Webzeitgeist : Design Mining the Web

Cesar Torres[1,2], Ranjitha Kumar[1], Scott Klemmer[1]

[1]Department of Computer Science, [2]Department of Art and Art History
ctorres7@stanford.edu, {oranju, srk}@cs.stanford.edu

Motivation:

Content, links, and metadata are common features used in web indexing software. Although this representation provides adequate information for content-based search, major web design components are left out of the equation.

In order to leverage the web as a design corpus, a more complete design representation is needed. We introduce an architecture for scalable design mining and present some initial results for a data-driven exploration of design patterns.

Methods:

Our system currently consists of a production set of 100K indexed web pages and a subset of 10K web pages. For each of these pages, we have stored the DOM structure, runtime attributes (CSS), and the bento segmentation block algorithm representation of the DOM. A static instance of each page is stored in order to ensure consistent results. Features are extracted for each visual block that encode design attributes (e.g. color, layout, typography, etc..) from a variety of sources:

Semantic Features [36 features]
A crowdsourcing experiment asked participants to label the five most important elements on a page. We stored frequency data as well as substring matches to class and id attributes.

Visual Features
GIST Descriptor [960 features]
A global image feature (GIST) descriptor encodes the strength of horizontal and vertical lines in an image.

Structural Components [29 features]
Structure is represented through relationships of the visual block to its surrounding DOM environment. (e.g. treeLevel, numChildren, numSiblings, etc...)

Rendertime Style and Attributes (CSS) [318 features]
Each visual block's rendertime styling was computed using a headless browser. This reduced the dimensional space (px, em, pt, %) to a common basis (px). Fixed-set value attributes are binarized whereas allowable variable-length attributes are determined using a thresholded frequency plot. All values were mapped to [0, 1]. Unbounded quantities' theoretical bounds were calculated using frequency counts in the [2, 98] percentile.

Architecture

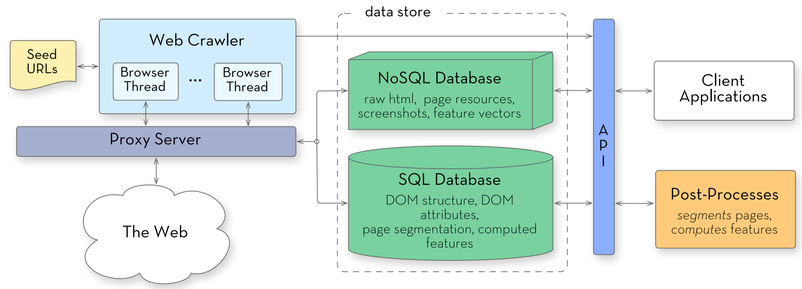


Figure 1. System Architecture - A web crawler begins by traversing the web from a seeded list. Typically, this crawler would only extract content-based information, limiting a complete design representation. Our approach extracts raw resources for a static representation and stores structure, runtime style, and visual information in a data store. A restfulAPI facilitates communicates between the data store and client applications.

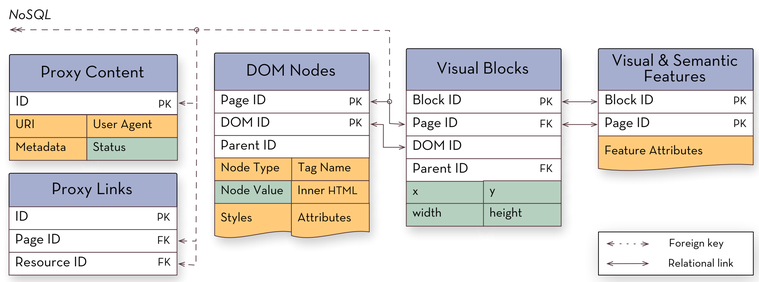
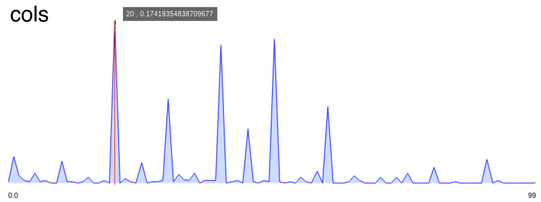


Figure 2. Schema - Raw resources were stored in a NoSQL database for quick random access, while DOM information and its representative visual block information were stored in a MySQL relational database. Note in particular the abstract feature table that provides a framework for extending the feature characterizations of visual blocks.

Feature Extraction:

Feature	*W3 Specification:*	*QtBrowser*	*Vector Representation*
float	rightlleftlnonelinherit	rightlleftlnone	isFloatRight, isFloatLeft, isFloatNone
font-family	family-name, genericlinherit	Helvetica, sans-serif	top 30 typeface (isHelvetica, etc..)
width	pxlptleml%lautolinherit	min: 0 , max: inf	theoretical mapping [0,1] 98th percentile : 1280px




```
<textarea cols='number'>
I am a text area.
</textarea>
```

I am a text area.

Results

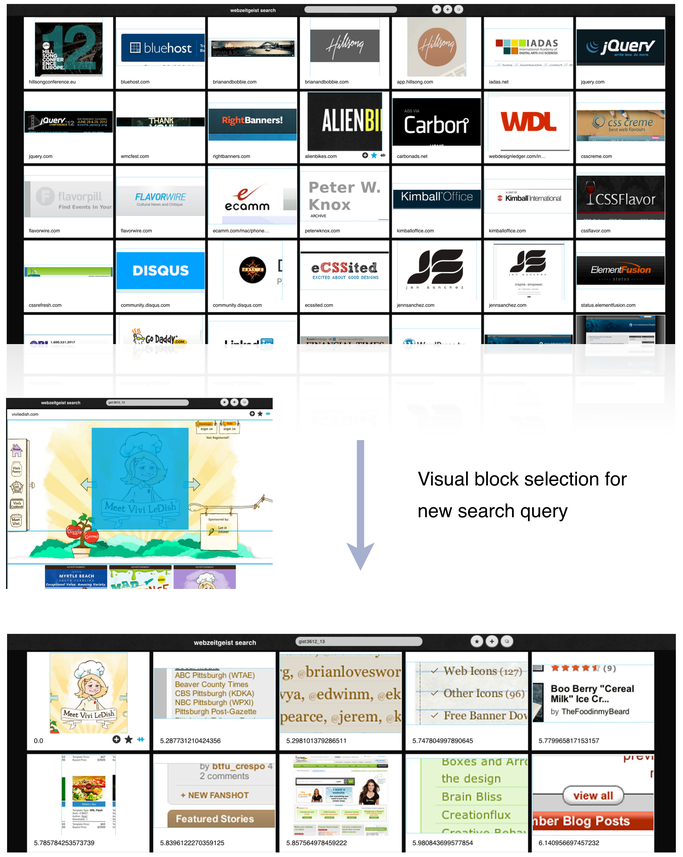


Figure 3. Search Application - The top figure displays results from a crowdsourcing study on semantic labels. The bottom figure displays the results of a 1000 node query on GIST features sorted by euclidean distance.

Future Work :

Omniscent
Through the Webzeitgeist infrastructure, we intend to train a distance metric for combinatorial search queries and present an interface for design exploration.

Probabilistic Web Inference Models
This work investigates learning design grammars using bayesian inference. We will construct a probability model that will infer web design patterns from a set of exemplars. An interface will allow for retargeting of visual blocks using the Webzeitgeist restfulAPI and semantic classifiers.