Obtaining Help on GUI Elements Using Screenshots

Tom Yeh Tsung-Hsiang Chang Robert Miller

EECS & CSAIL
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
{tomyeh,vgod,rcm}@mit.edu

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces. - Graphical user interfaces.

General terms: Human Factors

ABSTRACT
We present an image-based help mechanism for users to obtain information about novel and unfamiliar GUI elements. Instead of reading tooltips and/or using keywords to search, users can use the screenshots of the GUI elements directly to retrieve relevant information. Many offline and online resources contain screenshots and text about software applications that can be indexed and searched based on visual similarity. A preliminary experiment on a small pilot dataset has demonstrated the technical feasibility of a state-of-the-art image matching algorithm to handle screenshots.

Introduction
When users encounter novel and unfamiliar graphical user interface (GUI) elements, such as newly installed widgets or seldom used toolbar icons, they often rely on several mechanisms to figure out the roles and functions of these elements. They can try to decipher the visual meaning directly and make the best educated guess, but they do not always succeed. They can read the captions instead of guessing, but captions are not always available because of the scarcity of the screen estate. They can hold the mouse cursor on top of an element and wait for a tooltip to appear, but sometimes the tooltip may be too succinct to offer enough information or may never appear. They can turn to the official documentation for help, but locating the relevant sections or paragraphs in a lengthy documentation can be tricky. They need to either browse the table of contents to identify the section title that seems relevant, or use the words in the caption or the tooltip to search the index for paragraphs or sentences that might be related. But both methods are indirect and can be difficult to apply in the absence of a caption or a tooltip. In some applications, they can press a hot-key (e.g., F1) to directly retrieve help on the element with the cursor focus in a context-sensitive manner, but this feature has not become prevalent; it is not only costly to implement because of the need to insert triggers into all GUI elements, but also inflexible because of the difficulty of updating these triggers once the application is released. The last resort is often the Internet; users can use a search engine to find help from the vast online repository of tips, tutorials, Q/As, and discussions contributed by the entire user community. However, current search engines are based on keywords, a modality less than ideal for specifying the search terms and prone to ambiguity when the search target is a GUI element with rich visual attributes.

Approach
To overcome the shortcomings of the help mechanisms above, we propose an image-based method for obtaining help on a GUI element, where users can use the image of this element directly to find information about it (Figure 1). The input operation is similar to invoking the tooltip of a GUI element. A user moves the mouse pointer on top of the element he or she wishes to inquire. After holding the pointer at the location for a moment, a bounding box enclosing the GUI element appears alongside the tooltip. The image in the box is automatically captured and will be used as the query image to search a database of images of known GUI elements (more on the construction of such database in the next paragraph). The user can optionally expand the selection beyond a single element to its container elements (Figure 2). For instance, by holding down the mouse button (the longer the hold, the larger the selection grows), the user can specify not only individual GUI elements but also compound ones such as the entire dialog boxes and windows.
To construct a database to support image-based help, we can exploit a variety of resources that contain screenshots of various GUIs and helpful text about them. For example, user guides and documentations shipped with a software package almost always include screenshots of major windows and dialog boxes. Tutorials and reviews contributed by zealous early adopters are often embellished with informative screenshots. It is also not uncommon in a forum or discussion board for members to attach screenshots to their postings as they seek assistance regarding various software issues. We can aggregate all these resources into a database and index them using the visual features of these screenshots. This image-based index (as opposed to keyword-based) is what enables users to lookup information about any GUI element simply by using its captured image.

More interesting is the possibility to bootstrap using an existing keyword-based image search engine, instead of building a database from scratch (Figure 3). Once a GUI element is selected by a user, we can identify the tooltip text (e.g., lasso) and application name (e.g., Photoshop) and submit these keywords to a keyword-based image search engine (e.g., Google Image Search). The search engine will return a list of images visually related to the GUI element. Then, we can apply an image matching algorithm to filter these images and to keep only those visually related to the GUI element visually. The final result will be web pages containing not only its screenshot but also informative text about it.

**Potential Benefits**

The image-based method for obtaining help on GUI elements offers unique benefits for both software users and the creators of help creators. For software users, it is a direct and consistent mechanism for retrieving information about any GUI element across multiple resources. Instead of reading the tooltip text and then inferring the right keywords to search for information about a GUI element, users can directly use the image of the element and minimize ambiguities resulted from text-only input. Instead of dealing with inconsistent help mechanisms by different software vendors (e.g., pressing F1, clicking on a light-bulb, asking the paperclip man...etc), users can apply the same search-by-image mechanism regardless of what is built into the software, since screen capture can be performed independently. For help content creators, the proposed framework offers a flexible mechanism for adding contents. Unlike tooltips and captions whose modification requires the access to the source code and can be created only by developers, contents for image-based help can be created by anyone for any GUI element, simply by taking a screenshot and adding some text to it. For instance, any computer book publisher can easily adopt our proposed framework by indexing the contents of their books by the screenshots contained in them.

**Preliminary Experiment**

For the proposed framework to be feasible, we need an algorithm for matching images accurately and efficiently. There has been significant advancement in image matching algorithm in the computer vision community, such as the algorithm developed by [1] for retrieving building images. We performed a preliminary experiment to test the capability of this algorithm in dealing with screenshots. The pilot dataset we used consisted of 182 screenshots of all possible windows and dialog boxes of 4 applications on Windows XP (i.e, Local Area Connection, Internet Properties, New Connection Wizard, and VPN Client). The rank-1 retrieval performance is almost 100% (181/182).

**Future Works**

The next step on the technical front is to scale up the experiment to a more challenging scale this algorithm has been reported to be capable of handling (i.e., millions of images) and to develop specific algorithms to deal with the variations unique to screenshots, such as variations in skins, resolutions, and languages. On the usability front we plan to build a working prototype and design user studies to evaluate it.

**REFERENCES**