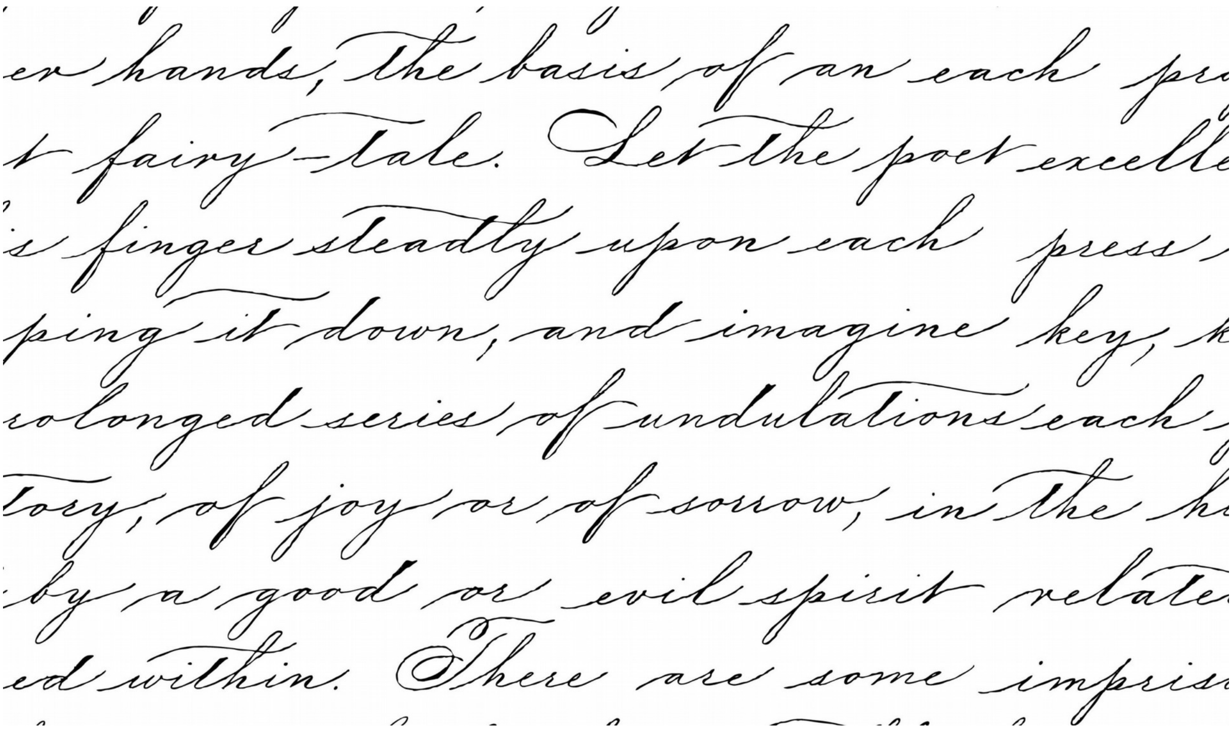
Handwriting Recognition

HUARD Alicia



Submitted on 28 April 2020

Abstract

Handwriting Recognition (HWR), also know as Handwritten Text Recognition (HTR), is the ability of a computer to recognize an understandable handwriting. This handwriting can come from paper documents, photographs and other.

The image of the written text can be recognized “off line” by optical scanning of a piece of paper or “on line” by detecting the movement of the tip of the pen on a touch-screen.

This reading sheet explains Handwriting Recognition through the comparison of different recognition systems such as Hidden Markov Models (HMM) and neural network.

Summary

Introduction

It’s around the fourth millennium BC, who appears the first handwriting. Writing makes it easier to communicate at a distance, but also over time.

But since the advent of the digital age, handwriting had almost no interest. The digitization of information has taken an important place these days and that why we need applications that allow fast scanning.

Artificial intelligence has encountered a major problem with handwriting recognition since 1950.
Over the years, new learning algorithms have appeared to try to solve it.

Recognizing and understanding handwriting uses all the functions of artificial intelligence such as:

- visual perception, which detects the text of a displayed image
- the follow-up of a sequence, which makes it possible to follow the course of a writing
- pattern recognition algorithm, which allows us to recognize characters
- automatic language processing, which differentiates words and sentences.

And finish with a semantic modeling to understand the meaning of words or sentences.

Problematic: How does artificial intelligence enable handwriting recognition?

Classification

Assigning a label to an object will help us to recognize it. Handwriting recognition is therefore a classification problem.

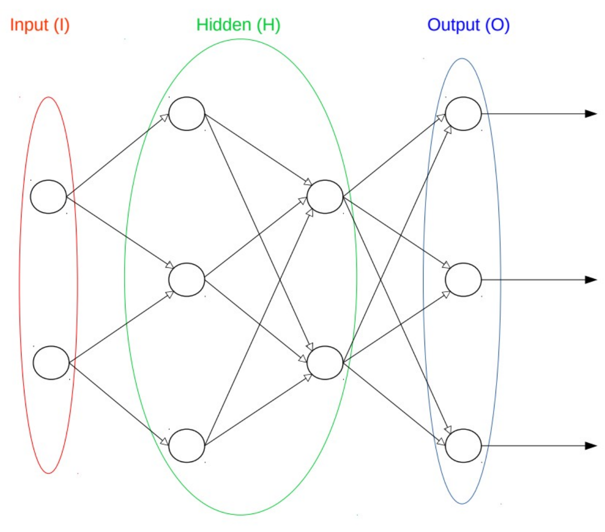
For example, for a photograph of a word or a sentence we seek to recognize all of the characters. The existence of time series in a word or a sentence makes recognition more complex. Their sequence is not random and that why it’s therefore important to model the dependence between the data during recognition.

2.1. Neural Network

A neural network is an assembly of interconnected neurons. There are several forms of neural networks such as the formal neuron, the perceptron and the multi-layer perceptron.

So I’m going to tell you about the multi-layer perceptron that appeared in 1974, which is capable of handling non-linear phenomena. As the name suggests, it is organized into several layers, the input layer, the hidden layers and the output layer.

Here is a diagram:



Multilayer perceptron

The input layer is not a real layer, it is a data carrier.

The hidden layers form a set of transformations of the input data and allow to execute an action at the output layer.

Each neuron in the output layer illustrates a class.

A deep neural network is a multi-layer perceptron that has more then 3 layers. The mechanism is different, the results of the first layer will serve as an input to the next and so on.

The weights of each layer are learned separately. The weights close to the output layer are strongly modified while the others are less. Deep networks therefore make it possible to manage very large data vectors.

Except that like neural networks, they don't use internal memory, however handwriting recognition is a problem with time dependencies.

2.2. Recurrent Neural Network

The recurrent neural network is a dynamic classifier that has internal memory in on of the hidden layers.

In a recurring network at least one of the layers must be recurrent. This is explained by the fact that for a layer *h*, the neurons *n* receive as inputs the outputs oh the neurons of the layer *h-1* at time *t* and the outputs of all the neurons of the layer *h* to *t-1*.

The weighted sum of each neuron *n* is modified compared to a multi-layer perceptron, it is therefore necessary to add a term in order to take account of the recurrence:

J is the number of neurons on the *h-1* layer and R is the number of *n* on the *h* layer.

$$z_n^{h,t} = \sum_{j=1}^{J} (z_j^{h-1,t} w_{nj}) + \sum_{r=1}^{R} (z_r^{h,t-1} w_{nr})$$

This architecture allows better performance on a problem such as handwriting recognition.

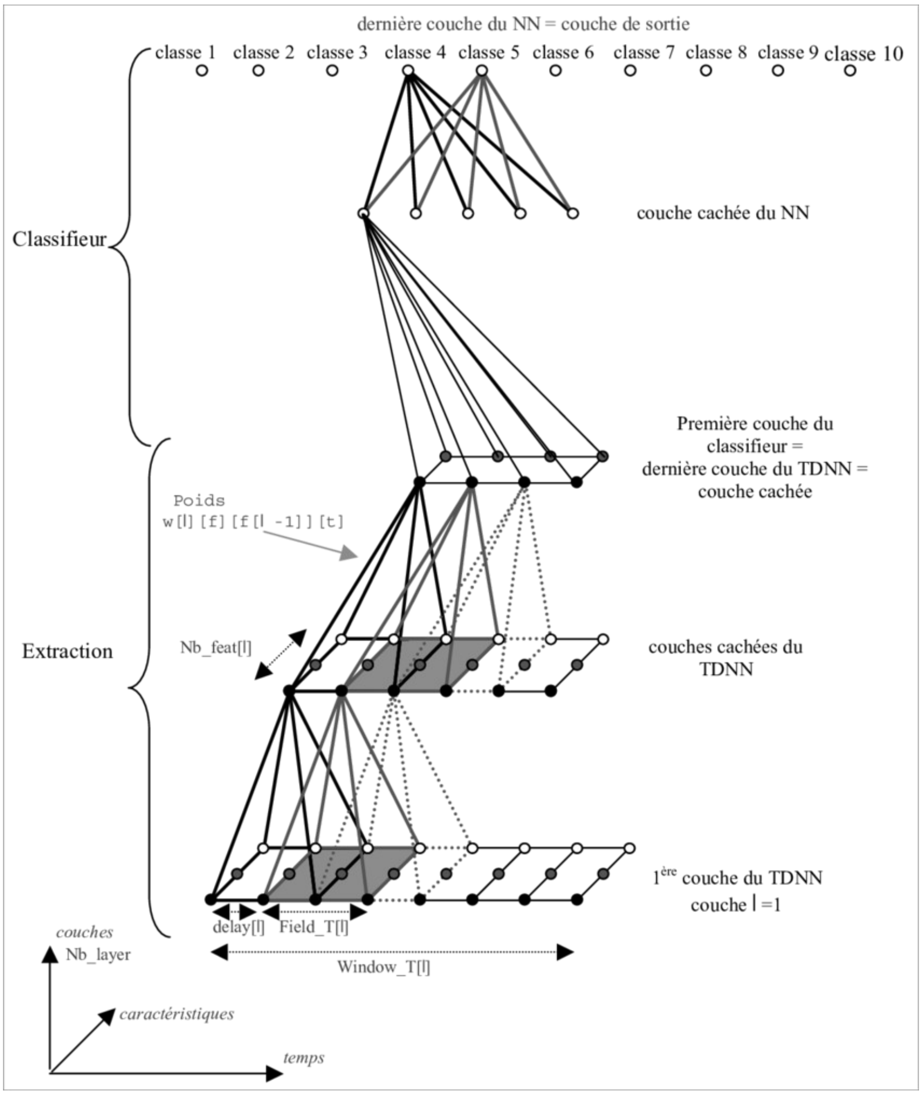
2.3. Time Delay Neural Network (TDNN)

TDNNs are convolutional networks. Initially, they were used in speech recognition, but they are now usable int the recognition of some characters such as numbers and words.

TDNN is delay network used for sequential data, therefore particularly suitable for handwriting recognition.

A TDNN has two configurable main parts:

- the lower layers
- classic multi-layer perceptrons



Characteristics of the parts

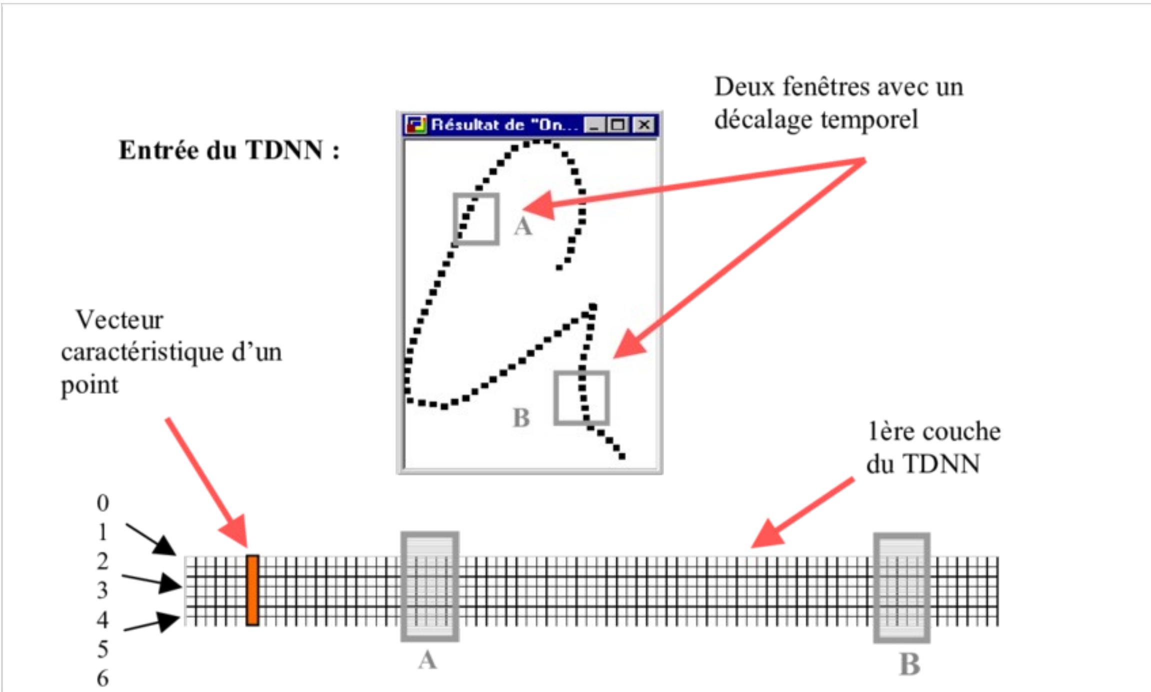
Classifier:

- the number of layers (*NN_nb_layer*)
- the number of neurons in each layer (*NN_nb_neurons*)

Extraction:

- the number of layers (*nb_layer*)
- the number of neurons of each layer according to the time direction (*window_T*)
- the number of neurons of each layer according to the characteristic direction (*nb_feat*)
- the size of the time window (*field_T*)
- the delay between each window (*delay*)

The first layer of the classifier part corresponds to the last layer of the extraction part.



Handwriting Recognition

Automatic handwriting recognition is the process of moving fro an image containing textual information to text in electronic format. Transforming these paper documents into digital documents makes it possible to be able to process them using information systems.

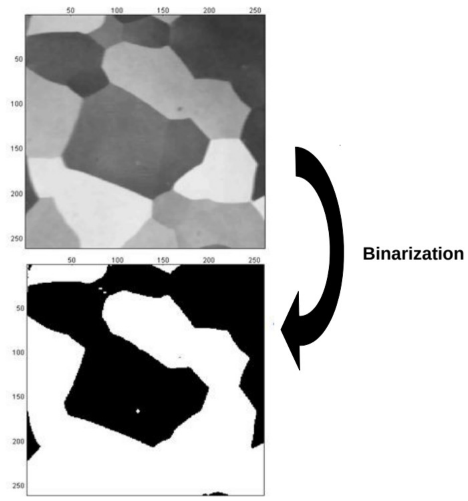
Handwriting recognition is divided into four parts: preprocessing, feature extraction, recognition and post-processing.

3.1. Preprocessing

The preprocessing step on small images of words or sentences improves recognition by normalizing the images or reducing the amount of noise. It is not mandatory to do this step, however it has been found that the use of preprocessing improves recognition.

To reduce the amount of information in an image, binarization can be used. Whereas to normalize the aspect of the images, you have to use the slope correction.

It is important to identify the most suitable binarization, as this step can lead to a loss of information.



The slope correction modifies an image to position the baseline horizontally.

Slope correction

3.2. Feature extraction

The characteristics presented to the recognition system are essential, the final results depending on their quality. Too much representation space can have a detrimental effect on recognition.

The characteristics have several variations: according to the scale of the representation, or according to the generation of their representation.

The generation of their representation can be separated into two categories, those defined by the designer and those automatic. One of the advantages of automatically generating characteristics is that it does not inject prior knowledge about the task. This allows, for example, the use of an automatic feature extractor for handwriting recognition. But also to link these structures to create representations of an increasing level of abstraction (low-level, medium level, high level). They can switch to the recognition system.

3.3. Recognition

This step deals with the heart of the problem. Handwriting recognition can be divided into two tasks:

- recognition of an isolated character
- recognition of a word or sentence.

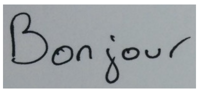
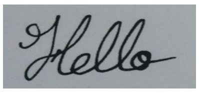
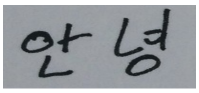
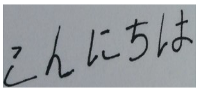
The Hidden Markov Models are best known for handwriting recognition. After having classified a sequence of characteristic vectors into an output sequence, it is now necessary to perform a post-processing step, in order to correct any errors.

3.4. Post-processing

The post-processing step allows you to correct errors due to the recognition step. It allows you to correct minor errors that don’t make sense for the language being processed.

The removal of ambiguities can be done at the level of characters to model a word or level of words for a sentence. There are three frequently used methods, lexicon decoding, dictionaries an N-grams.

The easiest way to correct these errors is to use a dictionary. It is enough to measure the distance between a produced word and the words of the dictionary and to find the nearest word, after which correct it.

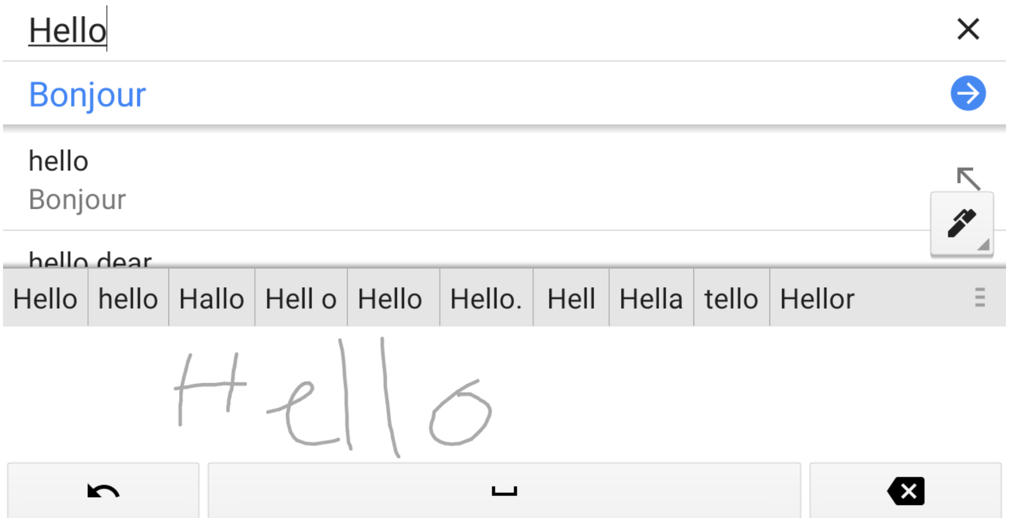
Pictures	Recognized text	Language
	Bonjour	French
	Hello	English
	안녕	Korean
	こんにちは	Japanese

Applications

4.1. Language recognition

The goal is to distinguish one language from another in a digital document (text, picture, audio).

Language recognition is required when we want to translate a text into another language. For example on some website, you are offered to translate it into your language.



4.2. Keyword detection

Searching for a keyword or sentence is important because it allows a user to access the item in a text more quickly. Example the shortcut Ctrl+f, which allows us to find the location of the search expression.

There are two main categories of methods, one search from a text request and the other from a picture.

Conclusion

The automatic analysis of scanned paper documents is a complex task, as we have seen above, which combines different techniques and methods. The existence of different algorithms and different statistical classification methods allows different approaches to the problems of handwriting recognition.

Even if there are still problems with handwriting recognition, one day we will find a solution which will allow an accurate handwriting recognition.

Bibliography

Ahmad, A., Viard-Gaudin, C., Khalid, M., and Yusof, R. (2004). Online handwriting recognition using support vector machine. *IEEE Region 10 Conference TENCON.*

https://scholar.google.fr/scholar?hl=fr&as_sdt=0%2C5&as_vis=1&q=Online+hand-+%09writing+recognition+using+support+vector+machine.&btnG=

Almazan, J., Gordo, A., Fornes, A., and Valveny, E. (2013). Handwritten Word Spotting with Corrected Attributes. *IEEE International Conference on Computer Vision.*

https://scholar.google.fr/scholar?hl=fr&as_sdt=0%2C5&as_vis=1&q=+Handwritten+%09Word+Spotting+with+Corrected+Attributes.&btnG=

Baldi, P. and Chauvin, Y. (1994). Smooth On-Line Learning Algorithms for Hidden Markov Models. *Neural Computation.*

https://scholar.google.fr/scholar?hl=fr&as_sdt=0%2C5&as_vis=1&q=Smooth+On-Line+Learning+Algorithms+for+%09Hidden+Markov+Models.&btnG=

Andrew, Y. and Jordan, M. (2002). On discriminative vs. generative classifiers : A comparison of logistic regression and naive bayes. *Advances in neural information processing systems.*

https://scholar.google.fr/scholar?q=On+discriminative+vs.+generative+classifiers+:+A+comparison+of+logistic+regression+and+naive+bayes&hl=fr&as_sdt=0&as_vis=1&oi=scholart

Bengio, Y., De Mori, R., Flammia, G., and Kompe, R. (1992). Global optimization of a neural network-hidden Markov model hybrid. *IEEE Transactions on Neural Networks.*

https://scholar.google.fr/scholar?hl=fr&as_sdt=0%2C5&as_vis=1&q=Global++%09optimization+of+a+neural+network-hidden+Markov+model+hybrid.&btnG=

Bourlard, H. and Morgan, N. (1991). Merging multilayer perceptrons and hidden Markov models : Some experiments in continuous speech recognition. *Neural Networks : Advances and Applications.7*

https://scholar.google.fr/scholar?hl=fr&as_sdt=0%2C5&as_vis=1&q=Merging+multilayer+perceptrons+and++%09hidden+Markov+models+%3A+Some+experiments+in+continuous+speech+%09+%09recognition.&btnG=