Text Line Segmentation in Handwritten Document Using a Production System

Stéphane Nicolas, Thierry Paquet, Laurent Heutte
Laboratoire PSI FRE CNRS 2645 - Université de Rouen
Place E. Blondel, UFR des Sciences et Techniques
F-76821 Mont-Saint-Aignan cedex, France
{Stephane.Nicolas, Thierry.Paquet, Laurent.Heutte}@univ-rouen.fr

Abstract

We present in this paper a digitization project of cultural heritage manuscripts and we discuss the underlying problems, particularly those relative to document analysis. Considering the drawbacks of traditional methods for text line extraction in handwritten documents, we propose to adopt a new approach for handwritten page segmentation, based on a traditional problem solving framework used in Artificial Intelligence.

1. Introduction

Libraries and museums contain collections of a great interest which cannot be shown to a large public because of their value and their state of conservation, therefore preventing the diffusion of knowledge. Today with the development of the numerical technologies, it is possible to access to this cultural patrimony by substituting the original documents by numerical high quality reproductions allowing to share the access to the information while protecting the originals. These last years, numerous libraries have started digitization campaigns of their collections. Faced to the amount of the digitized images produced, the development of digital libraries allowing access to these data by means of suitable search engines becomes a major stake for the valuation of this cultural patrimony. However such a task is difficult and expensive, and therefore requires prior studies concerning the technical appropriate means to use. Though they have a great interest for the study and the interpretation of literary works, modern manuscripts have been adressed by few digitization programs because of the complexity of such documents and the lack of appropriate tools.

The work presented in this paper is related to the Bovary Project, a digitization project of modern manuscripts concerning especially FLAUBERT's manuscripts (famous French novelist of the 19th century). After a brief presentation of the Bovary Project and pursued aims, we focus especially on the problem of image analysis of such particular manuscripts in section 3, and we overview the existing methods for handwritten document analysis in section 4. Finally we propose in section 5 a new approach for text line extraction in handwritten documents, based on problem solving formalism. First results and proposed improvements are discussed in sections 6 and 7.

2. The Bovary Project

In 2003 the municipal library of ROUEN has begun a program for digitizing its collections. For this purpose an efficient system of digitization allowing a high resolution display of the digitized documents has been purchased. One of the first aims of this program is the digitization of 5000 original drafts of Gustave FLAUBERT's novel "Madame Bovary". This manuscript set constitutes the genesis of the novel, in other words the successive drafts which highlight the writing and rewriting processes of the author. This digitization task is now completely achieved. The final aim of this program is to provide an hypertextual edition[1] allowing an interactive and free web access to this material. Such an electronic edition will be of great interest for researchers, students, and anyone curious to see FLAUBERT's manuscripts, especially because there is no critical edition of a full literary work of this author available on the web. This project called "Bovary Project" is a multidisciplinary project which implies people from different fields of interest: librarians, researchers in literary sciences and researchers in computer science.

From the document analysis point of view, FLAUBERT's drafts have a complex structure (see figure 1.a). They contain several blocks of text arranged in a non linear way, and numerous editorial marks (erasures, word insertion,., ...). So these manuscripts are very hard to decipher and interpret. The production of an electronic version of this corpus is a challenging task which has to respect some requirements in order to meet user needs.

[1] a first prototype of this hypertextual edition is available at the following URL: http://www.univ-rouen.fr/psi/BOVARY/

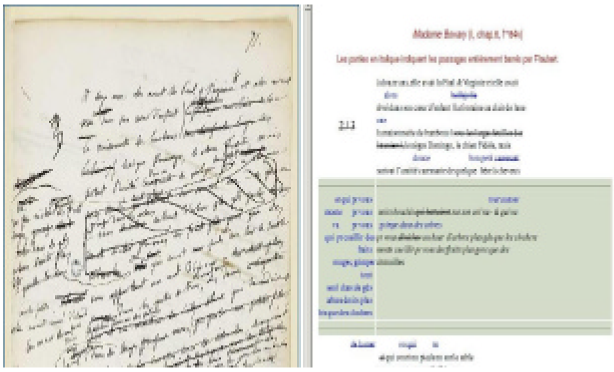

Fig. 1: A manuscript (a) and its associated diplomatic transcription (b)

3. Critical edition and manuscript transcription

In literary sciences the study of modern manuscripts is known as genetic analysis. This analysis concerns the graphical aspect of the manuscripts and the successive states of the textual content. The nature of a manuscript is dual. A manuscript should be considered as a pure graphical representation or as a pure textual representation. According to J. Andre [1], a manuscript is a text with graphical interest. As modern manuscripts reflect the writing process of the author, they may have a complicated structure and may be difficult to decipher. So in a genetical edition, transcriptions are generally joined to the facsimile of manuscripts. A transcription allows an easier reading of the manuscript. One can distinguish two types of transcription: the linear one and the diplomatic one. A linear transcription is a simple typed version of the text, which uses an adapted coding to transcribe, in a linear way, complex editorial operations of the author (deletation, insertion, substitution) sometimes located over one or several pages. On the contrary, the diplomatic transcription (fig. 1.b) has to respect the physical aspect of the manuscript, i.e. the disposition of graphical elements in the page (text line, erasure, insertion,...). For the Bovary Project needs, we are mainly interested by diplomatic transcription, because Flaubert's manuscripts are too complex, so that linear transcriptions cannot help the reader to decipher them. For a given page of manuscript, the production of a diplomatic transcription involves two different steps: first, one has to decipher each word of the page and then type, using a text editor, a full ASCII plain text version of the page, while respecting as much as possible the layout of the manuscript. Considering the extreme complexity of such manuscripts and the amount of documents to be processed, transcription is a very tedious task which can be performed only by few scholars or Flaubert specialists. But we think that the difficulty of the transcription task can be greatly reduced by providing the transcribers some automatic tools, such as image processing or document image analysis tools, integrated in an unified environment dedicated to the edition of diplomatic transcriptions. This allows to extract automatically, or using interactions with the transcriber, the informative entities of the document, and eventually include automatic word image transcription thanks to some automatic handwritten word recognition engine. This implies to be able to extract first the physical layout of the document. This is the aim of document image analysis.

4. Handwritten document image analysis: an overview

Document analysis is a crucial step in document processing, which consists in extracting the physical layout of a given document from its low level representation (image). Numerous methods have been proposed for the analysis of machine printed documents. Among the most popular ones, we can cite Kise's method based on area Voronoi diagram, O'Gorman's Docstrum method based on neighbor clustering and Nagy's X-Y cut based on the analysis of projection profiles (see [2] for an overview). These methods provide good results on printed documents, but are not directly adaptable to handwritten documents, because they generally take only into account global features on the page, and are thus dedicated to well structured documents. Unlike printed documents, handwritten documents have a local structure prone to an important variability: fluctuating or skewed text lines, overlapping words, unaligned paragraphs,... To cope with this local variability, the methods proposed in the litterature for text line segmentation in handwritten documents are generally "bottom-up" and based on local analysis. [3] proposes an approach based on perceptual grouping of connected components of black pixels. Text lines are iteratively constructed by grouping neighboring connected components according to some perceptual criteria such as similarity, continuity and proximity. Hence local constraints on the neighborhing components are combined with global quality measures. To cope with conflicts, the method integrates a refinement procedure combining a global and a local analysis. According to the author the proposed method cannot be applied on degrated or poorly structured documents, such as modern authorial manuscripts. A method based on a shortest spanning tree search is presented in [4]. The principle of the method consists in building a graph of main strokes of the document image and searching for the shortest spanning tree of this graph. This method assumes that the distance between the words in a text line, is less than the distance between two adjacent text lines. In [5] an iterative hypothesis-validation strategy based on Hough transform is proposed. The skew orientation of handwritten text lines is obtained by applying the Hough transform to the center of gravity of each connected components of the document image. This allows to generate several text line hypothesis. Then a validation is performed to eliminate erroneous alignments among connected components using contextual information such as proximity and direction continuity criteria. According to the authors this method is able to detect text line in handwritten documents which may contain lines oriented in several directions, erasures and annotations between main lines. An algorithm based on the analysis of horizontal run projections and connected components grouping and splitting procedures is presented in [6]. First the image is partionned into vertical strips and then an analysis of the run projections on each strip is applied. This method allows to deal with fluctuating or skewed text lines and to preserve the punctuation. [7] proposes a method for line detection and segmentation in historical church registers. This method is based on local minima detection of connected components and is applied on a chain code representation of the connected components. The idea is to gradually construct line segments until an unique text line is formed. This algorithm is able to segment text lines closed to each other, touching text lines and fluctuating text lines.

The main problem of these methods is that they generally take local decisions during the grouping process, and they sometimes fail to find the "best" segmentation when dealing with complex documents like modern manuscripts. Furthermore these methods do not use prior knowledge nor express it explicitly, thus making the adaptation of the system to different classes of documents difficult. To avoid these problems we propose a methodology based on Artificial Intelligence: problem solving using production systems.

5. Text line segmentation using a production system

Production systems are a methodology used in Artificial Intelligence for solving problems. They have been used successfully in many problems such as scheduling tasks, game solving, ... but this methodology is general enough to be applied to different types of problems. The main idea is to consider that a problem can be solved by finding a path in the state space of the problem under consideration.

Formally a production system is composed of three main elements: a global database, used as a description of a given problem state, a set of production rules which are applied on the global database to produce new states of the search space, and a control system allowing to find a path in the search space between the initial state and some final goal state. In a production system, the global database is manipulated thanks to some defined operations (the production rules), under the control of a global control strategy (search procedure) [8]. Solving a problem with a production system consists in searching a path in the space of all the alternatives, by developping a search tree.

When using a production system to solve a given problem, one has to define explicitly the problem to be solved, and the three major elements of the production system: the global database, the set of production rules and the control system.

The problem we consider here is that of text line segmentation in handwritten documents using an ascending grouping process. The aim is to cluster the connected components of the document into homogeneous sets, corresponding to the text lines of the document.

5.1. Problem representation and global database

When solving problems using a production system, one has to choose an adapted data structure (or database) to represent symbolically or numerically the states of the problem to be solved. This database can be a vector, a matrix, or a tree, depending on the nature of the problem. We consider a textual document as a set of objects with spatial relations between them. The objects we consider are text lines and connected components. A graph is a natural way to represent relations between objects, so we have choosen such a data structure as global database of our production system. The nodes of the graph represent the objects of the document (connected components, text lines), and the edges represent adjacency relations between the objects. The nodes of two adjacent objects in the document image are linked by an edge in the graph. This edge is weighted by the gap measure between the two objects in the image. The gap measure we have chosen is an Euclidian distance as defined in [9]. Further developments could refine this measure.

5.1.1 Initial state generation. First, connected components of the image are extracted using a blob coloring procedure, little noisy connected components are filtered and then adjacency relations between connected components are determined using a method based on morphological operations applied on the document image (successive erosions of the connected component extern contours until contact points are found between components). Finally the adjacency graph is constructed by linking with an edge the pairs of adjacent connected components. The nodes represent the connected components of the document, and the edges the adjacency relations. Each edge between two nodes is weighted by the value of the gap between the two corresponding components in the image. This value is expressed as an Euclidian distance [9]. From this adjacency graph, the initial state is obtained just by selecting a candidate as text-line seed. Up to now this candidate is chosen randomly among all the connected components of the image. The choice of this candidate has no incidence on the solution found. The cost of the initial state is set to zero.

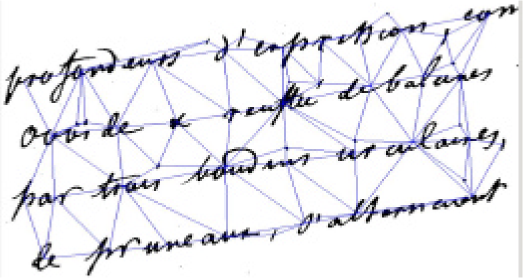

Fig. 2: adjacency graph

5.1.2 Goal state definition. We cannot describe the goal state explicitly, because we do not know it as it is what we are searching for. But we can describe it implicitly by defining some conditions a state has to verify to be considered has a goal state. Intuitively we can say that the segmentation is over, when the alignments are stabilized, that is, when none of the partial text lines can be extended further and no additionnal new line can be created. Formally, a goal state is reached when all the connected components are assigned to exactly one text line. In this case, all the nodes of the graph (database) correspond to exactly one text line. This condition constitutes the stop criterion for the search procedure.

5.2. Production rules

We consider the text line segmentation as a grouping process, which consists in iteratively grouping connected components into text lines. Each text line is built incrementaly. Given a text line, and a connected component, one has the possibility to group the connected component to the text line or not. The two production rules we consider are "merge", and "don’t merge”. When the rule "don’t merge" is applied, the rejected component becomes a seed for the construction of a new line. This starting new line is then considered as the current text line to develop first. To be able to apply a given rule, some pre-conditions have to be verified. For the two previous rules, the conditions are the following:

- the considered text line has to be the current one of the state
- the candidate component for the grouping has to be adjacent to the text line

These rules are applied only on adjacent connected components of the current text line, because there is no sense to group a component which is far. When there is no adjacent component to the current text line, no rule can be applied anymore, and another partial text line is choosen as the new current text line. Formally, using grammar notations, the two production rules we use can be expressed as follow:

- R_1: current line ? current line + connected component
- R_2: new line ? connected component

The operator + stands for "adjacent".

The application of these rules on a given state of the search tree allows to produce new children states, by modifying the global database.

5.2.1. Transition probabilities and rules costs. Each production rule is associated to a cost, which is the cost of the transition between a given current state and its child, by applying the rule. This cost is determined by computing the log-likelihood of the rule:

$$C_{R_i} = -\ln p_{R_i}$$

where p_{R_i} is the probability of the rule R_i. If several rules can be applied on the same entities, they must sum up to one:

$$\sum_i p_{R_i} = 1$$

Given an adjacent connected component of the current text line, we can apply both R_1 and R_2, so:

$$p_{R_1} + p_{R_2} = 1$$

$$p_{R_2} = 1 - p_{R_1}$$

p_{R_1} corresponds to the probability of the rule R_1 to be applied, that is, the probability of a connected component to be added to the current text line, derived using a proximity criterion between both entities:

$$p_{R_1} = e^{-\frac{d^2}{2s^2}}$$

where d is the euclidian distance between the adjacent connected component and the current text line, and s a normalization constant determined empirically on a collection of documents. Notice that this general formulation can be directly extended to a much larger number of features allowing the integration of contextual features.

5.2.2. Children states generation procedure. Given a current intermediate state, corresponding to a partial segmentation (partial partioning of the connected components), all the possible children states in the search tree are determined as follows: one applies all the valid rules, that is those that satisfy the above pre-conditions, on the database describing the current state, thus generating as many children as the number of applied rules. The children states are all the possible extensions of the partial partionning, obtained by applying one elementary operation (one production rule) on the current database. For example if there are four connected components adjacent to the current text line, one can apply both rules R_1 and R_2 on each of these four neighbors, thus generating eight children states. Figure 3 shows the principle of this procedure. We can see a line under construction (connected components in light gray, linked by edges). The other components connected by edges are those adjacent to the line which are candidates for the merging process. The rules are applied on each candidate, thus providing new states.

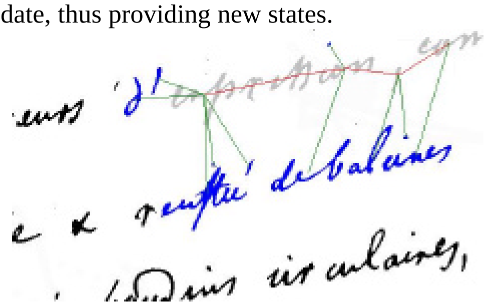

Fig. 3: generation of children states

For each child state, the cost of the path leading to this state is computed by summing the current cost of the path leading to the current state with the cost of the transition between the current state and its child, that is, the cost of the applied rule. Hence, the cost of a path in the search tree is obtained by summing the costs of the rules applied along this path.

5.3. Control system

The aim is to find the best partionning of the connected components into text lines. It consists in finding an optimal path in the search space between the initial state of the problem and a goal state. The initial state represents the unsegmented document (a set of connected components), and a goal state is a possible partionning of the connected components of the document into text lines (alignments). There are a lot of different possible configurations of components into text lines, but what we want to find is the best one. This optimal solution is obtained by applying on the global database a rule sequence which leads to a complete partionning (so that each component is affected to a line) and such that the global cost of the path is minimal among all the possible paths leading to a solution. This is a path finding problem. Several search strategies can be used to solve it. Among them we can cite: the Bristish Museum procedure, the Branch and Bound method, or A* algorithm [8].

The A* algorithm is one of the best search strategies because it leads to the optimal solution more quickly, thanks to the use of a heuristic to guide the search across the search space. The heuristic has to be admissible, i.e. an underestimate of the length of the remaining path to a goal. Determining a good admissible heuristic for the problem we consider is not trivial, so we have choosen, in a first time, to use no heuristic. In this case, the A* algorithm is equivalent to a Branch and Bound procedure. This search strategy consists in maintening a list of all the states generated so far, and not yet explored (these states are called "open" in the IA litterature), to which is associated the cost of the minimal path leading to each of them. At each step of the search, the state with minimal cost is extracted from the list, and then explored, that is we verify if this state is a goal according to the conditions described previously. If the conditions are verified, the search is over, and we are sure that the found goal is the best, according to Bellman's optimality criterion. If the current state is not a goal, it is then developped. It means that one generates all the possible children states of the search tree, using the development procedure described above. All the generated children states are then added to the "open" list. If one of the generated states is already in the list, the cost of the path leading to it is updated if smaller. This search procedure continues until a goal is found.

6. Implementation and first results

In a first time, we are mainly interested in extracting text lines, so we have integrated to this system only the two rules described previously. We have tested the method on few toy examples to verify the validity of the system. The method allows to find the best segmentation according to the defined quality measurement of the alignments, which is for the moment only based on proximity measure between connected components of a text line. However, it appears that this proximity criterion is not sufficient to define an alignment, if text lines overlap or if the interline distance is smaller than the intraline distance. We have shown that adding more contextual features is trivial according to our formulation and this certainly results in a general tool adaptable to various problems for layout analysis. We have not yet integrated more criteria, but it is clear that adding criteria such as direction continuity, similarity or contextual features will improve greatly the quality of the extracted text lines. For the moment, the interest of our method is limited because it allows to segment only well separated text lines, but we can notice that besides some classical methods, no assumption is made about the orientation of the text lines or about the reading order, allowing thus to extract multi-oriented lines as we can see on figure 4. Furthermore this approach is interesting because global quality of the segmentation is pursued, and not only local coherence.

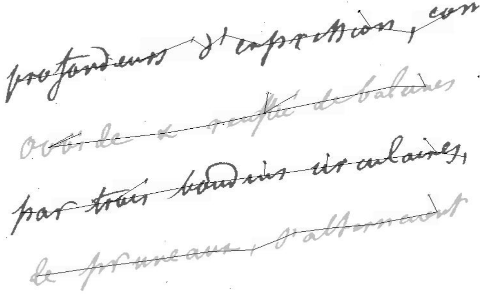

Fig. 4: segmentation result on skewed text lines

7. Conclusion and future works

In this paper we have presented a digitization project of authorial manuscripts, which aims to provide a hypertextual edition of a collection of the French novelist Gustave Flaubert drafts. We have focused on the difficulty to produce an accurate ASCII full text transcription respecting the layout of such complex handwritten documents, in order to facilitate the reading of the manuscripts and to allow an indexation of the manuscript fac-simile. We have underlined the fact that document analysis techniques should be helpful for human transcribers. However, classical methods for document image analysis are not robust enough to deal with this type of documents. So we have proposed a new approach for text line segmentation in handwritten documents. This approach is based on traditionnal IA problem solving framework using production systems. Although not mature enough to already provide good results on unconstrained handwritten documents, the proposed method allows nevertheless to avoid the errors caused by a local analysis, because the best solution according to the defined quality measure is pursued among all the possibilities of segmentation.

As evoked previously, future works will consist in adding new criteria for calculating the cost of the rules, such as direction continuity, similarity, and contextual features such as relative distance between the connected component to be added and the considered text line, according to the distance with the other neighbors. So we have to determine a feature vector for each rule of the production set. Defining the costs of the production rules in a multi-dimensional feature space will be possible thanks to a learning phase of the parameters associated to each rule on a learning database. In a second time, we will try to add new rules to the system, in order to extract not only text lines, but text blocks or other layout entities too.

8. Acknowledgement

Our work is sponsored by the "Conseil Régional de Haute-Normandie".

9. References

[1] Andre, J., Fekete, J.D., Richy, H., "Traitement mixte image/texte de documents anciens", Cahiers GuTenberg, No. 21, pp. 75-85, Juin 1995.

[2] Mao, S., Kanungo, T., "Empirical Performance Evaluation Methology and its Application to Page Segmentation Algorithms", IEEE trans. on PAMI, Vol. 23, No.3, pp 242-256, 2001.

[3] Likforman-Sulem, L., Faure, C., "Extracting text lines in handwritten documents by perceptual grouping", Advances in handwriting and drawing : a multidisciplinary approach, C. Faure, P. Keuss, G. Lorette and A. Winter Eds, pp. 117-135, Europia, Paris, 1994.

[4] AbuHaiba, I.S.I., Holt, M.J.J., Datta, S., "Line Extraction andExtraction and Stroke Ordering of Text Pages", Third International Conference on Document Analysis and Recognition, Vol. 1, pp. 390-393, August 1995.

[5] Likforman-Sulem, L., Hanimyan, A., Faure, C., "A Hough based algorithm for extracting text lines in handwritten documents", Third International Conference on Document Analysis and Recognition, Vol. 2, pp. 774-777, August 1995.

[6] Bruzzone, E., Coffetti, M.C, "An algorithm for extracting cursive text lines", Fifth International Conference on Document Analysis and Recognition, pp. 749-752, September 1999.

[7] Feldbach, M., Tönnies, K.D., "Line Detection and Segmentation in Historical Church Registers", Sixth International Conference on Document Analysis and Recognition ,Recognition, pp. 743-747, September, 2001.

[8] Nilsson, N.J., "Principles of Artificial Intelligence", Morgan Kaufmann, 1980.

[9] Seni, G. and Cohen, E., "External Word Segmentation of Off-line Handwritten Text Lines", Pattern Recognition, Vol. 27, No. 1, pp. 41-52, 1994.