Automatic indexing of French handwritten census registers for probate genealogy

Cédric Sibade, Thomas Retornaz, Thibauld Nion, Romain Lerallut
Christopher Kermorvant
A2iA, Artificial Intelligence and Image Analysis
40bis rue Fabert
75007, Paris - France
{cs,tr,tn,rl,ck}@a2ia.com

ABSTRACT

This paper describes the complete indexing process of the registers of a French census dating back to more than a hundred years, from image analysis to the integration into the information system, in the context of probate genealogy. The documents of interest are composed of a table of personal information in which the cells containing the first name, the surname and the relation to head of household must be extracted and recognized. More than 30 millions of cells were processed and their content either directly integrated into the information system or sent to keyers for manual validation, allowing an automation rate at 80% while keeping the error rate below 15% on average. Based on this project, we have started the development of a generic platform for table-based historical documents processing including new functionalities and a more generic and user-friendly table model definition interface.

Categories and Subject Descriptors

I.7.5 [**Document and text processing**]: Document Capture

General Terms

Experimentation

1. INTRODUCTION

The mission of probate genealogists is to locate legatees and to find missing heirs or beneficiaries in probate matters. In France, 2% of the 500,000 annual deaths leave no known legatees and this number has been increasing for the last 30 years due to the social mobility both geographically and in the family composition. When no legatee is known, the law stipulates that the relatives of the defunct must be searched up to the sixth degree. This search can take up to ten month and often needs to travel in several places to consul documents such as birth, marriage or death certificates. With the digitization and the indexing of these documents, the travels can be avoided and the search time can be reduced. However, the indexing of such historical documents is out of the scope of traditional OCR since they may be more than one hundred years old and often contain handwritten data.

Historical documents present many challenges for image processing and data recognition technique, in order to provide to the end-users an access to the information they contain. In addition to the degradation of the paper support that should be addressed, the writing style varies from period to period and from writer to writer. Each document may also have some specific aspects to take care of: abbreviation or specific jargon, slight modifications of the document model, empty field or table cell with specific interpretation, etc. Whatever the difficulties, end-users such as archivists and genealogists need to gain access to the document content by indexing its valuable information. Manual keying of such document is a tedious task and requires enrollment of numerous highly trained keyers. In order to limit errors, multiple keyers should work in a parallel way and therefore arbitration must be done to correct keyers discordance. In consequence, historical document manual indexation requires a large and costly human effort.

Concerning their content, historical documents can be of different types (historical letter pages, population census, company registers, mortgage books, etc.) and different periods of time. We have chosen to start working with tables as they represent a large volume of pages, with low variation in the data organization structure and are of primary interest for probate genealogists. In this paper, we present the work-flow we have set up for indexing a register of French census more than a hundred years old. Our existing recognition engines currently focuses on printed or handwritten *contemporary* documents in the form of plain letters, checks, money orders, forms or questionnaires. Therefore, our engine was adapted to table archive documents and historical writing style.

Section 2 presents previous research addressing table and archive document understanding and indexation. Then, Section 3 details the work-flow developed for the French census documents; an extract of the results is also presented. In Section 4, the evolution of this prototype is described; it focuses on the foreseen set of functionalities that is targeted for a generic platform for table-based historical document processing. Finally conclusions and future work are given in

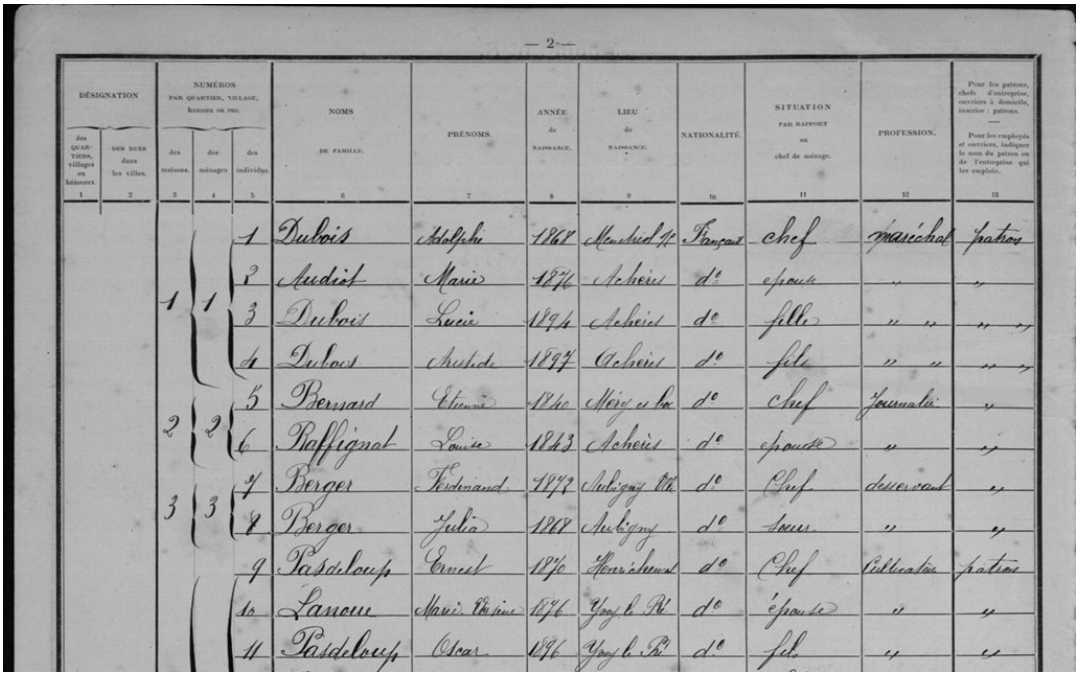


Figure 1: An extract of the French census, consisting of a table with 30 rows/inhabitants. For instance, the data to recognize on the first line are the *Dubois* surname, the *Adolphe* first name and *Chef* relation to head of household.

Section 5.

2. PREVIOUS WORK ON TABLE ARCHIVE RECOGNITION

The analysis of tables have been extensively studied since tables are a very common form of presenting data (see [6] for an *exercice de style* which aims at presenting a scientific paper using only tables). A variety of table recognition and analysis methods have been proposed for both printed and electronic documents, and from physical to logical and semantic analysis (see [14] for a comprehensive survey). Recognition of tables in historical documents has also been addressed, since a large amount of information is available in this form in historical documents (indexes, census tables, inventories) and the access to this information is easier than full text [1]. Different kinds of recognition techniques have been proposed to recognize tables in historical documents: techniques based on a formal language of document description [2, 8], techniques based on statistical machine learning algorithms [4], or bottom-up techniques, starting from low level features such as table lines to extract the structure [10]. Regarding the recognition of the information found in tables, the approaches based on word spotting [7] can not be used when all the data in the table must be recognized and indexed. Our approach is very similar to [10] except that we use a handwriting recognizer to recognize the information in the extracted table cells instead of using only human keyers.

3. FRENCH CENSUS PROCESSING

In this section, we describe a end-to-end project of automatic information extraction and recognition in French census historical documents.

3.1 Project presentation

The project consists in an automatic indexing of a French national census, dating back from more than 100 years ago. The census data consists of scanned census books, one per town, containing a table of handwritten data, one row per habitant. Figure 1 presents a sample page of one of the French departments, a French territorial division. Three of the columns must be recognized and indexed for the application: **surname** (*Nom de famille*) and **first name** (*Prénom*) for identity search and the **relationship of this person to the head of the family** (*Relation par rapport au chef de ménage*) to navigate through family relatives.

Aside these census pages, other additional pages are also scanned. They do not contain useful information so they need to be rejected. Figure 5 (a). is for instance a front page and Figure 5 (c). the household summary for the town.

Vocabularies of surnames and first names were constructed using data from other French census. These vocabularies are therefore available per department, allowing the access to area-specific demonyms and the access to occurrence probabilities for each surname or first name candidates (e.g. in France, *Martin* is more frequent than *Dubois*).

The goal of this recognition process is to fully automate genealogical queries, at a low error rate. Rejected pages as well as low confidence recognized words are validated by trained keyers. Then the full extraction data is integrated into a genealogical information system. The genealogist then makes a query to find heirs by their name, adding constraints on the location (department(s) or town(s)) and by family position; the query can be done with precise terms or approximate

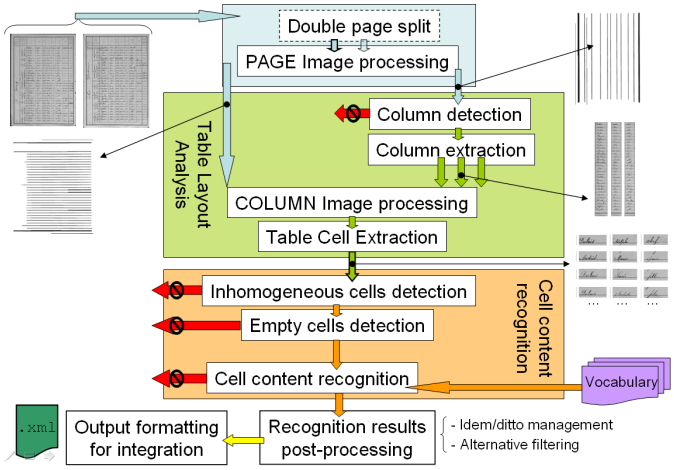
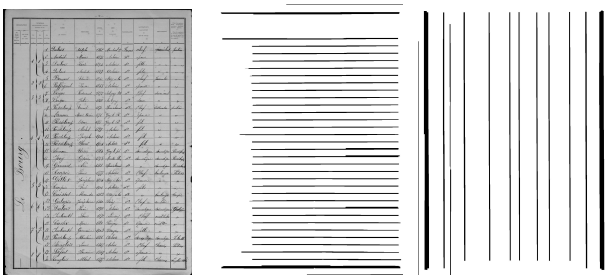


Figure 3: The complete processing work-flow.



(a) A single page (b) Lines extraction (c) Columns extraction

Figure 4: Lines and columns extraction on a single page containing a table.

spelling. All the matching results are displayed, including image snippet of the extracted row.

3.2 Processing workflow

A specific processing work-flow has been designed in order to register each table cell position. This registration process must be insensitive to variations in the scanned document: Figure 2 shows some examples of such variations: degradation of the document paper, irregular scan quality and variability of the content. Then each table cell snippet is recognized using a specific word recognizer trained on handwritten French dating from the beginning of the XX$^{\text{th}}$ century. Finally the indexation data is finalized, applying domain-specific post-processing on recognized words.

The next parts describe these three steps: first the table layout analysis, then the cell content recognition and finally the recognition results post-processing. Figure 3 summarizes the processing operations done at each step. Experimental results for one of the departments are presented in Section3.6.

3.3 Table layout analysis

Page pre-processing

The input page is scanned as a JPEG greyscale image. Generic image pre-processing is performed:

- if the page contains two tables, so if the page has a landscape orientation, the page is split in two portrait pages and each page is processed separately.
- the scale is modified in order to have standardized written or printed word heights; the target resolution is chosen to be between 200 and 300 dpi.
- a skew detection algorithm based on the binarized image page contour extraction estimates angles on the largest contours.
- the scan process may add some black borders that surround the document page. A processing applied on the binarized image page cleans these black borders by finding the abscissa (respectively the ordinate) where the number of white pixels on a column (resp. on a row) is lower than 0.95 of the black pixel count.

The result image is a skew-corrected greyscale image focusing on the scanned page.

Lines and columns detection

The next page processing step focuses on horizontal and vertical line extraction, as shown in Figure 4. A set of extensive morphological operations (cf. [11, 12]) with vertical and horizontal structuring elements is applied to the input greyscale image in order to retrieve two result binary images: one for the vertical lines and one for the horizontal lines. The morphological operations are chosen in order to clean small noises (typically small lines, written words) but also to enhance lines (enlarging them and reconnecting faded or discontinued segments).

The detailed algorithm for vertical lines detection is given below [1]:

- A mean filter along a sliding window is applied ($sz = 50$), we keep the absolute difference between image and filtered one. This step enhances vertical lines.
- Large erosions by segments in the vertical direction ($sz = 40$), and smaller erosions ($sz = 20$) on the cross direction are applied to reconnect lines
- Closing ($sz = 900$) in the direction of the lines removes all the lines that are not long enough
- Image is binarized using the well known Niblack[9] binarization ($\alpha = \frac{1}{3}$) applied on the whole image
- Component labeling is then used to remove noise and remaining small lines (remove all cc $<$ 0.3$*$width image size). We obtain Figure 4(c)

A similar process is applied to detect horizontal lines as shown on Figure 4(b).

[1] all given parameters are given in pixels and tuned for 200 dpi images, they should be scaled (up/down) accordingly for other resolutions

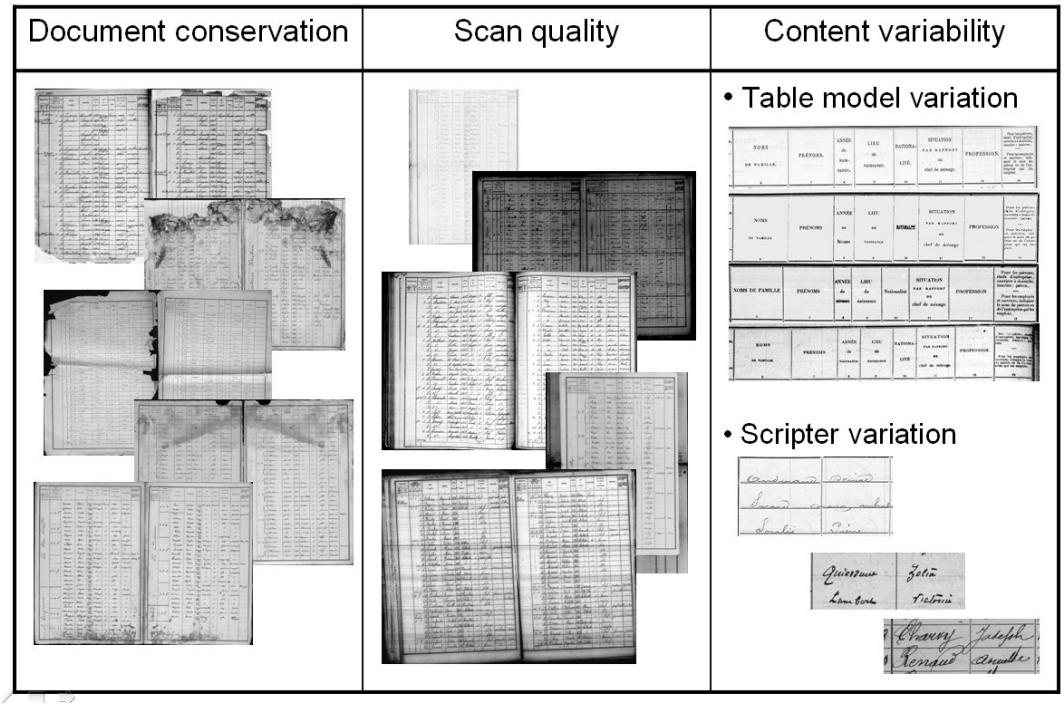


Figure 2: Examples of the variations of the input documents: degradation, scan quality and content.

A Hough transform (using techniques similar to [5]) is finally applied on these two images of lines, in order to get the lists of abscissae and ordinates of all the lines in the document. These lists may still contain some undesired line coordinates or miss some important ones, due to highly degraded document (low contrast, line-added by the scanner process, bold writing which is not filtered by the line detection process).

Validation of the type of page with a template matching technique

In order to filter unwanted lines, to recover missed ones and to reject documents that do not have lines (cover page) or lines at the wrong places (household summary page), the ordered abscissae of the vertical lines (resp. the ordered ordinates for the horizontal lines) are compared to a model list of abscissa (resp. a model list of ordinates). This sequential comparison algorithm searches for a series of coordinates that matches this template; one coordinate deletion or insertion is allowed per interval between two vertical lines for the undesired or missed lines. The Figure 5 illustrates the rejection of two out of scope documents as their vertical lines do not match the user-defined template, compared to the one of Figure 4(c). A similar interval matching is done on the rows, searching for horizontal lines regularly spaced out by a defined interval.

Finally cell snippets are extracted by cropping the pre-processed greyscale image on a box whose coordinates are given by the filtered line detection algorithm.

3.4 Cell content recognition

Cell preprocessing

Before applying word recognition, some basic preprocessing and quality classification operations have been applied:

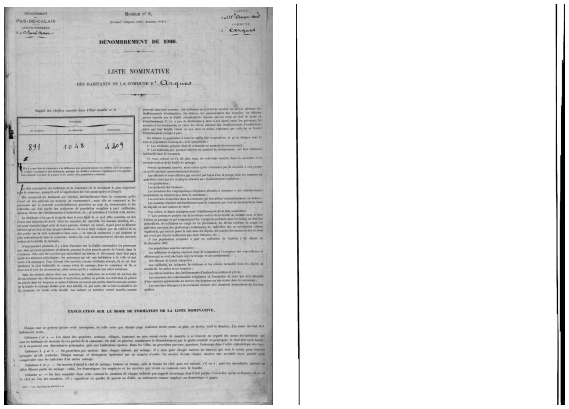
Inhomogeneous cell detection : detect very low quality image where the snippet contains paper tears (as in Figure 7(a)), smudges or water drops (cf. figure 7(c)); the goal is to detect large differences in the cell background color. The black text is removed by applying a large dilatation in vertical and horizontal directions and by classifying the background type using the mean cell color and its standard deviation as features.

Binarization : a binarization step is applied at the cell level, by thresholding using techniques similar to [13] (parameters: $\alpha = 3$, $box_{size} = 10$).

Noise removal : basic image cleaning (e.g. close tiny gap using directional closing, remove salt and pepper noise).

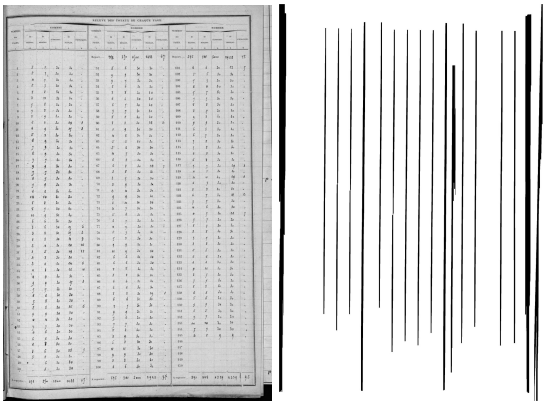
Empty cell detection : if a cell is seen as empty, it can either have a meaning for the document (e.g. equivalent to *idem*, so *repeat previous line data*) or it has no particular meaning which would need to be translated for the user as *empty* and interpreted upon context (is it an error? does it mean end of the document? is it a separator for households splitting the table in multiple group of rows?). To detect them, we simply work on the binarized image, center it on written text or on the lower table line cell and measure the height of the resulting centered snippet. The Figure 6 depicts this quick decision protocol.

Line removal : removal of bounding line for each table cell.



(a) A front page

(b) Columns are not matching



(c) A table page, summarizing household count

(d) Columns are not matching

Figure 5: The type of page is detected with a template containing the expected position of the columns.

The resulting image is a bilevel 200 dpi, noise and line free, de-slanted snippet. The cells that have been rejected as having a heterogeneous background will be validated by the keyers or rejected if nothing can be read (as seen in Figure 7(a) and Figure 7(c)).

Cell content recognition

The content of each cell is recognized by a isolated word recognizer based on a hybrid HMM (Hidden Markov model) using a grapheme-based feature extraction. First, each word is decomposed into a sequence of graphemes (which are a letter or a part of letter), then a neural network (MLP) is used to compute the posterior of each grapheme class and these posteriors are integrated into a word model HMM (composed of the letter HMM models) to compute a recognition score for each word in the lexicon (see [3] for details). The recognizer was trained on a database of French handwriting from the beginning of the XX$^{\text{th}}$ century. The lexicon for the surnames and first names were available for each department. Typical dictionary sizes per department are 30,000 to 120,000 for surnames and 3,000 to 15,000 for first names. As these dictionaries have been built from previously indexed French census, the occurrence probability of each entry can be computed and used in the recognition process to promote more frequent alternatives. The result of this recognition phase is:

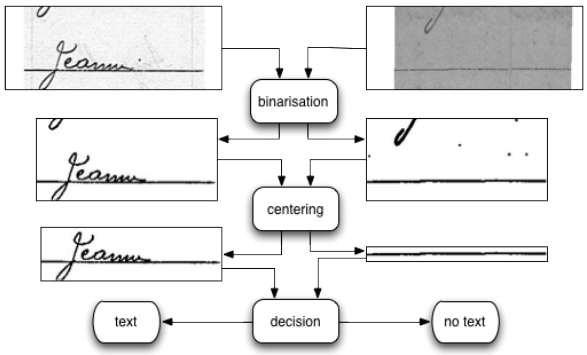


Figure 6: Illustration of the empty cells detection algorithm.

- the cell position within the page
- a list of 3 best alternatives, with their recognition score
- additional information regarding the inhomogeneous and empty cell detection

3.5 Recognition results post-processing

This last step goal is to apply specific rules and to consolidate the results at page level:

Recognition result filtering : we apply some client-specific context rules on empty fields or field with the words *idem*, *ditto* or their abbreviations (*id*, *do*):

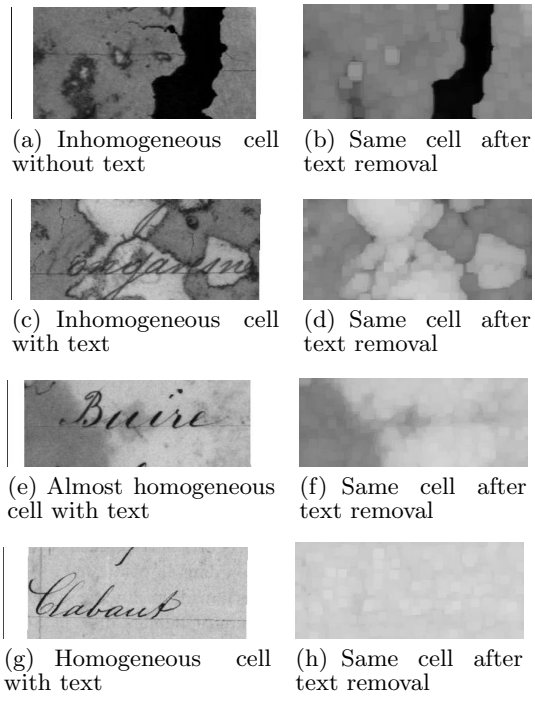
- the surname and relation to head of household columns can have such 'idem' field, but not the first name column. In this case, an error is emitted for the first names;
- too many empty cells is considered as an error;
- a whole row with empty cells (for surname, first name and relation to head of household column) is considered as an error.

Page level score computation : we compute a simple score for the page, as the average of all the surname best score. This gives as a rough quality indicator for the page. A page rejection can be applied if this score is under a defined limit, by resetting to 0 all cells score, keeping recognition alternatives but forcing validation by human keyers.

Data encoding : we encode the resulting data into a specified file format (A2iA XML format).

3.6 Recognition results

The recogniton results are presented in Table 1. The true values were available for three different departments and one prefecture town and used for evaluation. The department $D1$ is the set containing the cleanest and simplest images



(a) Inhomogeneous cell without text
(b) Same cell after text removal
(c) Inhomogeneous cell with text
(d) Same cell after text removal
(e) Almost homogeneous cell with text
(f) Same cell after text removal
(g) Homogeneous cell with text
(h) Same cell after text removal

Figure 7: Detection of inhomogeneous cells with text removing.

whereas department $D3$ presents many of the quality defects that could be seen in the whole censuses.

The recognition results were evaluated using two combined methods:

- either with the best recognition result or by looking if the true data is one of the three alternatives.
- using an exact letter to letter matching between the true value and the recognition value or allowing one character difference (the string edit distance must be lower or equal to 1).

The target was to maximize the automation rate with a limited error rate. An automation rate of 80% was chosen, meaning that only 20% of the documents were sent to keyers. The error rate (substitution rate) was computed at word level only on the documents processed automatically since the error rate of the keyers is not known. The error rates at 1 character and with 3 alternatives range from 5.5% for the cleanest data set to 21.0% for the most difficult set. On average, the error rate is below 15% which might seem high but is satisfactory given the difficulty of the task.

3.7 Data integration

The complete workflow, including the keying of the documents rejected by the recognition process is the following:

1. The paper document is scanned. First name and surname dictionaries are created for each geographic region using already indexed data.
2. The geometrical model of the table is defined if it is different from the generic one. The recognizers are configured and the dictionaries are prepared (duplicates and typos are filtered and occurrence frequencies are estimated).
3. The set of images with the defined table model is processed in batch, producing XML files.
4. After a quick quality verification, a score thresholds is estimated for each column to achieve a minimum automation rate of 80%.
5. The rejected pages and cells are keyed manually, using trained human agents. Each rejected cell is located in the whole page along with three recognition alternatives. The keyer can choose between these recognition suggestions or enter his own proposition.
6. Both automated and keyed data are integrated into the information system. It is now ready to be queried by the genealogists.

With this worklow, we have processed 27 different French departments or prefecture towns, representing more than 340,000 pages with 30 rows and three columns per page to process, so more than 30 millions cells.

4. TOWARDS A PLATFORM FOR HISTORICAL DOCUMENT PROCESSING

Based from the French census indexation project described above, we have started to develop a generic platform for archive documents and in particular for documents containing tables. The main goal of the future *ArchiveReader* platform is to provide a clear user interface or API:

- to configure the table template that will be matched against the images and used to extract the correct columns, and within them their rows, eventually reaching the cells of interest.
- to define which recognition engine to apply on each group of homogeneous content cells. The recognition operation is defined in terms of content (digits, alphanumerical code, word(s) belonging to a pre-defined vocabulary, single letter, cross or tick signs, etc) and writing style (with models adapted specific countries and specific periods of time).

A typical work-flow for the indexation of a table document is:

1. Scan the archive documents;
2. Define groups of similar documents in order to define one processing per group. The next steps are applied to each group;
3. Define which columns, rows and cells are regions of interest (ROIs); describe their content and define constraints to help the indexation;
4. Assign pages and/or ROI(s) to keyer(s);

Test set name	Content	Test set size	Vocabulary size		Number of alternatives	Char. edit distance	**Automation=80%**	
			Surname	First name			Surname Error rate	First name Error rate
$D1$	Well preserved and clean pages	2800	31230	3040	1 3	0 1	13.9 5.5	14.6 8.8
$D2$	Faded pages, difficult scripters, paper quality defects	9200	27350	2450	1 3	0 1	19.9 11.1	27.0 21.0
$D3$	Pages with water drops and scratch defects	8300	123280	12300	1 3	0 1	25.2 17.8	25.4 15.2
$P1$	Difficult scripters, tiny & slanted writings	6000	27070	1630	1 3	0 1	21.2 17.0	23.5 17.6

Table 1: Recognition results for 3 departments ($D1$, $D2$ and $D3$ and a prefecture $P1$) for surname and first name: the word level error rate is given at 80% of automation rate for four test sets with different quality levels. The evaluation is made with 1 or 3 recognition alternatives allowing one character substitution (edit distance equal to 1) or not.

5. Validate and/or cross-check with indexed data;
6. Publish indexation results.

We have developed an automatic table recognition tool which shares the same preparation phases (scan, classify then select and configure ROIs). Using this tool in the work-flow allows the recognition engine to either replace the keying step or to work in parallel with human keyers.

A screen-shot of the prototype, called TableModeler, is shown in Figure 8. The sub-window on the left side gives a tree representation of the table model used for the data extraction and recognition.

4.1 Improvements on table layout analysis

In order to target a table included into a text page or in case of bad page placement during the scan process, the registration of the table position has been improved. The user can select printed keywords around the table and link them to the table's bounding box; after the page pre-processing steps, the position of the table is located by finding one or more of these keywords.

An additional functionality has been added to the lines and columns detection phase, to address tables where delimiting lines have either disappeared or were never present. In this case, the goal is to locate writings in cell that are separated by white spaces. By inverting the image (having white writings on black background) and by applying an adapted plain line detection algorithm as in section 3.3, these virtual separation lines can be extracted.

The definition of the table template also benefits from a novel approach: the user selects the column of interest by drawing a bounding box on the first cell of each column and by giving the number of cells per column. By doing that, the column width is known, as well as the cell height. The table template matching algorithm can be finally done as with the French census prototype, by looking at the columns defined by the first cells, having equally spaced rows.

4.2 Adding new recognition functionalities

Concerning the configuration of the recognition engine, the user can now customize:

- The recognizer's model, by choosing the country or the writing time period. The following models are currently available: French and British from the beginning of the XX$^{\text{th}}$ century and contemporary French, British and American. New models will be available in the future, including American writing style from the beginning of the XX$^{\text{th}}$ century.
- The cell geometry, by giving the bounding box of the first cell of the column and the type of line delimiters (either a plain or a 'virtual' line).
- The cell type
  - for word recognition using a vocabulary list, the user can choose the dictionary files, specify if it contains occurrence probability and the minimum and maximum number of words to find.
  - for recognition of isolated characters, e.g. one letter for the person sex, the parameters are the type of character (numerical only, letter only or alphanumerical), their number and an optional regular expression constraint.
  - for digits column, e.g. age or year, the content constraint can be a range or a list of accepted values.

5. CONCLUSIONS

In this paper, we have described a end-to-end workflow for the indexation of historical documents containing personal information stored in tables. First names, surname and relation to head of household information, representing more than 30 millions cells, were extracted, automatically recognized or manually validated, and integrated into an information system used for probate genealogy. Based on this success, the development of a more generic platform for table-based historical document processing was started, in order to ease and accelerate the processing of similar documents in the future.

6. ACKNOWLEDGMENTS

We would like to thank the Coutot-Roehrig company and especially G. Postansque and M. Bouamama for their constructive collaboration, which helps in a first time to achieve the French census objectives and in a second step to define the basis of the future ArchiveReader platform.

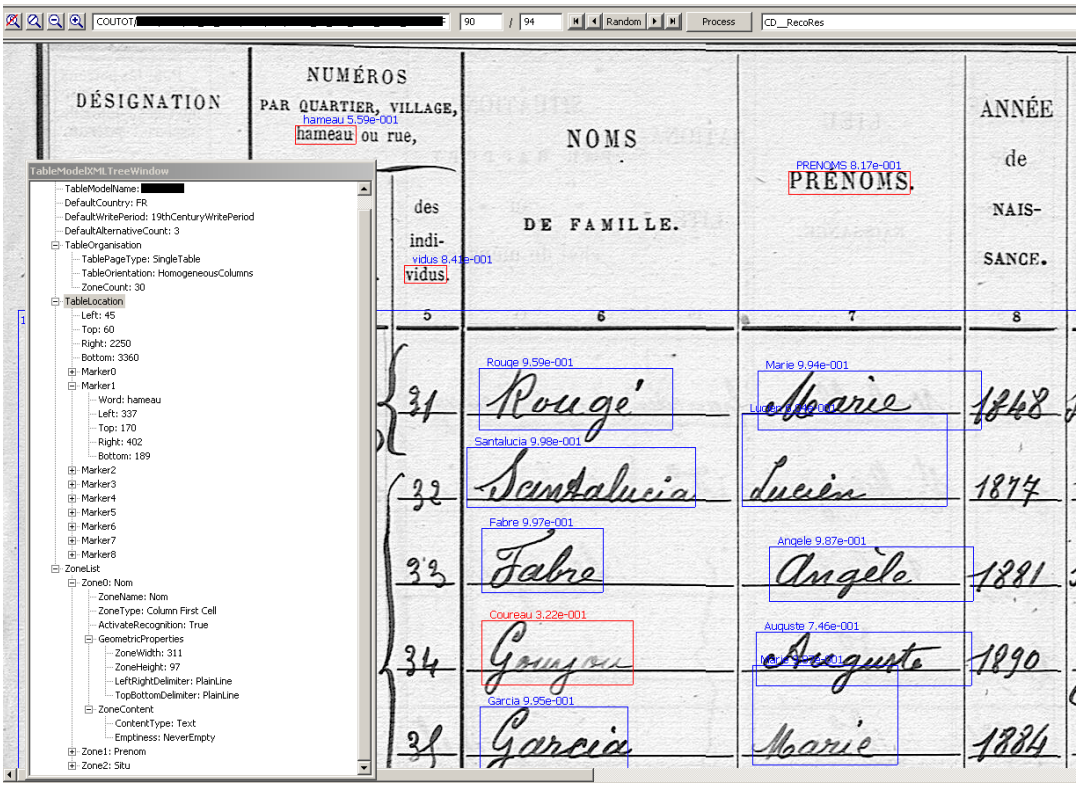


Figure 8: Screenshot of the TableModeler tool on the French Census. Box are drawn on the recognized surname and first names, with the best score.

7. REFERENCES

[1] M. Bulacu, R. Van Koert, L. Schomaker, and T. Van Der Zant. Layout Analysis of Handwritten Historical Documents for Searching the Archive of the Cabinet of the Dutch Queen. In *Proc. of the Int. Conf. on Document Analysis and Recognition*, volume 1, pages 357–361, 2007.

[2] B. Coüasnon. DMOS, a generic document recognition method: application to table structure analysis in a general and in a specific way. *International Journal on Document Analysis and Recognition*, 8(2-3):111–122, Mar. 2006.

[3] N. Gorski, V. Anisimov, E. Augustin, O. Baret, and S. Maximov. Industrial bank check processing: the A2iA CheckReader. *International Journal on Document Analysis and Recognition*, pages 196–206, 2001.

[4] K. Laven, S. Leishman, and S. Roweis. A statistical learning approach to document image analysis. In *Proc. of the Int. Conf. on Document Analysis and Recognition*, ICDAR '05, pages 357–361, 2005.

[5] L. Likforman-Sulem, A. Hanimyan, and C. Faure. A Hough based algorithm for extracting text lines in handwritten documents. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 2:774–777, 1995.

[6] D. Lopresti and G. Nagy. A tabular survey of automated table processing. In *International Workshop on Graphics Recognition*, volume 1941, page 93. Springer, 2000.

[7] R. Manmatha and T. M. Rath. Indexing of Handwritten Historical Documents - Recent Progress. In *Proc. of the Symposium on Document Image Understanding Technology*, pages 77–85, 2003.

[8] I. Martinat, B. Coüasnon, and J. Camillerapp. *An Adaptative Recognition System Using a Table Description Language for Hierarchical Table Structures in Archival Documents*, volume 5046 of *Lecture Notes in Computer Science*, pages 9–20. Apr. 2008.

[9] W. Niblack. An Introduction to Digital Image Processing. *Englewood Cliffs, N.J.: Prentice Hall,*, pages 115–116, 1986.

[10] H. Nielson and W. Barrett. Consensus-based table form recognition of low-quality historical documents. *International Journal on Document Analysis and Recognition*, 8(2-3):183–200, Feb. 2006.

[11] J. Serra. *Image Analysis and Mathematical Morphology.* Academic Press, Inc., Orlando, FL, USA, 1983.

[12] P. Soille. *Morphological Image Analysis: Principles and Applications, 2 edition.* 2003.

[13] C. Wolf and J.-M. Jolion. Extraction and recognition of artificial text in multimedia documents. *Pattern Anal. Appl.*, 6(4):309–326, 2004.

[14] R. Zanibbi, D. Blostein, and J. R. Cordy. A survey of table recognition: Models, Obervations Transformations, and Infrences. *International Journal on Document Analysis and Recognition*, 7(1):1–16, 2004.