Abstract

This paper is concerned with the presentation of a declarative knowledge base, the Conceptual Model, which describes the invoice domain as generally as possible. Such a model is based on a semantic network that is able to describe the invoice domain by different levels of abstraction. The Conceptual Model can be used for the labelling procedure of physical rectangles, extracted from invoices, in order to construct a model (Document Model) for each class of invoices. The Document Model contains physical coordinates for each rectangle, which can be extracted from an invoice, and the related semantic label. Once the Document Model is constructed, it can be applied to understand an invoice instance, whose class is univocally identified by its logo.

Keywords: Conceptual Model; Document Model; Document understanding; Semantic network.

1 Introduction

The problem of document understanding has held a growing importance in the last few years. Many firms have to handle a large amount of documents every day. One way to face the problem is to acquire the information by converting each document into an electronic format, but the information that they contain is difficultly retrieved in a selective fashion. Hence the need to create a database, extracting data from the documents. When the amount of data to acquire is considerably large, the automatic data entry in the database results particularly desirable; hence the effort to build up systems which are able to understand as wide domain of documents as possible.

The first generation of works has concerned with the design of systems oriented to understand a single type of documents with a fixed layout, by using a model which provides logical relationships among fields [1] or physical field coordinates [2], [3], [4]; they usually exhibit good performances in field locating, but they lack flexibility.

The problem of system flexibility is much less studied and it is concerned with the task of multi-classes of document images understanding. The systems aimed at flexibility usually use a knowledge-base oriented approach to document interpretation, which is used directly in the reading task without a predefined goal of reading [5], [6], [7]. Such a characteristic provides these systems with a considerable degree of flexibility, but it produces a probabilistic semantic labelling of physical objects, with loss of reliability.

For example [5] proposes an open system architecture for document image understanding. The system performs an automatic document classification among a limited variety of classes, and reads the document by using a document independent knowledge. Such a knowledge contains world features and specific ones for each class of document.

These two kinds of knowledge are defined and developed independently. The knowledge about a specific domain is considered as a part of the input, while the system is provided only with a set of problem-solving strategies and image processing tools.

A quite analogous approach, applied to the domain of business letters, is described in [7].

In [6], as well, is presented a similar system for physical object labelling, starting from a knowledge base formalized by Horn clauses. The correct label is assigned only by taking into account spatial relationships with other objects of the document, whose labels are already known. All these systems deal with a probabilistic labelling of physical objects.

Some systems have been proposed in order to come to a compromise between the flexibility of open systems architecture and the considerable performances of systems oriented to the fixed layout understanding. These systems [8], [9], [10] express a middle policy of development among the foregoing ones. They aim at constructing, automatically or semi-automatically, a model for each class of documents, which contains object coordinates and their semantics. Moreover they provide a procedure of class recognition.

In [8], [9] a knowledge-based system for the identi-
fication of regions in documents with different layouts is proposed. The system is based on a modular knowledge representation structure, called Geometric Tree, whose intermediate nodes represent classes of logical and physical structures which are refined in the lower levels. Leaves of Geometric Tree represent the logical and physical model of a specific class of layouts for the universe of documents under consideration. During the on-line understanding phase the system attempts to match the physical layout of the current document with the structures contained in the Geometric Tree, executing a depth traversal. Understanding a given document amounts to finding a path in the Geometric Tree from the root to one of its leaves. When the system is not able to classify a given document, it provides a knowledge acquisition component which allows the modification and extension of the knowledge base, interacting with the user who completes the layout description.

Similarly in [10] is presented a method to recognize the layout structures of multi-kinds of table-form document images, and to construct a model for each type of table-form. The model is organized in a Structured Description Tree, which describes a logical class of table-forms, and in a Classification Tree, which contains different physical models that can be described by the same logical one. This system as well purposes to recognize a physical structure and to associate it with its logical representation. In the event that the physical model of a table-form is not contained in the knowledge base, the system constructs the new model for the current layout, acquiring its physical and logical features, and adds it monotonically in the database.

This paper describes a part of a project concerned with the design of a system which is capable of understanding documents belonging to invoice domain.

Invoices present a large variety of layouts, whose classes are usually identified by their logo. In order to realize a flexible system without loss of performances to understand each type of documents of invoice domain, a system which deals with the construction of a model for each class of invoice has been designed. Such a system is based on a knowledge base, for the invoice domain, that attempts to capture some structural and logical similarities by describing logical relationships among document parts and their physical constraints.

In this paper the invoice understanding system and a possible model (Conceptual Model) which describes such general knowledge about invoice domain are presented. The Conceptual Model is used by the invoice understanding system in order to construct [11], for each class of invoice, a specific model (Document Model) that is used during the phase of document reading (on-line understanding phase). The description of the Conceptual Model for the invoice domain is based on a semantic network [12] which provides a declarative and procedural knowledge representation system.

The overall structure of the system for invoice document processing is illustrated in Section 2. In Section 3 the Conceptual Model developed for the description of invoice domain is described. Finally, in Section 4 some conclusions are reported.

2 The invoice domain

Invoices represent a universe of documents that may be divided into classes, each of them in a one-to-one correspondence with its logo. Even if invoices don't have a fixed layout, the instances of a class are characterized by the invariance of logical and physical structure. The invoices belonging to a certain class consist of parts, which we call "objects"; each of them has a logical label and contains a particular type of data. In most cases objects are surrounded by segments to form open or closed rectangles. The objects in an invoice are: a logo which identifies the class of an invoice; upper, middle and lower section which contain, respectively, data related to clients, products and totals; items that are rectangular regions which contain data and sometimes the corresponding keywords (see Fig. 1).

Classes of invoices present some structural or logical similarities, which can be captured by a general knowledge, that describes logical relationships among document objects and their physical constraints.

A general knowledge on domains characterized by logical relationships variance among objects is used in [5] during the on-line understanding phase for probabilistic semantic attributions to the objects. Such an approach could be used in the invoice domain, but it would lead to a probabilistic object labelling. On the contrary, we want to design a system which has a deterministic knowledge on the position of an object.

In order to design a robust system to understand each invoice class and to guarantee a high degree of flexibility, we present a two level knowledge approach to invoice understanding. It is based on specific invoice class models for the on-line understanding phase and a general knowledge, which describes the invoice universe, used for the construction of specific models.

2.1 Invoice modelling

As we have discussed, a class of invoices is characterized by a fixed logical and physical structure. This
structure is represented by a specific model that we call Document Model, which describes all the objects appearing in the invoices of the class.

A Document Model of a document $D(n)$ is a set of $n$ 5-tuple $d_i = \{l_i, x_{i\text{min}}, y_{i\text{min}}, x_{i\text{max}}, y_{i\text{max}}\}$, $0 \leq i \leq n - 1$, where $l_i$ is the semantic label of $d_i$ and $(x_{i\text{min}}, y_{i\text{min}}, x_{i\text{max}}, y_{i\text{max}})$ are the physical coordinates which locate univocally $d_i$. We call $d_i$ an object of the document $D(n)$.

Such a model may be absolute or relative. In the former case, each object has absolute coordinates with respect to the scanner coordinate system. In the latter, each object has relative coordinates with respect to an object $d_k$ of $D(n)$ considered as reference. We have chosen a Document Model with relative coordinates because it allows the system to be tolerant to skew variation in the on-line understanding phase.

An example of invoice and related Document Model is reported in Fig. 1.

Figure 1: Example of invoice and related Document Model. In such a model the object "description" and the related coordinates $(x_{\text{min}}, x_{\text{max}}, y_{\text{min}}, y_{\text{max}})$ are reported.

A Document Model for each class of invoices is constructed by means of a higher level knowledge; we call it Conceptual Model. The Conceptual Model we define for our domain is a declarative knowledge base, described by frames of a semantic network. The model is described in Section 3. This kind of knowledge ensures flexibility to the system in the invoice domain, in the spirit of capturing some features common to the various classes, and it is used as a base for the Document Model construction for each class of invoices. Each Document Model constructed is mapped to the logo of the class and is used in the on-line understanding phase [11].

2.2 The invoice understanding system

After acquiring an invoice instance by a scanner, and converting it into an electronic format, the system attempts to classify its logo [13], [14], [15]. We have assumed the hypothesis, widely satisfied, of univocal mapping between logo and invoice structure. As a consequence, the successful classification of a logo instance means having recognized the invoice class which the current invoice instance belongs to. In fact, the successful classification of logo means the presence, in a database of models, of the Document Model which describes the class of the instance. On the contrary, if logo is not recognized, the system cannot map the invoice instance to its model. In this case a new Document Model is built.

Depending on the successful or unsuccessful logo classification, the system provides two different alternatives.

1) On-line understanding phase of the current invoice instance. In such a case, the invoice can be read directly by using a proper Document Model. The first step is a deskewing procedure to align the coordinate system of the model to that of the current instance. Afterwards, under Document Model supervision, the system can locate each object with its coordinates, related to the barycentre of logo and to a particular segment that divide two of the three sections each invoice consists of.

2) Construction of a Document Model for the unknown invoice class and subsequent on-line document understanding, as described in 1). In such a case, segments and rectangles are extracted and the above mentioned three sections are located [11]. Finally, a procedure in order to assign a semantic label to each rectangle is executed [11] [16]. Labelling is driven by a Conceptual Model representing a general knowledge about invoice domain. The result is a Document Model for the class of invoices which the current instance belongs to. Such a model can now be used for instance understanding.

The logical scheme of the whole system is reported in Fig. 2.

3 A Conceptual Model based on a semantic network

Invoice domain is characterized by a wide variety of physical and logical fixed structure classes, each of them mapped to its logo. Nevertheless, it is possible to provide the system with a general knowledge for the whole domain, oriented to capture some similarities among classes of invoices. We call it Conceptual Model. The Conceptual Model describes invoice domain by a semantic network, inspired on [12]; in particular, in this paper, we describe a model resulting from the analysis of about thirty different classes of invoices.
3.1 The semantic network

The semantic network we describe allows us to represent an a priori knowledge on the invoice domain by nodes and directed labelled edges, called links. Each node represents a concept that is described by a set of characteristics, some of them are further specified by particular substructures. Edges are used to express binary relations between the nodes. Moreover such network allows to describe the invoice domain by different levels of abstraction: each level focuses on particular logical or physical characteristics; a specific link to pass from a level to another one is provided.

Our network is built by nodes, links and substructures. We use one type of node: concept; three types of links: concrete, specialization and part; and three types of substructures: attribute description, link description, function description.

The link concrete establishes relationships between different levels of abstraction and a hierarchy among them.

The links specialization and part represent, respectively, two different hierarchies, that define partial non-reflexive orders on the set of nodes; they are:

- Generalization: type of hierarchy that relates a concept to more generic ones. Such a link is called specialization of and its inverse specialization; in other frameworks is often called is_a;
- Aggregation: type of hierarchy that connects a concept with other concepts which describe its parts. It is called part and its inverse part of. In this context, aggregation models a physical part hierarchy [17].

The nodes of the semantic network are described by frames. Each concept frame contains the following slots:

- "Name of concept";
- "Frequency", which indicates the observed frequency of the object modelled by such a concept;
- "Attribute", which is described by a particular substructure, "attribute description", in terms of "Role", "Type of values", "Number of values", "Restrictions" and "Computation of values". The slot "Computation of values" contains a pointer to the substructure "function description", which describes the function computing the value of the related attribute; its specification slots are "Name" and "Arguments";
- "Specialization", which contains a list of pointers to other nodes that specialize the related concept;
- "Part" and "Concrete", which are described by the substructure "link description". This substructure describes the role of a part (or of a concrete), by the slot "Role". In general, the values of a part (or of a concrete) may be defined by one or more concepts; such concepts are attached to the slot "Goal Node". Finally, the slot "Frequency" provides the observed frequency of such a part (or concrete);
- "Structural Relation", which contains a pointer to a function that provides the probabilities for the states of the variable representing all the possible contextual relations for a concept at the named object level. Such a function simply contains and returns a vector of the probabilities for each state of the contextual relation the related concept may assume. The possible states are related to the join of the variables the current concept instantiation depends on. Their probabilities derive from the joint probability of the variables.
- "Judgement", which is described by the substructure "function description" whose slots are "Name" and "Arguments". The function, attached to such a substructure, provides a credibility order among the selected rectangles for the labelling phase [11], [16]. The judgement function of the current concept to be instantiated assigns, to each extracted candidate rectangle, the probability of the state it assumes.

3.2 The Conceptual Model

The semantic network, introduced in Section 3.1, is used to describe the Conceptual Model of the invoice domain (see Fig. 6). The Conceptual Model describes the invoice domain at four different levels of abstraction: image level - primitive level - geometry level - named object level. The "image level", in practice, refers to the pixel level; the "primitive level" is concerned with the treatment of segments; the "geometry level" describes the physical rectangles composed by segments; the "named object level" is related to labelled objects. In this paper we deal with the "named object level" and the "geometry" one, which are described in the following sections.
3.2.1 Invoice modelling at the named object abstraction level

At the named object abstraction level, the concept INVOICE is the root. Each INVOICE can be divided into three sections: MID.SECT, where products and their codes, prizes and related taxes are described; UP.SECT, where data related to clients, their codes and conveyance modalities are reported; LOW.SECT, where totals are summarized. Each of these sections is in a part-of relationship with INVOICE concept (see Fig. 6). A frame description for MID.SECT is presented in Fig 3.

![Figure 3: A frame description of the concept MID.SECT](image)

Furthermore, we model each section as an aggregation of concepts that represent the specific items appearing in an invoice, as description, total, quantity and so on. An example of a frame of the concept TOTAL in a part-of relationship with MID.SECT is reported in Fig. 4. Note that the judgement function gives a measure of certainty that the extracted candidate rectangles be an instance of TOTAL. Such a function considers the rectangles of middle section that satisfy to particular physical constraints (the related frame is described in Section 3.2.2 and in Fig.5), and their position with respect to the rectangle already labelled as an instance of DESCRIPTION.

![Figure 4: The frame for the concept TOTAL](image)

It can be observed that the item quantity may satisfy different physical constraints, and has different adjacency relationships with other items depending on the class of the invoice; its concrete concepts are represented by IL_Cat.IT.RC or III_Cat.IT.RC at the geometry level. The related probabilities of representing the concept QUANTITY are given by the slot “Frequency” of the substructure that describes its concrete concepts. Moreover it can appear on the right or on the left of the item description. These two alternatives are given by the states of the variable representing all the possible contextual relations the concept QUANTITY may satisfy; the related probabilities are given by the function attached to the slot “Structural Relation”.

3.2.2 Invoice modelling at the geometry abstraction level

At the geometry level a generic invoice can be modelled by a collection of rectangles whose generic features are reported in the frame which describes the concept RECTANGLE.

Depending on geometric features, a generic rectangle can specialize in section rectangles and item rectangles. Section rectangles divide the invoice into three parts and item rectangles contain data and keywords. Section rectangles are modelled by the concept SECT.RECT, the corresponding frame describes its features. Similarly item rectangles are modelled by the concept ITEM.RECT.

In its turn SECT.RECT specializes in three differ-
ent concepts: \textit{SECT RECT U}, \textit{SECT RECT M} and \textit{SECT RECT L}, which are described by three frames. \textit{SECT RECT U}, \textit{M} and \textit{L} represent the concretes respectively of \textit{UP SECT}, \textit{MID SECT}, and \textit{LOW SECT} described at the named object abstraction level (Fig. 6).

Parts of \textit{SECT RECT U} are \textit{LOGO BOX}, which models the area that contains the invoice logo, and \textit{ITEM RECT U}, that is the concept modelling the item rectangles of the upper section. The \textit{LOGO BOX} at the geometry level represents the concrete of the concept \textit{LOGO} at the named object abstraction level and corresponds to the concept \textit{IMAGE} at the image abstraction level. Similarly, \textit{ITEM RECT U} represents the concrete of the concepts that are part of \textit{UP SECT}. Furthermore, \textit{ITEM RECT U} is a specialization of \textit{ITEM RECT}.

\textit{ITEM RECT M} and \textit{ITEM RECT L} are part of \textit{SECT RECT M} and \textit{SECT RECT L}, respectively. Both \textit{ITEM RECT M} and \textit{ITEM RECT L} are specializations of \textit{ITEM RECT}.

Examining the physical structure of the middle section, it can be observed that width is a clear discriminant feature for some items; they can be clustered in three categories with respect to their width, with a constraint on the first category cluster that is to contain only the widest of them. This is a common feature of all invoices, for which the item description is always the widest one. For this reason \textit{ITEM RECT M} specializes into \textit{I.Cat.I.T.RC M}, \textit{II.Cat.I.T.RC M} and \textit{III.Cat.I.T.RC M}. The frame describing \textit{II.Cat.I.T.RC M} is reported in Fig. 5.

In general, such a clustering makes the instances of \textit{ITEM RECT M} semantic labelling phase easier, because it leads to a preliminary selection of the possible rectangles which may represent the item the system is searching for.

Moreover I, II and III.Cat.I.T.RC M are the concrete concepts for the concepts of the named object level that model the invoice \textit{items}.

On the whole the semantic network that describes the Conceptual Model for the invoice domain is illustrated in Fig. 6.

4 Conclusions

Invoices are a universe of documents divided into classes. Each class can be described by a Document Model which contains physical coordinates of the \textit{items}, represented by physical rectangles, and their semantics. Such a model can be used to understand an invoice instance, whose class is recognized univocally by its logo. The classes of invoices present some structural or logical similarities, which can be captured by a general knowledge, that we call Conceptual Model. It describes logical relationships among \textit{items} and their physical constraints. This paper presents a Conceptual Model based on a semantic network that is able to describe the invoice domain by different levels of abstraction.

The Conceptual Model can be used to drive a labelling procedure of physical rectangles, extracted from an invoice instance of an unknown class. The result is a Document Model for each class of invoices, and, globally, a monotonic set of knowledge, oriented to invoice understanding. The labelling procedure is not treated here, because it is beyond the aims of this paper [11], [16].

The Conceptual Model presented in this paper is part of an Invoice Understanding System, that we are developing at the DSI (Dipartimento di Sistemi e Informatica, University of Florence). The system is under implementation on a Sun Sparc system.
Figure 6: The semantic network that describes invoice Conceptual Model.
References


