Exploring Semantic Roles of Web Interface Components

Kang Zhang

Department of Computer Science
The University of Texas at Dallas
kzhang@utdallas.edu

Jun Kong

Department of Computer Science
North Dakota State University
jun.kong@ndsu.edu

Abstract—The adaptability of Web interfaces in response to changes in the interaction context, display environments (e.g., mobile screens) and user’s personal preferences is becoming increasingly desirable due to the pervasive use of Web information. One of the major challenges in Web interface adaptation is to discover the semantic structure underlying a Web interface. This paper presents a robust and formal approach to recovering interface semantics using a graph grammar approach. Due to its distinct characteristics of spatial specification in the abstract syntax, the *Spatial Graph Grammar (SGG)* is used to perform semantic grouping and interpretation of segmented screen objects. We use the well-established image processing technology to recognize atomic interface objects in an interface image. The output is a spatial graph, which records significant spatial relations among recognized objects. Based on the spatial graph, the SGG parser recovers the hierarchical relations among interface objects and thus provides semantic interpretation suitable for adaptation.

Keywords—Interface adaptation; graph grammar; Web semantics; layout;

I. INTRODUCTION

The rapidly growing Web provides a convenient and promising approach to accessing information, even in a 3D fashion [Cos02]. The availability of wireless network and mobile devices makes it possible to access information from anywhere at anytime. In such a pervasive environment, the adaptability of Web interfaces in response to changes in the interaction context, display environments (e.g., mobile screens) and user’s personal preferences is highly desirable. One major challenge in Web interface adaptation is to interpret the Web interface semantics, i.e. page

segmentation, that groups semantically related interface objects in a hierarchical structure and accordingly tag the semantic role of each object [Yu03].

Web pages have been created independently by different designers and developers around the world. This implies that various styles of look-and-feel and different implementation techniques may have been used. It is therefore challenging to devise with a generic approach for extracting interface semantics. Many researchers have explored heuristic approaches [Ahm08, Bal06, Bur05, Buy01a, Buy01b, Che03, Che05, Gu02, Gup03, Mil02, Yan00, Yan01, Yu03] to discovering information organization underlying a Web page. These heuristic approaches assume that Web designers follow some common guidelines to design and present Web pages. For example, closely related contents may be placed in proximity and the HTML tag *div* could be used to organize information. The guidelines can be distilled to be heuristic rules, used to derive information organization from the perspectives of visual layout and DOM structure (if available).

Heuristic approaches, however, cannot handle the irregular HTML usages and styles satisfactorily. For example, tables in HTML are designed to organize and display tabular data, implying that information in a table is closely related. However, by not displaying the table border, many Web developers use a table as an organization grid to layout pictures and texts. In this case, information enclosed in a table may not necessarily be semantically relevant. Ahmadi and Kong [Ahm08] have evaluated six heuristic rules on three genres of Web sites (e.g. news, travel and shopping), and concluded that heuristic rules have different accuracies on different genres. Furthermore, heuristic approaches can group closely related information, but they are not capable of tagging semantic roles. Most heuristic

Figure 1. Web Interface Interpretation

rules are defined on a DOM structure. In fact, even a simple Web page has a complex DOM structure. The complexity of DOM structure could significantly affect the performance of page segmentation.

Different from the traditional heuristic approaches, a visual language-based approach performs Website-dependent page segmentation based on graph grammars [Kon08]. Instead of requiring a common pattern of design and presentation across different Web sites, this method only assumes a consistent pattern applied to Web pages within a Web site. Based on this assumption, Web patterns are formalized as a graph grammar, which formally and hierarchically specifies the information organization in a Web page. Specifically, according to different application domains, the graph grammar can be customized to derive a fine-grained segmentation, such as recognizing the title, publishing time and content in an article. This approach formalizes and analyzes HTML Web pages based on the available DOM structures. However, the DOM-based specification and analysis require identifying atomic interface objects, such as sentences or images. For example, in an HTML Web page, a sentence (i.e. an atomic interface object) may be separated by several HTML tags, which make it hard to identify the start and end of the sentence. Consolidating information pieces into an atomic interface object is a critical step in page segmentation. The information consolidation can significantly simplify analyzed Web pages, and thus increase the performance of page segmentation.

This paper proposes a novel approach to page segmentation, taking advantage of graph grammars to provide robust page segmentation without relying on the DOM specification. Various grammar formalisms [Bra95, Bot04, Cor97, Cos97, Cos00, Fis98, Kon06, Rek97, Sch95, Sch99, Zha01] have been developed for different purposes.

Due to the unique spatial specification capability in the abstract syntax, the Spatial Graph Grammar (SGG) [Kon06] is used in our approach for analyzing Web interfaces. Spatial specifications in the abstract syntax enable designers to model interface semantics with various visual effects (e.g., a topological relation between two interface objects). On the other hand, our approach analyzes a Web page, or any interface page, directly from its image, instead of DOM structures. An image processing technique [Bar09] could divide an interface image into different regions and recognize atomic interface objects, such as texts, buttons and etc, in each region. Based on the imaging technique, a spatial graph, in which nodes represent recognized atomic objects and edges indicate some significant spatial relationships (such as touch and containment), is generated. The SGG parser then parses the spatial graph to discover the hierarchical relations among those interface objects. Our approach has the following advantages.

- **Efficient.** Visual analysis on an interface image can efficiently trim the original Web page by consolidating information pieces into atomic interface objects. Dividing a Web page into several regions supports an incremental parsing, which narrows the parsing scope and thus reduces search space with an improved performance. Furthermore, our approach supports analyzing dynamic contents and Web interfaces, where DOM specifications are not available.
- **Robust.** When interpreting graphical interfaces, non-uniformity, i.e. diversified styles and layouts independently developed by different designers, imposes a technical challenge. We observe that user interfaces created from the same organization or in the same Web site follow a uniform design so that they are homogeneously generated based on some common styles. A domain-specific

analysis can avoid inconsistency and design conflicts, and thus improve robustness.

To our knowledge, the above approach is the first to combine image processing with graph grammar techniques, which effectively addresses the issue of non-uniformity and simplifies page segmentation.

II. APPROACH OVERVIEW

Figure 1 shows an overview of our approach. Web interface interpretation is performed in two main steps: (1) recognize interface objects in an interface image and (2) interpret the interface through graph grammars.

In the first step, advanced image processing technology is used to divide an interface into several regions and recognize atomic interface objects (such as a button or a check box) in each region. HTML tags may separate an atomic interface object into several pieces. For example, in order to highlight several words, a sentence, i.e., an atomic interface object, may be separated into several pieces by HTML tag `<b>`. Since the separation is completely content-dependent, it is challenging to derive general rules to consolidate those pieces into atomic interface objects. Distinct from previous approaches, this paper proposes an effective approach that recognizes atomic interface objects directly from an interface image, instead of DOM structures. Image analysis techniques, such as those of Barkol *et al.* [Bar09], are used to identify the interface layout and objects in the image.

Consolidating information pieces as atomic interface objects can significantly simplify the original Web interface and facilitate page segmentation. Furthermore, such a visual analysis on the interface image can divide an interface into several regions. Those regions provide natural boundaries between composite interface objects. In other words, we can interpret interface objects incrementally – first interpret interface objects within one region and then consider the relations between different regions. Such an incremental analysis is inherently consistent with the hierarchical design of Web interfaces. Designers in general first divide an interface into several layout regions, and then determine the organization and layout of interface objects within each region. For example, a Web interface can be divided into several regions, such as top, bottom, main region and etc [Ahm08]. In addition, the incremental analysis could handle irregularity. If a region contains style exceptions, not captured by a graph grammar, we can simply interpret those exceptional interface objects as the direct children of the node representing the corresponding region. Furthermore,

limiting the spatial parsing in a small region can reduce the search space and improve performance.

Next, visual analysis on the interface image results in the generation of a spatial graph. In a spatial graph, nodes represent recognized interface objects, and edges indicate spatial relations, each indicating a close semantic relationship. We have evaluated different Web sites, and discovered that four spatial relations strongly imply a close semantic relation between two objects, i.e., touching, containment, vertical and horizontal relations within a small distance. Accordingly, each edge in a spatial graph represents one of those four relations. Meanwhile, a spatial graph also records the coordinates of each recognized interface object. Therefore, other spatial relations can be derived from those coordinates. Compared with the source HTML file, the spatial graph facilitates page segmentation by (1) consolidating information pieces together and (2) removing all redundant contents (such as empty columns or tables, which are used for adjusting layout).

In the second step, a graph grammar is applied to the automatically generated spatial graph for analyzing the hierarchical structure. In order to provide high usability, interface designers in general follow previous successful experiences, which can be summarized as guidelines, to organize and present interface objects. For example, semantically related contents are placed in proximity. Those guidelines provide valuable hints to recover the interface semantics underlying an interface. Our approach uses a graph grammar to formalize guidelines.

A graph grammar offers a computational paradigm of mathematical precision and visual specification [Var02]. It defines computation in a multi-dimensional fashion based on a set of rewriting rules, i.e., *productions*. Each production consists of two parts: a left graph and a right graph, the difference of which visually indicates the changes caused by a computation. Since interface objects are organized in a hierarchical structure, it is natural to formalize such a structure using a graph grammar. More specifically, the left graph in a production may include a composite interface object, which is made of a set of atomic/composite interface objects in the right graph. Each production groups related objects locally and a complete graph grammar provides a systematic specification to glue low level groups into a higher level group. Based on the graph grammar, a parser constructs a hierarchical parsing tree, in which a leaf node indicates an atomic interface object and an intermediate node represents a composite object, bottom up as a coherent interpretation for a Web interface.

This paper uses the spatial graph grammar [Kon06] to analyze Web interfaces: the distinct capability of spatial specification in the abstract syntax provides designers the flexibility to define various types of spatial relations; and the context-sensitivity increases the expressiveness. The incremental parsing in our approach supports reusing a portion of a graph grammar in different Web sites. Even though two Web pages from different Web sites may be different at a high level, some low-level patterns, such as the organization of a paragraph, may be used repetitively across different Web sites. Therefore, the corresponding productions can be reused in different Web sites.

In summary, our approach uses the image segmentation technology to recognize atomic interface objects and applies the spatial graph grammar to specify patterns underlying Web pages. Based on the grammar, a graph parser takes an interface image (including recognized atomic interface objects with their spatial properties) as input and produces a hierarchical structure of interface organization.

III. AN EXAMPLE

Having introduced the theoretical foundation of our approach, this section uses an example to illustrate our approach.

Web designers often follow a pattern to organize and present information when creating a Web page. Such a pattern keeps a consistent layout across the whole Web site. Consider the example of Northwest Airlines (NWA) Web page that includes several topics, such as flight search and hotels. Those topics are organized and displayed with the same pattern: each topic has a title and several lines of texts. A search interface is displayed on the right of the central region.

The image analysis process recognizes layout segments (frames) and GUI objects: their location on the page, with associated type, text value and other attributes. The structure of the original Web page is constructed as a graph as shown in Figure 2(a), using the graph grammar parsing and induction tool called VEGGIE [Ate07]. Each topic is displayed within a rectangle, recognized as a *Default_Value* object, and includes several lines of texts, each being recognized as a *text* object.

Based on the generated graph, we formalize the pattern underlying Web pages as a graph grammar. More specifically, the left graph in a production includes one composite interface objects (other nodes in the left graph are context objects), and the right graph contains several atomic/composite interface objects, which hold required spatial relations. In the right graph, the spatial relations of

containment, touching and vertical/horizontal relation within a short distance are abstracted as

(a). The structure of the original Web page (b). An interpretation

Figure 2. Interpreting Web pages

edges while other spatial relations are defined through spatial specification (in a production. Each production defines a local composition of interface objects. In other words, the composite interface object in the left graph is made up of the atomic/composite interface objects in the right graph.

Figure 3 presents the productions used to derive a topic on the left side of the central region. There are four productions, each having a left graph and a right graph. Each left or right graph has one or more nodes connected via embedded small rectangles called *vertices*. A vertex may be prefixed with an integer number unique in the production. The number is used to mark the contextual information in the graph, a technique inherited from the Reserved Graph Grammar formalism [Zha97, Zha01]. This technique of marking the contextual information allows context-sensitive graph grammars to be parsed correctly and efficiently.

Production P1 in Figure 3 combines two adjacent lines of texts as a composite interface object *texts*. The node *default_value* in P1 serves as a context object. Production P2 is useful when a topic includes odd number of lines of texts. Production P3 further combines two composite interface objects *texts* as a larger composite object. Finally, Production P4 abstracts all lines of texts as a composite interface object *topic*. Based on the graph grammar in Figure 3, a parse tree is generated as shown in Figure 2(b) that gives the interpretation for the Web page in Figure 2(a).

IV. CONCLUSION

Given the increasing types of browsing interfaces, such as a mobile screen, it is desirable to adapt a Web interface for different display requirements. Such requirements may include

Figure 3. Productions used to analyze topics

various interface constraints and end user’s personal preferences. One critical challenging issue in interface adaptation is to discover the semantic structure underlying a Web interface. Different from existing heuristic approaches, this paper develops an approach to page segmentation based on image analysis and graph grammar. Instead of analyzing the document source and DOM structure, our approach uses the traditional image segmentation technology to recognize atomic interface objects and divide an interface into several regions. The image segmentation result is a spatial graph, in which nodes represent recognized interface objects and edges model spatial relations, implying a close semantic relation between two objects. Based on the spatial graph, spatial parsing is performed to extract the semantic of the corresponding Web page. Due to the distinct capability of spatial specification in the abstract syntax and expressiveness with context-sensitivity, the *Spatial Graph Grammar (SGG)* is chosen as the definition formalism for page segmentation.

Our approach is in fact general enough to be applicable to interpreting general graphical user-interfaces or any digital displays. The ability of automatic page segmentation on graphical or Web interfaces sets a solid foundation for many useful applications. For example, semantically grouped Web page components could be arranged for small-form displays such as cell phone screens. Another example application is in understanding and testing software that generates graphical user-interfaces by extracting the semantic components and their relationships from the interfaces.

As the future work, we will conduct more experiments on real-world examples to investigate issues like scalability and efficiency. Especially, we will investigate how to improve the efficiency of designing a graph grammar. Though the reuse of patterns across different Web sites can reduce the efforts of designing a graph grammar, we plan to further improve the efficiency by applying a grammar-induction algorithm to semi-automate the grammar design. Given sample interface snapshots, human experts first highlight most important substructures and give them some user-friendly names. Then, based on those manually highlighted substructures, a grammar induction algorithm produces a graph grammar. In addition to elaborating an induced graph grammar, human experts can also add spatial specifications to the generated graph grammars. Once the automatically produced graph grammar is evaluated and refined, it is used to interpret Web interfaces.

An analysis based only on the document source and DOM structure may not produce an optimal result due to the document’s complexity. DOM structure does provide useful information for page segmentation. For example, one paragraph may cross several lines. In the image analysis process, each line of texts is recognized as one individual interface object. We anticipate that when integrating the image analysis results with the corresponding DOM structure, we could combine those individual lines of texts together and further simplify the spatial graph. In the future work, we plan to combine the DOM-based analysis with image analysis in the first step of our approach.

V. ACKNOWLEDGMENTS

The authors would like to thank O. Barkol and R. Bergman of HP Labs for providing us the image segmentation results. This work is supported in part by ND EPSCoR State Seed Grants #FAR0015845.

VI. REFERENCES

- [K.L. Ates and K. Zhang, "Constructing VEGGIE: Machine Learning for Context-Sensitive Graph Grammars", *Proc. 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'07)*, Patras, Greece, pp. 456-463, 2007.
- [Ahm08] H. Ahmadi and J. Kong, "Efficient Web Browsing on Small Screens", *Proc. ACM International Working Conference on Advanced Visual Interfaces - AVI'2008*, pp. 23-30, 2008.
- [Bal06] S. Baluja, "Browsing on small screens: Recasting Web-page Segmentation into an Efficient Machine Learning Framework", *Proc. WWW'06*, pp.33-42, 2006.
- [Bar09] O. Barkol, Omer, R. Bergman, A. Pnueli, S. Schein, "Semantic Automation from Screen Capture", *HP Labs Technique Report HPL-2009-161*, 2009.
- [Bot04] P. Bottoni, M. D. Marsico, P. D. Tommaso, S. Levialdi, D. Ventriglia, "Definition of visual processes in a language for expressing transitions", *Journal of Visual Languages and Computing*, 15(3-4), pp.211-242, 2004.
- [Bra95] F. J. Brandenburg, "Designing Graph Drawings by Layout Graph Grammars", *Proc. International Workshop on Graph Drawing*, LNCS 894, pp.416-428, 1995.
- [Bur05] R. Burget, "Visual HTML Document Modeling for Information Extraction", *Proc. RAWS'05*, pp. 17-24, 2005.
- [Buy01a] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke, "Accordion Summarization for End-Game Browsing on PDAs and Cellular Phones", *Proc. ACM SIGCHI'01*, pp. 213-220, 2001.
- [Buy01b] O. Buyukkokten, H. Garcia-Molina, A. Paepcke, "Seeing the Whole in Parts: Text Summarization for Web Browsing on Handheld Devices", *Proc. WWW-10*, 2001.
- [Che03] Y. Chen, W. Y. Ma, and H. J. Zhang, "Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices", *Proc. WWW'03*, pp. 225-233, 2003.
- [Che05] J. Chen, X. Xie, W. Ma, and H. Zhang, "Adapting Web Pages for Small-Screen Devices", *IEEE Internet Computing* 9(1), pp. 50-56, 2005.
- [Cor97] A. Corradini, U. Montanari, F. Rossi, H.Ehrig, R. Heckel, and M. Löwe, "Algebraic Approach to Graph Transformation – Part I: Basic Concepts and Double Pushout Approach", In G. Rozenberg (Ed.), *Handbook on Graph Grammars and Computing by Graph Transformation: Foundations*, Vol.1, World Scientific, pp.163-245, 1997.
- [Cos97] G. Costagliola, A. De Lucia, S. Orefice, and G. Tortora, "A Parsing Methodology for the Implementation of Visual Systems", *IEEE Transactions on Software Engineering*, Vol.23(12), pp. 777-799, 1997.
- [Cos00] G. Costagliola and G. Polese, "Extended Positional Grammars", *Proc. 16th IEEE Symposium on Visual Languages*, pp.103-110, 2000.
- [Cos02] G. Costagliola, S. D. Martino, F. Ferrucci, and F. Pittarello, "Human-Computer Interaction: An Approach for Authoring 3D Cultural Heritage Exhibitions on the Web", *Proc. the 14th international conference on Software engineering and knowledge engineering*, pp. 601-608, 2002.
- [Fis98] T. Fischer, J. Niere, L. Torunski, and A. Zündorf, "Story Diagrams: A New Graph Rewrite Language Based on the Unified Modeling Language and Java", *Proc. Theory and Application to Graph Transformations*, LNCS 1764, pp. 296-309, 1998.
- [Gu02] X. D. Gu, J. L. Chen, W. Y. Ma, and G. L. Chen, "Visual Based Content Understanding towards Web Adaptation", *Proc. AH'02*, pp. 164-173, 2002.
- [Gup03] S. Gupta, G. Kaiser, D. Neistadt, and P. Grimm, "DOM-based Content Extraction of HTML Documents", *Proc. WWW'03*, pp. 207-214, 2003.
- [Kon06] J. Kong, K. Zhang, and X.Q. Zeng, "Spatial Graph Grammar for Graphic User Interfaces", *ACM Transactions on Human-Computer Interaction*, Vol.13(2), pp. 268-307, 2006.
- [Kon08] J. Kong, K.L. Ates, K. Zhang, and Y. Gu, "Adaptive Mobile Interfaces Through Grammar Induction", *Proc. 20th IEEE International Conference on Tools with Artificial Intelligence*, pp.133-140, 2008.
- [Mil02] N. Milic-Frayling, and R. Sommerer, "SmartView: Flexible Viewing of Web Page Contents", *Proc. 11th World Wide Web Conference (poster paper)*, 2002.
- [Rek97] J. Rekers and A. Schürr, "Defining and Parsing Visual Languages with Layered Graph

- Grammars”, *Journal of Visual Languages and Computing*, 8(1), pp.27-55, Feb. 1997.
- [Sch95] A. Schürr, A. Winter, and A. Zündorf, “Graph Grammar Engineering with PROGRES”, *Proc. ESEC’95*, LNCS 989, pp. 219-234, 1995.
- [Sch99] A. Schürr, A. J. Winter and A. Zündorf, “The PROGRES Approach: Language and Environment”, In H. Ehrig, G. Engels, H. J. Kreowski, and G. Rozenberg (Eds.), *Handbook on Graph Grammars and Computing by Graph Transformation: Applications, Languages and Tools*, Vol.2, World Scientific, pp.487-550, 1999.
- [Var02] D. Varró, “A Formal Semantics of UML Statecharts by Model Transition Systems”, *Proc. ICGT 2002*, LNCS 2505, pp.378-392, 2002.
- [Yan00] Y. D. Yang, J. L. Chen, and H. J. Zhang, “Adaptive Delivery of HTML Contents”, *Proc. WWW’00*, pp.24-25, 2000.
- [Yan01] Y. D. Yang and H. J. Zhang, “HTML Page Analysis Based on Visual Cues”, *Proc. 6th International Conference on Document Analysis and Recognition*, pp. 859-864, 2001.
- [Yu03] S. P. Yu, D. Cai, J. R. Wen, and W. Y. Ma, “Improving Pseudo-Relevance Feedback in Web Information Retrieval Using Web Page Segmentation”, *Proc. International Conference on World Wide Web*, pp. 11-18, 2003.
- [Zha97] D.Q. Zhang and K. Zhang, “Reserved Graph Grammar: A Specification Tool for Diagrammatic VPLs”, *Proc. 1997 IEEE Symposium on Visual Languages (VL’97)*, Capri, Italy, pp. 284-291, 1997.
- [Zha01] D. Q. Zhang, K. Zhang, and J. Cao, “A Context-Sensitive Graph Grammar Formalism for the Specification of Visual Languages”, *The Computer Journal*, 44(3), pp. 187-200, 2001.