Measuring Performance of Web Image Context Extraction

Sadet Alcic  
Institute of Computer Science  
Heinrich-Heine-University of Duesseldorf  
D-40225 Duesseldorf, Germany  
alcic@cs.uni-duesseldorf.de  

Stefan Conrad  
Institute of Computer Science  
Heinrich-Heine-University of Duesseldorf  
D-40225 Duesseldorf, Germany  
conrad@cs.uni-duesseldorf.de  

ABSTRACT
Images on the Web appear with textual contents providing meaningful information to their semantics. Methods that automatically determine and extract the Web image context from an HTML document are widely used in different applications. However, the performance of the image context extraction has rather been evaluated on its own. Keeping this imperative in mind, we present a framework to objective evaluation and comparison of the performance of image context extraction methods. This is achieved by collecting a large ground truth dataset consisting of diverse Web documents from real Web servers and by defining performance measures adapted to fit the special properties of the context extraction task. To show the capabilities of the proposed framework, common extraction methods from the literature have been evaluated and the results are summarized in this paper.

Categories and Subject Descriptors
H.3 [Information Storage and Retrieval]: Content Analysis and Indexing—Indexing methods; I.7.2 [Document Preparation]: Index Generation, Multi/mixed media

General Terms
Data Extraction from Web Documents

Keywords
web image context extraction, web image indexing, web image mining

1. INTRODUCTION
Web Image Context Extraction (WICE) deals with the problem of automatically determining the textual parts of an HTML document which are associated with an image within this document. The textual information - referred to as image context - is assumed to share the semantics with the associated image and thus is a key technique in many Web images processing applications.

A Web user looking at a Web image mostly has the intention to know more about the depicted objects and determines the textual context unconsciously. While humans perform this task almost intuitively by exploring the image environment searching for structural and semantic hints, the automatic image context extraction confronts with the well known semantic gap. To our knowledge, there are no reliable methods to ascertain the semantics of images, thus the semantic-based hint cannot be used in a machine based approach. On the other hand, the structural hints are barely manageable due to the high diversity of designing patterns and programming skills among Web designers. Moreover, Web documents contain clutter and noisy contents like layout graphics, functional and navigational objects, and advertisement making the context extraction even more difficult.

In the past decade, a number of approaches employed WICE methods in different application areas. Particularly, the context plays a main role for storage, maintenance and access of Web images and is thus the holy grail for commercial image search engines [2, 3]. Further, in different Data Mining and Machine Learning applications, WICE provides high level image descriptors which improve methods relying only on visual features. But since the WICE methods present only a preprocessing step in existing approaches, only the impact of the methods on their main application was evaluated in the literature. Moreover, to our knowledge there are hardly attempts to compare the results of different WICE algorithms at all.

With this contribution, we bridge this gap providing a framework which can measure and compare the quality of existing WICE methods. Common approaches to WICE are implemented within the framework and are used to show the functionality of the system. We further provide test data collections extracted from real web servers as input for the WICE methods and different quality measures to judge the extraction quality.

The remainder of this paper is organized as follows: Section 2 presents related work and highlights the differences in our approach. In Section 3 the problem of image context extraction is briefly formulated. The framework design with all its modules and a short presentation of the implemented WICE algorithms are presented in Section 4. Section 5 summarizes the evaluation results. And finally the work is concluded in Section 6.
2. RELATED WORK

Image context extraction plays a main role for a number of applications in the area of web image retrieval and web mining. In this section we want to present some previous works with focus on their evaluation of the used extraction methods.

Souza Coelho et al. [6] list four sources of evidence (textual descriptions within a Web document that can be associated to web images): description text (image alternative text and the text within common anchor tags), meta tags (located in the HTML head of the document), full text, and text passages (plain text, placed next to image in document representation). Within an evaluation task they analyze the image retrieval performance based on the different evidence sources applied for image indexing. They have found that the surrounding text passage consisting of 20 terms before and after the image (passages of 10, 20 and 40 terms were separately inspected) performed best for single evidence ranking. However, other methods to passage extraction were not investigated.

Another application, where the surrounding text passage plays a great role is Image Rover [12]. The image indexing in this system is based on textual and visual cues of the images while the textual are obtained from the Web documents text. Document text parts are weighted depending on their structural conditions and the surrounding text has one of the highest weights. The system performance has been evaluated by applying the target test paradigm [13], which tests how efficiently a system performs in finding a target image in the data collection. As in [6] the presented evaluation shows only the impact of one context extraction method to the image retrieval task and no other extraction methods were applied for comparison.

Tian et al. [14] use beside visual and relational image features also textual ones that are extracted from the sibling nodes of an Web image in the DOM representation of an HTML document. Based on the associated features the images are classified and the classification performance was analysed. As an alternative textual context extraction method VIPS [5] was mentioned but not applied in the scenario due to its high complexity.

The Vision-based Page Segmentation (VIPS) algorithm [5] partitions a Web document in visual blocks. In [4, 8] this method was successfully applied to image context extraction by using the block structure to find text-to-image associations. The effectiveness of VIPS was only judged indirectly. Cai, et al. [5] have tested VIPS’s segmentation quality employing 5 human users who classified the segmentation results to perfect, satisfactory, fair and bad. He, et al. [8] evaluated the web image search and the image clustering results to perfect, satisfactory, fair and bad. However, other methods to passage extraction were not investigated.

In our work is most similar to the experimentation studies applied in [7]. In order to evaluate their suggested WICE method (based on image use cases), a dataset of 100 Web pages was collected from real Web servers and the image context was determined by volunteers, while at least 3 people have processed every document. Differences in the context associations determined by different humans were replaced by the broadest context. The image context extraction was performed using the proposed algorithm and the VIPS demo application [1] for PDoC values of 5, 6, and 7. As evaluation metrics the precision and recall values were computed for the complete data set, where precision defined the percentage of correctly extracted image contexts over the complete extracted contexts, and respectively the recall presents the percentage of correctly extracted contexts over the complete correct contexts.

In our evaluation framework the performance metrics are adapted to handle partially accordance of extracted and relevant context, because we think that the testing on exact matches between ground truth data and computed image context is a too strong criterion. The large ground truth collection prepared to assist the framework contains a much higher variety than those used in previous works. Further, at least one extraction method from the different method classes (heuristics-based, DOM-based and vision-based) was evaluated. To our knowledge, we are the first to integrate such a variety of WICE methods within one evaluation task.

3. PROBLEM FORMULATION

Web Image Context Extraction terms the process of determining the textual parts of an HTML document that are associated with an image from this document. In general this process can be described by a function \( f \) taking the HTML document and an image within this document as input, which computes a set of begin- and end-markers as limits determining the image context in the documents’ source. We formalize \( f \) as follows

\[
f(D, I_D) = \{(b_i, e_i)|0 \leq b_i \leq e_i \leq |D|\},
\]

with \( D \) as the HTML document and \( |D| \) the document size as number of characters, \( I_D \) an image in document \( D \) and \((b_i, e_i)\) as a pair of begin and end markers, pointing to the start and end position of one part of the images context in the string representation of the document. Since the image context can be distributed over the document the index \( i \) is used to enumerate the separated context parts.

With this formulation the implemented algorithms are supposed to compute a finite set of tuples \((b_i, e_i)\), each outlining the parts of the context of an image within a document.

As the HTML documents are represented in plain text, the output of WICE methods can be of the following types:

1. **Set of tuples**. This representation corresponds to the output of the defined function \( f \). The markers define the start and end limits of the image context.

2. **Markups within Document**. The results can be stored directly within the input document using special tags and attributes specifying the associated image.

3. **Context String**. The most common way to present the context to human users would be the string representation, because it consists of text in natural language.
In our framework we chose the last method as output format, since it does not depend on the input document and is most evident for human users. Moreover, the conversion from the tuples-based output to the string representation is straightforward.

4. EVALUATION FRAMEWORK

4.1 Framework Design

The evaluation framework proposed in this work consists of different modules as depicted in Fig. 1. All modules are extensible since they are independent from each other. Only the defined interface specifications have to be followed.

The flow starts with preparing a suitable collection consisting of documents derived from real Web servers. A document of this collection is then passed to the image filter, which removes the noisy images from the document. In the next step, this document serves as input for the WICE method and for the Oracle. Although the input and output specifications of Oracle and WICE methods are the same - the output is a set of image-content pairs in both cases - they have a significant functional difference, since the Oracle produces the ground truth benchmark. In the next step the efficiency of a WICE method can be estimated by comparing its output to the benchmark data. To do this, a suitable measure is applied to quantify the correspondence between extracted contents and the ground truth. Detailed information to the particular modules will be presented in the following subsections.

4.2 The Oracle and Test Collections

To be able to measure the performance of WICE methods on a particular document and image pair, we need to specify what the image context is. In other words, the Oracle in the workflow in Fig. 1 has to replace a feasible method.

Our aim is to provide a test collection which will envelop the high variety of HTML documents on the web and further contain a huge amount of documents to test the robustness of the extraction methods. An obvious way to create such a benchmark is to collect Web documents from various Web servers and to assign an human expert who will decide which context belongs to what image on a given rendered web page. Though this step is not always deterministic and suffers from the subjectiveness of the human, we trust to our expert at least to have a good idea of which textual parts should be selected as image context. The resulting test collection comprises 80 randomly chosen documents from 10 different categories of the yahoo directory. As these documents are analyzed by our expert, we decided to save them with all the data (e.g., images, style sheets and scripts) needed to view the documents offline.

Although this collection comprises a high variability, larger amounts of Web pages are needed to investigate the performance and reliability of the extraction algorithms. But since manual context detection is tedious and labour intensive for the expert, automatization of this process is needed.

According to [10] building a specific web data extractor tailored to a particular document or a set of documents sharing the same template is feasible. For this purpose, a script was written that periodically revisits a domain and stores the corresponding source if a significant change of the document is detected since the last access. The higher the content change rate of a Web domain is, the faster the collection grows (thus most popular news domains are suitable to this approach). 11 different domains were applied to this task, resulting in 11 different collections. Finally, for each of the collections, a rule based extractor [10] was implemented. The extractor is based on the DOM tree of a document, where each image is represented by an image tag. An example for an extracting rule is given below:

```c
// 1st rule
IF ImgTag has ParentNode with Attribute a = "b"
THEN
  extractAllTextNodes under ParentNode as Context;
  exit;
// 2nd rule
...```

The order of the rules is very important, since the images covered by different rules may overlap. For some templates, rules can be much more complex, excluding particular textnodes under a parent node.

The properties of the resulting test collections are summarized in Table 1. According to these values, to our knowledge this is the greatest existing test collection of extracted image-context pairs.

In the following step a document of a collection is passed to an Image Filter which will be described next.
4.3 Image Filtering

Web images are used for different purposes within Web documents. They can be roughly categorized to navigation, decoration, advertisement and story. But only the last category is expedient for context extraction and further processing. Thus a filtering method basically relying on the image dimensions was employed to determine the story images.

We follow a common filtering approach as used in [4] by accepting only images with height and width lengths greater than 60 pixels (this eliminates the most background images). Further, we expect a side length ratio smaller than 2.75 (since navigational images and advertisements own a grater ratio). Also images that are used more than one time within a document are assumed to be not of the story part.

The image dimensions are taken from the image tag attributes if available, otherwise the image is loaded from the server. Although, some advertisement images could not be detected correctly, this method proved to achieve high accuracy in our test runs.

4.4 Web Image Context Extraction Methods

There is a variety of WICE methods proposed in the literature. They all can be roughly categorized by their principal approaching to context determination. In this work we distinguish

- **Heuristics-based Approaches** - Extractions are performed using empirically determined heuristic rules without any document structure analysis.

- **DOM-based Approaches** - The extraction rules are based on the DOM tree structure. Image context is assumed to be found within structural blocks framed by DOM nodes.

- **Visual-based Approaches** By analyzing different tag properties a visual-block structure of a document is computed. The image context is assumed to be within same visual block as the image.

In the proposed framework at least one method of each category has been implemented. Next, we will describe every of them in detail.

4.4.1 Full Plain-Text

One of the simplest methods to context detection is to associate the complete text of a web document with the images within this document. At first glance, this approach seems to be defective since in multi-topic documents every image will get the same context resulting in reduction of the precision. Nevertheless, this method has been successfully applied to image indexing for web image retrieval [11]. We have implemented this baseline in our system to show what accuracy is reachable without any selection but also to compare and highlight the benefits of particular WICE methods.

4.4.2 N-Terms-Environment

Web images and their context information appear next to each other in web documents. This fact allows the presumption, that image and corresponding context are neighbored in the document source code, too. [6] has chosen the environment of 20 terms as image context to index Web images. This context selection was further applied in [12].

The implementation of this method requires that the web page is transferred to an sequence $S$ of words and images. The WICE problem can now be formulated as finding a sub-sequence of $S$ which contains the $N$ words before and after an image. Since the images are also stored in a sequence $S$ the computation of such a subsequence $s_i$ for an image at $S[i]$ is straightforward:

$$s_i = (S[i-N],...,S[i-1],S[i+1],...,S[i+N]),$$

which has to be adapted when the borders of $S$ are reached. Further, the images possibly occurring within $s_i$ must be eliminated.

In this evaluation, we have applied the 10 and 20 term neighborhood as proposed in [12].

4.4.3 The Monash Extractor

Fauzi et al. [7] propose a DOM-Tree based method to image context extraction which is based on a presumption that there are three main characteristics of images within HTML documents: listed images, semi-listed images and unlisted images. The distinction of these types is done based on the DOM-Tree representation. In particular, starting at an image node, the algorithm traverses up the tree until the number of text nodes in the actual subtree changes. The actual node at which the algorithm stops is stored as $n_1$. Then the algorithm searches for semi-listed images under $n_1$.

If semi-listed images are found, then the boundaries of the image segment are determined by looking for repeating structures within the children of $n_1$ and the text within these boundaries is considered as image context. If not, the algorithm traverses a second time upward the tree until the number of text nodes in subtree of the regarded parent node changes. At this node, referred as $n_2$, the algorithm checks the subtrees looking for repeating structures which would indicate a list. If such a list is detected then the text un-
under \( n_2 \) is considered as image context, else the image is an unlisted image and the whole text under \( n_2 \) represents the context segment.

The authors did not name their extraction method. To be able to refer to this WICE method, we have chosen to call it the Monash Extractor referring to the institution the authors belong to.

4.4.4 Siblings Extractor
The siblings extractor is another DOM-based method, which is much simpler than the Monash method. This kind of context determination was used in [14]. The basic idea behind this method is that DOM nodes represent structural blocks within a webpage and thus the sibling text nodes of an image node are used as context providers. For implementation, it is just necessary to find the parent node of the image, that contains text nodes within its subtree. The text of these text nodes is then extracted and defines the image context.

4.4.5 Vision-based Page Segmentation (VIPS)
The VIPS algorithm [5] was originally developed for web page segmentation. It is an hierarchical top-down approach, which starts with the whole page as initial block. For each block, a Degree of Coherence (DoC) is computed using heuristic rules based on the DOM Tree structure and visual cues obtained from the browser representation. The DoC value determines how much the contents within a block correlate to each other. It ranges from 1 to 10 while 10 represents the highest correlation. At the beginning a Permitted Degree of Coherence (PDoC) value is specified, which controls the segmentation granularity. If a particular block has a DoC value smaller than PDoC, this block has to be subdivided and this rule is repeated until all blocks on the bottom fulfill the mentioned condition.

Although the VIPS algorithm was not primarily designed to extract the context for a web image, we can use the segmented block structure to assign the text of a block to an image within this block as used by Cai et al. in [4]. He et al. [8] estimated a PDoC of 5 as a best suited value. We have extracted the image context for a PDoC value of 5, 6 and 7, since in our runs, the PDoC value of 5 seemed to be too coarse.

Since the VIPS library exists only for the Windows operating system, we run a batch job over our web site collections and stored the results in XML files for different PDoC values. In a further processing step the image context was extracted from the XML files.

4.5 Output format
The storage of image and context associations needs the definition of a unified output format. As we already mentioned in the last section, we prefer storing the complete text representation of the context over the storage of begin and end markers referring to absolute positions in the source code of the original documents, since the first does not require any retention of source documents.

For each document the image url and the corresponding context are stored within a Comma Separated Values (csv) file. This format was used for both collections, the extracted results as well as the ground truth.

4.6 Performance Measures
The comparison of the extracted context with the ground truth data needs appropriate performance measures which reward the congruence between the extracted and the relevant context and penalize any divergence of these document excerpts.

Since the exact specification of image context can even be difficult to experts, testing on exact matches poses a too strong criterion. Instead partially accordance between the experts judgements and the output of a context extracting algorithm should be considered.

The definition of WICE as a function of an HTML document \( D \) and an image \( I_D \) that computes the textual parts representing the image context in Section 3 allows us to view at the context determination task from the Information Retrieval (IR) perspective, where \( I_D \) is the query on a single document \( D \).

Keeping this interpretation in mind, we can adopt the common IR measures to quantify the performance of WICE methods. In IR the concept of precision \( P \) is defined as the ratio of the correctly extracted objects to all extracted objects, and the concept of recall \( R \) is the ratio of the correctly extracted objects to all relevant objects. Since these measures complement each other, the harmonic mean of both comprised in the F-score provides a suitable performance measure to compare the extracted objects to the relevant objects. It is defined as follows:

\[
F_{\text{score}} = 2 \cdot \frac{P \cdot R}{P + R}
\]

The usage of the mentioned IR measures in the context extraction scenario requires a specification of what the extracted and relevant objects are. As the context is written in natural language, we can simply use the individual words of the document as these objects. The context can then be represented as a sequence of words. The computation of the intersection for two word sequences, as required to compute precision and recall, is accomplished by the longest common subsequence of the two sequences. Therefore the algorithm proposed in [9] by Hirschberg was adopted.

Further, we wanted to analyze the stability of the WICE methods within a specific document collection \( C \). Therefore, we compute the standard deviation of the F-score within a collection \( C \), which is defined as

\[
\sigma(C) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (F_i - \mu)^2}
\]

where \( F_i \) is the F-score for a document \( d_i \in C \) and \( \mu \) is the average F-score over the complete collection \( C \). Further \( N \) is the number of documents in \( C \).

5. EVALUATION RESULTS
The results of the performed evaluation are summarized within the diagrams in Figures 2 and 3. The diagrams show the precision, recall, F-score and the standard deviation of the F-score for every included WICE method. The precision and the recall values were admitted in the diagrams to
reason the $F$-score. The title of each diagram refers to one of our test collections.

Figure 2 shows the results for the test collection created with simpler extraction rules. Since the extraction rules are based on the DOM tree, we had the assumption that the DOM based WICE methods will reach good results on these collections.

As shown in the diagrams this assumption is supported by the evaluation results. The Monash and the Siblings methods perform best with an $F$-score value in the range of 85 to 100 percent. The standard deviation of $F$-score for the DOM-based methods is relatively small and hence the reliability of these algorithm is high. The N-Term Extractor has a lower accuracy, since its $F$-score ranges within the middle third. The precision value indicates that about the half of the extracted text does not belong to the image context. This can be explained by analyzing the a small portion of the test documents, considering that the image context is mostly placed either before the image or after it. The N-Term methods choose the $N$ words before and after the image, therefore half of them is falsely selected. The recall values vary from 30 to almost 86 percent depending on the collection. Of course, the recall for 20 term method is always higher that the 10 term method. An significant advantage of the 20 term over the 10 term method could not be detected, as there is no winner according to the $F$-scores. It is interesting, that the standard deviation of $F$-score for the N-Term methods is very small indicating the similar structure for all the documents within a test collection.

The VIPS algorithm has almost always an excellent recall value but its precision is relatively small which results in small $F$-score values. This indicates, that the visual seg-
In Figure 3 the diagrams with the evaluation results for the collections which needed more complex extraction rules are shown. Also the results for the diverse collection of manual extracted context pairs is included in this Figure.

To start with the VIPS based extractors, we remember these are based on the internet explorer browser. But because the collected BBC pages could not be displayed by this browser due to java scripting failures, VIPS was not able to deliver any context and thus the values are missing. On the other collections, VIPS performed best with the highest PDoC values. A higher PDoC value than 7 was not suitable, since then the visual block solely contained the image. The F-score ranges from fair for the telegraph collection to poor for the Yahoo! collection. As already discussed for the first group of evaluation results, VIPS has a high F-score standard deviation which signals that its reached performance varies from image to image.

The baseline algorithm extracting the full text of the document achieved always a recall of 100 percent (that was predictable), but since the precision value is nearly to zero the overall F-score behaves nearly the same. This fact implies the necessity for applicable WICE methods.

Figure 3: Evaluation results for test collections with more complex extraction rules.
The N-Term extractors reach very similar performance values as in the first evaluations group: the performance is not much dependent of \(N\), the \(F\)-score ranges in the middle third, and the standard deviation of \(F\)-score is low, indicating a high reliability and repeatability of extraction performance.

The siblings extractor could not solidly reach high performance values as in the first evaluation group, since the more complex template structures of the telegraph and the Yahoo! website yield the simple extractor to failure. For these collections, the other algorithms performed better. But surprisingly, the siblings extractor has reached a very high \(F\)-score in the diverse collection arguing that most of the webpages available in the internet contain a simple structured HTML code.

The Monash extractor performed best for almost all collections. Nevertheless, when the template structure gets more complex, its reliability and repeatability gets lower indicated in higher \(F\)-score standard deviation.

6. CONCLUSION AND FUTURE WORKS

In this work, an evaluation framework for image context extraction was proposed and tested for different common extraction methods.

As a main result of the evaluation task, the WICE methods based on DOM reached the best performance (almost always an \(F\)-score over 90), indicating that many HTML documents gathered from popular domains are well structured (with regard to their DOMs). As a second result - to our surprise - VIPS, one of the mostly applied methods to image context extraction in the literature, performed poorly on the test data, since the page blocks computed by VIPS were too broad.

The obtained results can be used to help scientists with the decision which method to use in their applications when image context is needed.

Further, the framework is easy extendable by other metrics and other extraction methods. Therefore, this system can also be used to evaluate and compare the performance of new extracting methods, which will be the main scope of our future work.

7. REFERENCES


