statistical features and 8 directional features based on histogram of gradients, and are computed from a sliding window of width 9 pixels shifted by 3 pixels along the word image. We then apply first-order regression to get 68-dimensional feature vectors [1]. The inputs of the pixel-GHMM and the hybrid model are the pixel values of the frames. The image is first normalized in height. Then we apply a sliding window of 39 pixels width with a shift of 3 pixels, and rescale the extracted frames to 32×32px for computation efficiency with the neural network library. A PCA decorrelates the features and reduces the vectors' dimension to 30 for the pixel-GHMM.

### 3.4. HMM modelling

The topology of character HMM remains the same throughout the experiments. We build six-state models for characters (lower and upper case and accentuated letters, digits and some punctuation - 81 for Rimes, 78 for IAM), with self-loops and transitions to the next state only (see Table 2). We add two-state "blank" character models for the beginning and the end of words, corresponding to empty context. The HMM are built and trained using Kaldi [17].

| **Number of states** | 3 | 4 | 6 | 8 |
|---|---|---|---|---|
| **WER - Rimes** | 25.2% | 20.8% | 17.4% | 23.0% |
| **WER - IAM** | 32.5% | 27.1% | 24.3% | 27.2% |

**Table 2**. Effect of number of states in character models. WER are computer with about 22 Gaussians per mixture (feature-GHMM)

### 3.5. Convolutional neural networks

The topology of the convolutional neural networks is similar to LeNet5 [8], as shown on Figure 1. They consist of three convolutional layers with 32, 64 and 128 feature maps and 5×5px filters, followed by max-pooling operations. On top

of this architecture is a fully connected hidden layer of 700 hidden units, and an output softmax layer with as many units as there are states in the HMM models (490 for Rimes, 472 for IAM). They are trained on a GPU using Alex Krizhevsky's *cuda-convnet* library[1]. We randomly split the original training set of words into a network training set (90%) and a network development set (10%). After training, we re-align the training data with the ConvNN and the HMM in hybrid combination, and perform a second pass of training.

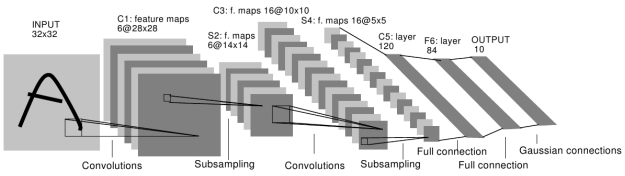


**Fig. 1**. LeNet5 convolutional neural network topology (from [8]).

### 3.6. Combination of ConvNN and HMM

We tested two combination schemes. For the ConvNN/HMM hybrid, the predictions of the network are rescaled by the state priors and used directly by the HMM for decoding. The ConvNN-GHMM tandem is a standard GHMM system, which features are obtained by applying a PCA reduction to 50 dimensions on the logarithm of the rescaled predictions.

### 3.7. Language model

We take into account the prior distribution of words using a unigram language model. The *optical scaling* factor is tuned on an independent validation set (see Figure 2).

## 4. RESULTS

### 4.1. Databases

**Rimes [18] -** We work on the word recognition task proposed for the ICDAR 2009 competition. The training set is composed of 44,197 images with 4,508 unique words and the development set contains 7,542 images with 1,636 unique words. The systems are tested on the ICDAR 2009 evaluation set, made of 7,464 images, with a closed vocabulary composed of either 1,612 words (WR2 setting, 0% OOV) or 5,335 words (WR3 setting, 0% OOV).

**IAM [19] -** We also extract words from the IAM database. The training set is composed of 53,841 words, the development set is made of 8,566 words, and the set test contains 17,616 words. We used the smallest closed vocabulary, made of only the words present in the annotations for development,

[1]http://code.google.com/p/cuda-convnet/

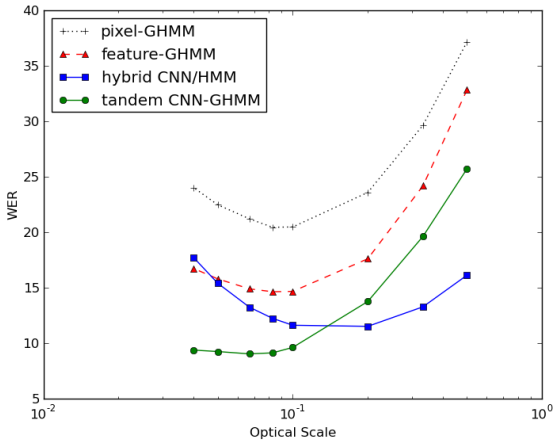


**Fig. 2**. Optical scale tuning on Rimes development set.

and we carried out the test experiments with the vocabulary containing all annotations of the datasets (i.e. similar to the WR3 setting for Rimes). This is the same experimental setup as [1] except that we have a unigram language model, estimated from the training set annotations.

### 4.2. Result analysis

On Rimes database, we first trained two standard GHMMs, based on 32x32 pixel frames reduced by PCA to 30 dimensions (pixel-GHMM) and on 9px-wide frames from which 68 features are extracted (feature-GHMM). The latter is better in terms of Word Error Rate (WER), so we bootstap the training of the ConvNN with the forced alignments obtained with this system. The predictions of the networks, scaled by state priors, are used in hybrid scheme with the HMM (ConvNN/HMM hybrid), and also reduced to 50 dimensions by PCA as features for a new GHMM (ConvNN-GHMM tandem). We then build a context-dependent (CD) system, based on a CART tree with one root for all characters and about 1,000 leaves, using Kaldi's mechanism. The final system is then further optimized using a few iterations of Maximum Mutual Information (MMI) training.

The HMM, or rather the transition model, used in the hybrid experiments is derived from the feature-GHMM. We studied the impact of the transition model in hybrid results. The WERs on Rimes development set are shown on Table 3. Between the first and second iteration of ConvNN training, we update the transition probabilities of the HMM to fit the new forced alignments. The effect of this update is small (no improvement using the first-pass ConvNN, 0.03% absolute improvement using the re-trained network). Interestingly, using the transition model derived from the pixel-GHMM does not cause a dramatic degradation of the performance (only 0.22% absolute), showing that the transition model only plays

a minor role in the final performance.

The results are presented in Table 4, and compared to the best known results on these tasks. The hybrid system brings a relative improvement of about 30% compared to feature-GHMM, and the final context-dependent MMI trained tandem yields a relative improvement of about 45% for Rimes WR2 setting. When comparing the systems using the same inputs, i.e. pixel-GHMM and ConvNN-based models, we record up to 60% relative improvement. The results are comparable to the state-of-the-art recurrent neural networks, which are the best single systems on this task, and part of the combination of systems presented in [20].

On IAM database, we followed the same approach, with the same sliding windows, PCA dimensions, and neural network architecture. The number of states, optical and prior scaling factors are optimized, but found to be close to the optimal values for Rimes. The results are shown on Table 5. The improvements brought by the ConvNN in hybrid mode and by the handcrafted features are similar, compared to the pixel-GHMM. The tandem system is again better ; the final relative improvement is about 35%. For fair comparison with [1], in which no language model is applied, we also report the results without LM, and observe that our system is still better than a single context-dependent GHMM, and close to a combination of a MLP/HMM hybrid and two GHMMs.

| **Hybrid model** | **WER** |
|---|---|
| First-pass CNN, original trans. model | 12.9% |
| First-pass CNN, updated trans. model | 12.9% |
| Second-pass CNN, original trans. model | 12.3% |
| Second-pass CNN, updated trans. model | 12.2% |
| Second-pass CNN, pixel-GHMM trans. model | 12.4% |

**Table 3**. Effect of transition model on WER on Rimes development set.

| **Model** | **Rimes-WR2** | **Rimes-WR3** |
|---|---|---|
| pixel-GHMM | 19.8% | 21.4% |
| feature-GHMM | 14.6% | 16.4% |
| ConvNN/HMM hybrid | 10.0% | 11.7% |
| ConvNN-GHMM tandem | 8.5% | 9.9% |
| + CD | 8.0% | 9.4% |
| + MMI training | **7.9%** | **9.2%** |
| 7 RNN + HMM [20] | - | **4.8%** |
| RNN [21] | **6.8%** | 9.0% |
| Tandem LSTM-HMM [6] | - | 9.7% |

**Table 4**. Word error rate on the test set for the different systems on the ICDAR-2009 evaluation set for two different vocabulary sizes (WR2 and WR3).

| **Model** | **Dev** | **Eval** |
|---|---|---|
| pixel-GHMM | 24.4% | 31.7% |
| feature-GHMM | 20.6% | 24.9% |
| ConvNN/HMM hybrid | 19.5% | 25.2% |
| ConvNN-GHMM tandem | 14.9% | 20.6% |
| + CD | **14.5%** | **20.5%** |
| - LM | - | **23.7%** |
| MLP/HMM + 2 GHMM [1] | - | **21.9%** |
| CD-GHMM [1] | - | 32.7% |

**Table 5**. Word error rate on the dev and test sets for the different systems on IAM database

## 5. CONCLUSIONS AND FUTURE WORK

The experiments described in the previous sections showed that the combination of a convolutional neural network and a HMM can achieve good performance on isolated handwritten word recognition. The presented systems can outperform both a Gaussian HMM based on the same pixel inputs, and a GHMM working with handcrafted features. The tandem combination is consistently better than the hybrid approach, which confirm prior studies, and the results we report on Rimes are comparable to the state-of-the-art LSTM networks, which potentially use the context of the whole image to make the predictions at a given position. A number of parameters must be optimized in these models: the sliding window size and shift, the topology of the convolutional neural network, so the proposed systems are promising but surely not optimal. It would also be interesting to study the impact of the bootstraping model's quality on the final system performance. Improvements can be envisioned if we take advantage of the structure of the convolutional network, particularly the replicated weights, and build a Space-Displacement Neural Network (SDNN) [14, 10] to extract the ConvNN features.

## 6. ACKNOWLEDGMENTS

This work was partly achieved as part of the Quaero Program, funded by OSEO, French State agency for innovation and was supported by the French Research Agency under the contract Cognilego ANR 2010-CORD-013. H. Ney was partially supported by a senior DIGITEO Chair grant from Ile-de-France.

## 7. REFERENCES

[1] A.-L. Bianne, F. Menasri, R. Al-Hajj, C. Mokbel, C. Kermorvant, and L. Likforman-Sulem, "Dynamic and Contextual Information in HMM modeling for Handwriting Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 10, pp. 2066 – 2080, 2011.

[2] N.Y. Hammerla, T. Plötz, S. Vajda, and G.A. Fink, "Towards Feature Learning for HMM-based Offline Handwriting Recognition," in *International Workshop on Frontiers in Arabic Handwriting Recognition*, 2010.

[3] F. Grezl, M. Karafiat, S. Kontar, and J. Cernocky, "Probabilistic and Bottle-Neck Features for LVCSR of Meetings," in *International Conference on Acoustics, Speech and Signal Processing*, 2007, vol. 4, p. 757.

[4] G. Dahl and D. Yu, "Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.

[5] P. Dreuw, P. Doetsch, C. Plahl, and H. Ney, "Hierarchical Hybrid MLP/HMM or rather MLP Features for a Discriminatively Trained Gaussian HMM: A Comparison for Offline Handwriting Recognition," in *International Conference on Image Processing*, 2011.

[6] P. Doetsch, "Optimization of hidden markov models and neural networks," M.S. thesis, RWTH Aachen University, Aachen, Germany, Dec. 2011.

[7] Y. Le Cun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," *International Symposium on Circuits and Systems*, pp. 253–256, May 2010.

[8] Y. Le Cun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[9] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Convolutional Neural Network Committees For Handwritten Character Classification," in *International Conference of Document Analysis and Recognition*, Beijing, 2011, vol. 10, pp. 1135–1139.

[10] Y. Bengio, Y. Le Cun, and D. Henderson, "Globally trained handwritten word recognizer using spatial representation, convolutional neural networks and Hidden Markov Models," in *Neural Information Processing System*, 1994.

[11] Y. Le Cun, L. Bottou, and Y. Bengio, "Reading checks with multilayer graph transformer networks," in *International Conference on Acoustics, Speech, and Signal Processing*, 1997.

[12] H. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach, Chapter 7*, vol. 247 of *The Kluwer international series in engineering and computer science: VLSI, computer architecture, and digital signal processing*, Kluwer Academic Publishers, 1994.

[13] H. Hermansky, D. P. W. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," *International Conference on Acoustics Speech and Signal Processing*, vol. 3, no. 28149, pp. 1635–1638, 2000.

[14] Y. Bengio, Y. Le Cun, C. Nohl, and C. Burges, "LeRec: a NN/HMM hybrid for on-line handwriting recognition.," *Neural Computation*, vol. 7, no. 6, pp. 1289–1303, 1995.

[15] R. Buse, Z. Q. Liu, and T. Caelli, "A structural and relational approach to handwritten word recognition.," *IEEE transactions on systems, man, and cybernetics*, vol. 27, no. 5, pp. 847–61, Jan. 1997.

[16] A. Vinciarelli and J. Luettin, "A new normalisation technique for cursive handwritten words," *Pattern Recognition Letters*, vol. 22, pp. 1043–1050, 2001.

[17] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, Dec. 2011.

[18] E. Augustin, M. Carré, E. Grosicki, J.-M. Brodin, E. Geoffrois, and F. Preteux, "RIMES evaluation campaign for handwritten mail processing," in *Workshop on Frontiers in Handwriting Recognition*, 2006, number 1.

[19] U. V. Marti and H. Bunke, "The IAM-database: an English sentence database for offline handwriting recognition," *International Journal on Document Analysis and Recognition*, vol. 5, no. 1, pp. 39–46, 2002.

[20] F. Menasri, J. Louradour, A-L. Bianne-bernard, and C. Kermorvant, "The A2iA French handwriting recognition system at the Rimes-ICDAR2011 competition," in *Document Recognition and Retrieval Conference*, 2012, vol. 8297.

[21] E. Grosicki and H. ElAbed, "ICDAR 2009 Handwriting Recognition Competition," in *International Conference on Document Analysis and Recognition*, 2009.