# Yet Another Hybrid Segmentation Tool

Andrés Sanoja, Stéphane Gançarski

# Yet Another Hybrid Segmentation Tool

Andrés Sanoja
LIP6
Université Pierre et Marie Curie
Paris, France
andres.sanoja@lip6.fr

Stephane Gançarski
LIP6
Université Pierre et Marie Curie
Paris, France
stephane.gancarski@lip6.fr

## ABSTRACT

In this paper[1] we present an overview of a prototype we are developing for in the context of web archives (page comparison, crawling and information retrieval). It analyses pages based on their DOM tree information and their visual rendering. This tool implements a modified version of VIPS with the aim of enhancing the precision of visual block extraction and the hierarchy construction. First, the visual rendering of a page, produced by several browsers, is segmented into rectangular blocks. Then, the extracted blocks are analysed looking for visual overlaps, which are analysed using a adapted version of the XY-Cut algorithm and resolve the overlap. As a result we may have different shapes of blocks, rectangular and non-rectangular blocks. Finally, the visual block tree, representing the layout of the page is analysed in order to have a more coherent layout disposition.

## Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

## General Terms

Algorithms

## Keywords

digital preservation, web archiving, web page segmentation

## 1. INTRODUCTION

The purpose of this work is describe the open-source, platform-independent and functional prototype under development for web page segmentation. It is based in a modified version of the VIPS algorithm [2] enhancing the block detection procedure to avoid or to solve overlapping and to set the order of blocks into the hierarchy, according to its position in the document. It is based on the idea that sometimes the source code is not enough to be able to reproduce the user perspective and also the rectangle shape could not be well suited to frame a block because overlapping may occur. Our main objective is to enhance the precision of block detection, therefore a better block information for building web search indexes [1] and web archiving [4], for example.

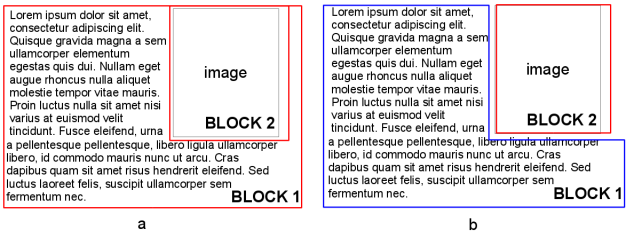


**Figure 1: Overlapping blocks problem**

After segmenting a page, under the user perspective, it is possible that one node can be under the area of two sibling blocks (figure 1a), this means overlapping. We think this is ambiguous. The visual representation does not correspond to the underlying source code and thus it is not obvious to know to which block a node corresponds to. Figure 1b depicts how with we think this ambiguity could be resolved: block 1 with a non rectangular shape will fix the overlap problem. The position of the blocks in the hierarchy do not necessarily correspond with the layout flow of the document. We think that the hierarchy should be self-contained if we observe it, in other words, infer the structure or the layout of a page from it. For example a header should comes before footer. A similar problem was reported by Yang [5] who use two approaches to web page segmentation (VIPS and Gestalt-based one) to classify the roles of images as block features. They need to have a set of no overlapped blocks, and they present a method for achieved it.

In this paper we describe a prototype for segmenting web pages in the framework of the SCAPE project, particularly in the web archive context. The segmentation algorithm is introduced. It follows a hybrid approach for enhancing the precision of page segmentation. It uses a modified version of VIPS [2] for DOM segmentation and adapted version of the XY-Cut algorithm[3] for image segmentation.This paper is organized as follow. In section 2 we describe the hybrid approach which we propose. Section 3 introduce the proto-

[1]This work was partially supported by the SCAPE Project. The SCAPE project is co-funded by the European Union under FP7 ICT-2009.4.1 (Grant Agreement number 270137)

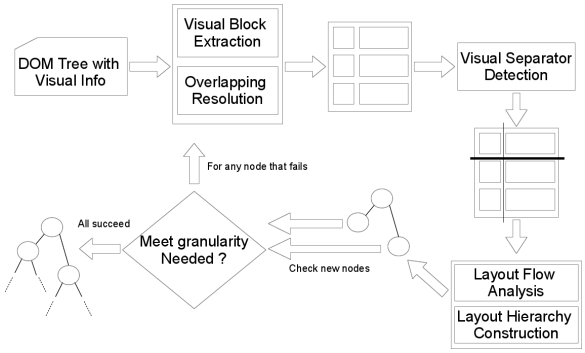


**Figure 2: Segmentation Algorithm: VIPS Algorithm with enhancements**

**Figure 3: Data Visualization and User Interaction**

type. Section 4 concludes.

## 2. HYBRID APPROACH FOR WEB PAGE SEGMENTATION

In this section we describe our hybrid segmentation approach. In order to get a more precise segmentation we enhance the VIPS Visual Block Extraction phase and hierarchy construction. Figure 2 depicts the segmentation algorithm, based on the original algorithm with enhancements.

The segmentation algorithm is applied recursively to obtain the visual blocks and the hierarchy. If overlaps are detected the blocks affected are analysed using the area held by them and the XY-Cut algorithm is applied. A rectangular block is removed if the new non-rectangular blocks are found inside. The original block is kept if the new block is slightly different. It is possible to remove several original blocks in the case of that the new non-rectangular block covers them. At the same time the resulting hierarchy is analysed in order to have an order similar as in the layout flow. The blocks position are comparing to previously defined layout patterns

## 3. PROTOTYPE DESCRIPTION

An open-source, platform-independent and functional prototype is been developed through the hybrid web page segmentation approach. It is composed of a module for managing the data visualization and user interaction, a module for preprocessing input files and another for page segmentation. Two input files are required: 1) the decorated HTML which is the original source document without tags that give no information for the segmentation and for each remaining tags, extra attributes are added to denote the visual cues and 2) the page screenshot

To produce these files the web pages are downloaded and rendered using several browsers. The reason to use several browsers is that we would like to evaluate the segmentation results. These pages are processed using Selenium. The visual cues are obtained with a set of JavaScript scripts that are injected to the browsers and the screen-shots using selenium features. Figure 3 depicts the web application that is available to visualize and manage the resulting blocks. Altering the blocks geometry is considered as feed-back from user, that is stored for later use.

The page processing is implemented as described in the section 2. Ruby 1.9.2 is used as programming language for implementing the segmentation algorithm, Hpricot/Nokogiri libraries are used for HTML/XML manipulation and a modified version of the XY-Cut algorithm for image manipulation .

## 4. CONCLUSION AND FUTURE WORK

In this paper we described a prototype for Web Page Segmentation. This tool aims to be part of the tools developed in the framework of the SCAPE project. Our paper points out the issues of efficiently segmenting web pages and improving the quality and the relevance of the segmentation. To address this issue, we propose an hybrid approach that use both the DOM analysis and image segmentation. We look forward to improve more significantly the precision of the segmentation and include those insights in all aspects of an archive, indexes for example.

## 5. REFERENCES

[1] E. Bruno, N. Faessel, H. Glotin, J. Le Maitre, and M. Scholl. Indexing by permeability in block structured web pages. In *Proc. of the 9th ACM symposium on Document engineering*, DocEng '09, pages 70–73, New York, NY, USA, 2009. ACM.

[2] D. Cai, S. Yu, J.-R. Wen, and W.-Y. Ma. Extracting content structure for web pages based on visual representation. In *Fifth Asia Pacific Web Conference (APWeb'03)*, 2003.

[3] B. Kruatrachue, N. Moongfangklang, and K. Siriboon. Fast document segmentation using contour and x-y cut technique. In C. Ardil, editor, *WEC (5)*, pages 27–29. Enformatika, Çanakkale, Turkey, 2005.

[4] M. B. Saad and S. Gançarski. Using visual pages analysis for optimizing web archiving. In *Proc. of the 2010 EDBT/ICDT Workshops*, EDBT '10, pages 43:1–43:7, New York, NY, USA, 2010. ACM.

[5] X. Yang and Y. Shi. Enhanced gestalt theory guided web page segmentation for mobile browsing. In *Proc. of the International Joint Conference on Web Intelligence and Intelligent Agent Technology*, pages 46–49, Washington, DC, USA, 2009.