A Strategy for Retrospective Conversion of Documents

Pierre Héroux, Eric Trupin, Yves Lecourtier

A Strategy for Retrospective Conversion of Documents

Pierre Héroux, Éric Trupin and Yves Lecourtier
Laboratoire Perception, Systèmes et Information
UFR des Sciences et Techniques
Université de Rouen
F-76821 Mont-Saint-Aignan Cedex
Pierre.Heroux@univ-rouen.fr

Abstract

This paper proposes a strategy for retrospective conversion of documents. This strategy consists in an interpretation cycle where document analysis and document understanding interact. This cycle is initialized by the extraction of the outline of the layout and logical structures of the document. Then, each iteration of the cycle consists in the detection of inconsistencies in the document modeling. Treatment are applied to correct these inconsistencies. The cycle ends when no more inconsistency occurs. The method used for document objects classification is based on the fusion of statistical and structural classifier results. A structural comparison method is used for document structure classification. The different levels of inconsistency which may occur in the document representation are detailed. Several treatments to solve theses inconsistencies are presented. Finally, future prospects concerning incremental learning of new classes of documents are developped in order to implant this strategy in a document processing system.

keywords: document structure, retrospective conversion, solving inconsistencies

1 Introduction

This paper proposes a strategy for a retrospective conversion of documents. Retrospective conversion consists in rebuilding the modeling of a document from the document image. Retrospective conversion is useful because it allows to benefit of advantages of electronic documents such as re-edition, diffusion, indexing and archiving.

A document can be described by its layout structure and its logical structure [3]. The layout structure hierarchically models the visual aspect of documents. It is obtained by extracting and classifying graphical objects of the document image. These graphical objects are called layout objects. The logical structure represents the document organisation on the basis of the meaning of the content. The logical structure describes the way a document can be parted into title, sections, subsections, paragraphs...

The extraction of the layout structure is called document analysis. Document understanding consists in constructing the corresponding logical structure.

The strategy presented in this paper can be applied to a wide range of documents which can be grouped into classes. A document class contains documents sharing a part of the layout and logical structures. The part of the structure shared by all the documents of a class is called the generic structure. Objects can also be grouped into classes. An object class is described by a generic object which expresses the features shared by all specific objects of the class.

Our strategy is based on a dynamic interaction between document analysis and document understanding. It consists in a so called interpretation cycle inspired by [5] which is initialized by extracting prim-

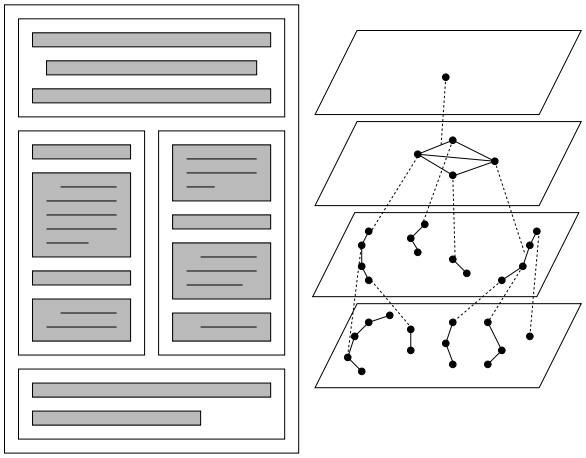

Figure 1: Layout structure of a document

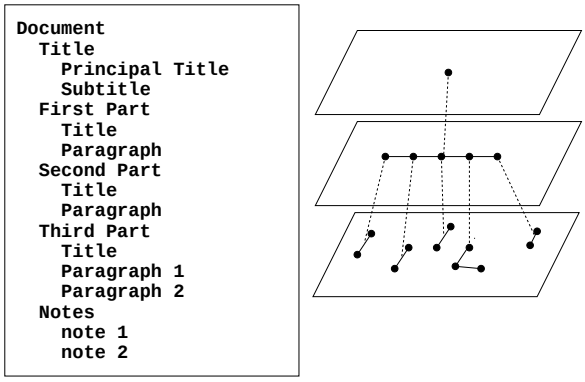

Figure 2: Logical structure of a document

itive layout and logical structures. The cycle continues by locating inconsistencies in the document modeling and proposing different solutions to solve these inconsistencies. This cycle ends when no more inconsistency is detected.

Key elements of this strategy are the way objects are extracted, how they are classified, the detection of inconsistencies in the global document representation and the way these inconsistencies are solved. Section 2 deals with the cycle initialisation which consists in the extraction of the layout structure and the construction of a primitive logical structure. In section 3, object and structure classification methods are detailed. Section 4 presents the different kinds of inconsistencies which may occur in the document representation. In section 5, differents methods are proposed to solve inconsistencies. Finally, in section 6, some conclusions and prospects are reported.

2 Structure extraction

A complete retrospective conversion of documents has to construct a document modeling which represents, at least, the layout structure and the logical structure of the document. Our strategy is based on a cycle which makes document analysis and document understanding interact. The cycle is initialised by a phase which provides primitive versions of layout and logical structures.

The layout structure is obtained by extracting graphical objects from the document image [1]. This is performed by a segmentation algorithm applied on the document image after low level processing (deskewing and binarisation). Extracted objects are then associated in composite layout objects according to size and proximity criteria. Then, they are labelled (text, graphic, image...) according to graphic criteria (size, black pixel density...). New composite objects are then constructed with adjacent objects identically labelled. Finally, a first version of the layout structure is obtained.

A structural classification algorithm which compares the specific structure to be identified with structures representing document classes gives a first hypothesis concerning the document class. Assuming that a document class contains not only a generic layout structure but also a generic logical structure, the outline of the logical structure is constructed by instanciating the generic logical structure corresponding to this hypothesis. This instanciation is performed by associating a logical equivalent to basic layout objects.

This initialises the interpretation cycle.

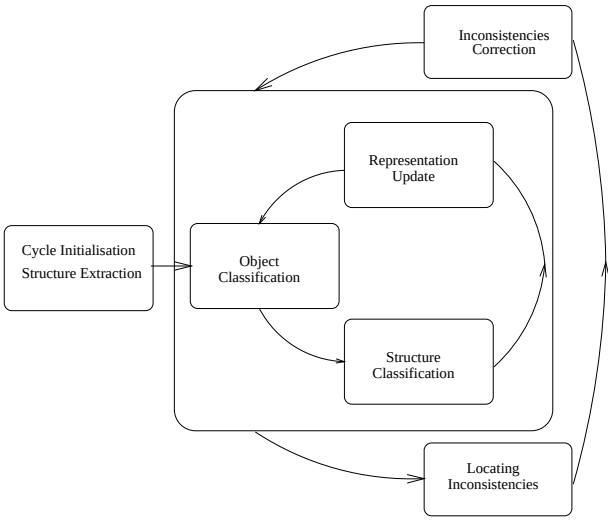

Figure 3: Interpretation cycle

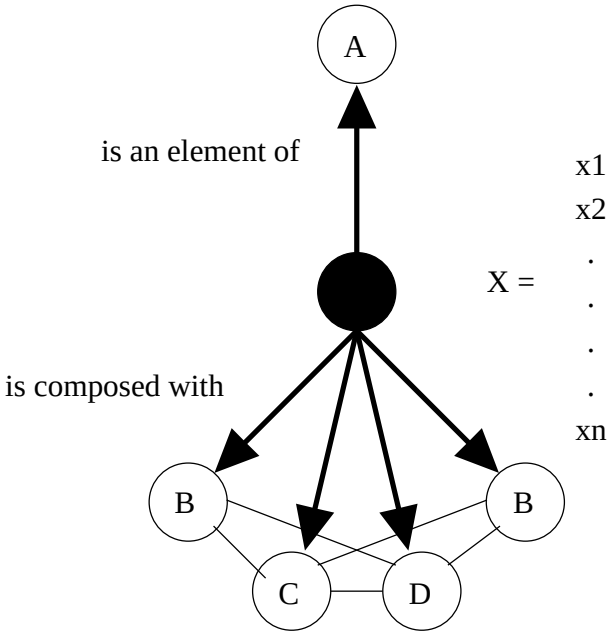

Figure 4: Structure of an object

3 Object and structure classification

In this section, we describe the different methods used for object and structure classification. Object classification is used to determine the class of a specific object. It is performed, for layout objects as well as logical objects, by comparing the specific object to be identified with generic objects representing object classes. Each specific object contains three kinds of features : a numeric feature vector, a graph and a list of plausible labels. The numeric feature vector contains numeric information concerning the object independently from the rest of the document. The graph represents the organisation and relations between the constituents of the object (except for basic objects). The list of plausible labels is generated when the hierarchically superior object is labelled. Then, hypothesis are launched concerning the labels of its constituents.

A statistical classification method computes distances (Hamming distance, euclidean distance, maximum distance...) between the feature vector of the specific object and feature vectors of generic objects. This distance expresses the proximity between the object to be identified and the classes in the representation space. Finally the statistical classifiation method returns a weighted list of plausible labels. The weight associated to each label is inversely proportional to the computed distance.

A structural classification method establishes another weighted list of plausible labels. This list is weighted by the inverse of the distance between the graph representing the organisation of the specific objects constituents and the graphs of generic objects. A graph of a generic object represents the organisation of consituents common to all associated specific objects. The graph distance is computed by first determining the number of nodes of the subgraph common to the compared graph [4]. Two overlapping rates t_1 and t_2 are computed by dividing the number of nodes of the common subgraph with the number of nodes of each of the compared graphs. If one of these rates is 1, this means that one of the graph is included in the other one. In this case, if the other rate is very small, then the included graph is very small in regard to the other one. If the compared

graphs equal, t_1 and t_2 are equal to 1. A similarity measurement can be established as

$$d = \frac{1}{t_1.t_2} - 1.$$

Finally, a fusion of the three weighted lists is performed and provides a unique weighted list of plausible labels for the object to be identified. Different methods can be used to perform the fusion of the lists. It can use the position of each class in the lists, a weighted sum between the lists, or multiplying the weights considering them as probabilities...

The first step of the interpretation cycle consists in initialising the layout and the logical structures. When no object has been classified, the classification method only takes into account object intrinsic features (feature vector).

In the following steps, the object classification method takes into account not only intrinsic features of the object, but it also refers the hierarchical context of the object by considerind the labels of the object constituents and the propositions concerning its class emitted by the classification of the hierarchically superior object. This has the effect of making the representation converge on stability.

An hypothesis is proposed for the label of each object. This hypothesis corresponds to the first element of the weighted list returned by the classification method.

When an hypothesis has been provided for each object, structure classification is used to determine the class of the document. The structure classification is performed, for layout structure as well as logical structure, by comparing specific structures representing the document to be identified with generic structure representing document classes. The classification of each of these structures proposes a class for the document. The structural classification method exposed in [2] is used.

4 Locating inconsistencies

After object and structure classification, the next step in the interpretation cycle is the detection of inconsistencies in the description of the document. Inconsistencies can be separated into several categories or levels. The solving of an inconsistency will depend on its level.

First, we define what we call intrinsic inconsistency. It refers to the fact that no generic object contains the features observed for the specific object. The object can not be associated to any of the known object classes. On the contrary, an object is said intrinsically consistent if its features are plausible in regard to the known object classes.

The next consistency level is called contextual neighboring consistency. An object is said to be consistent at the neighboring contextual level if there is at least one generic object which includes this object and its neighbors as its constituents in the observed configuration.

The hierarchical consistency deals with the fact that an object associated to a specific class can, or not, be a constituent of an object of an other class. An object is said to be hierarchically consistent if its class is compatible with the class of the hierarchically superior object.

Finally, we define the abstraction level consistency. It deals with the compatibilty between the class of a logical object and the class of the corresponding layout object. This mapping between layout and logical object is not always possible. A logical object not always correspond to a single layout object, for instance, a paragraph can be split into two text blocks on two columns. However the abstraction level consistency can always be evaluated for structures. The results of layout structure and logical structure classification must correspond to the same document class.

Object consistency is tested for each object and struture.

5 Solving inconsistencies

Once inconstencies in the document representation has been detected, they have to be solved. Several treatments can be applied to solve inconsistencies.

First, object or structure classification may have returned an erroneous result. A low cost treatment consists in testing consistency, not any more with the first element of the weighted list of plausible labels,

but with classes in the following propositions.

To make the classification more reliable, new elements can be taken into account to propose a new class for an object. For instance, the knowledge of the class of neighbour objects can efficiently guide the classification. This contextual knowledge is integrated by the structural classification. On the other hand, determining the class of an object gives an information which can be used as a priori to determine the class of its constituents. This is introduced by the weighted list of plausible labels. However, it can only be taken into account at the iteration of the interpretation cycle following the classification of the hierarchically superior object.

Inconsistencies may result from other errors. The construction of the structure can be the reason of an inconsistency. Some objects have to be merged, others have to be split. The processing used in the structure extraction may have been unappropriate. In this case, they have to be called into question. The structure extraction can be repeated for a small part of the document by using methods which are more appropriate. At each iteration in the interpretation cycle, we have a better knowledge of the document and of the context of unidentified objects. This contextual knowledge is used to apply more appropriate methods. In section 6, prospects are proposed to dynamically qualify the relevance of methods.

Finally, in spite of the use of the previous solutions, inconsistencies may remain in the description of the document. This can lead us, not to call into question the data, but the knowledge which might be insufficient. If a specific object belongs to an unknown class, it can not be identified. Then, the class of this object has to be learned. Incremental learning of object classes is refered later as a futur prospect.

The principal problem is to choose the treatment which will solve a given inconsistency. Inconstencies are caused by a classification errors, an erroneous extraction due to unappropriate methods or an unsufficient knowledge. The different treatments to solve inconsistencies are not equivalent in computational complexity. A new object classification has a lower cost than performing a structure extraction (followed by object classification). The learning of a new class would have an even greater cost. Our strategy consists in first trying the lower cost treatments.

The treament applied to a single inconsistency can lead to call into question a great part of the document representation. The tried treament is given up when the total number of inconsistencies has not decreased after few interpretation cycle. Then treament with a greater computational cost is tried.

6 Conclusion

This paper proposes a strategy for retrospective conversion of documents. This is based on the interpretation cycle which consists in classifying each object analysing the consistency of the description and solve the inconsistencies. This cycle makes document analysis and document understanding dynamically interact. On one hand, the logical structure is initialised from the knowledge of the layout structure. On the other hand, the layout structure is not fixed and inconsistencies in the logical structure can lead to call into question the layout structure.

The document representation describes three different contextual relations between objects (neighboring relations, hierarchical relations, layout-logical relations). These differents levels of relation are exploited by the classification methods.

This proposed strategy will be implanted in a document processing system which would be able to process a wide range of documents and provide a convenient representation.

Futurs prospects should deal with the characterisation of extraction methods. This would permit to choose the method which is the more appropriate to the processed data. A method should be evaluated for a particular application domain. This evaluation could consist in determining whether the results of the method have lead to inconsistency or not each time they have been applied. This also allows to determine the efficiency domain of the method.

The need for incremental learning of new classes has been expressed. This could be done by comparing the unidentified configuration with the knowledge. Exploiting similarities and differences between an observed situation and the knowledge could allow to build new object classes.

An operator could validate, correct the representation or complete the parts of the representation left unidentified. His intervention through an graphical user interface could be taken into account for incremental learning.

References

[1] S. Diana, E. Trupin, F. Jouzel, J. Labiche, and Y. Lecoutier. From acquisition to modelisation of a form base to retrieve information. In *Fouth International Conference on Document Analysis and Recognition*. IAPR, 1997.

[2] P. Héroux, S. Diana, É. Trupin, and Y. Lecourtier. A structural classifier to automatically identify form classes. *Advances in Pattern Recognition, Lecture Notes in Computer Science*, 1451:429–439, 1998.

[3] International Standard Organization. *ISO 8613 : Information Processing. Text and Office System, Office Document Architecture (ODA) and Interchange Format*, 1989.

[4] L. Miclet. *Méthodes structurelles pour la reconnaissances de formes*. Eyrolles, 1984.

[5] J.-M. Ogier, R. Mullot, J. Labiche, and Y. Lecourtier. Interprétation de document par cycles perceptifs de construction d'objets cohérents. application aux données cadastrales. *Traitement du Signal*, 12(6):627–637, 1995.