• Information Integration and Informatics (III)

Recent years have seen massive growth in the scale, diversity, and complexity of data. Moreover, the data are often used in unanticipated and new ways that frequently require repurposing, transforming, and/or integrating multiple, uncoordinated, and sometimes variously restricted data sources over which data users have no control. The abundance and heterogeneity of data and data sources have created increasing demands on and opportunities for information technologies.

The Information Integration and Informatics (III) program focuses on the processes and technologies involved in creating, managing, visualizing, and understanding diverse digital content in circumstances ranging from individuals through groups, organizations, and societies, and from individual devices to globally-distributed systems. Further, data are only part of a “knowledge life cycle” that progresses from data through knowledge and insight and, ultimately, to action. III funds innovative information technology research that can transform **all stages of the knowledge life cycle**.

III-funded projects are expected to lead to advances that are driven by specific i**nformation-technology challenges**. Projects directed mainly at data-collection building and use, that apply existing data technologies to (perhaps) novel data sets, or that propose other activities with limited computing and information technology research potential are not appropriate for this program. III-supported activities can range from theoretical investigations to projects grounded in **multi-disciplinary collaborations** where data are central to the III-area research. In the case of multi-disciplinary projects proposers should explain the **utility of the proposed work to the application domain** and demonstrate **expertise in that domain** among the project participants. Regardless of research modality, proposals should make clear what computing and information technology challenges are being addressed and how the effectiveness of the work will be assessed

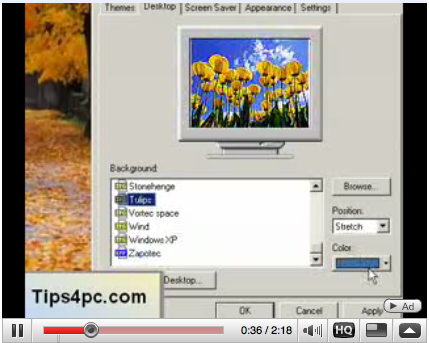
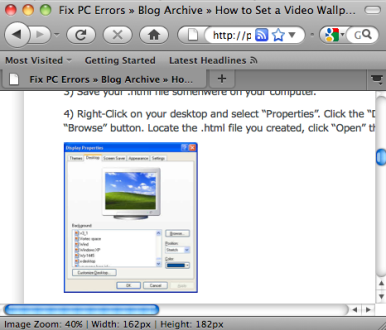
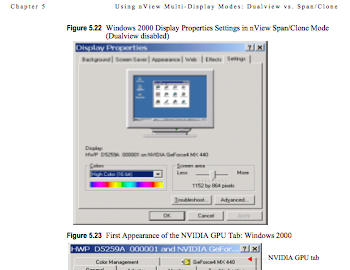
# III: Small: Searching Online Screenshot Instructions using Text and Images

Learning never ceases. In the increasingly digitalized world, we frequently encounter novel tasks we need to perform using a computer application but do not know how. Every such encounter is a learning opportunity. For example, we may want to learn how to set up a laptop to access a virtual private network or how to configure a printer to print in the duplex mode. Sometimes it may be the first time we need to perform the task. But often we may have done the same task a while ago and simply have trouble remembering the details. In both cases, we would like to find relevant step-by-step instructions so that we can acquire or refresh the knowledge needed to perform the task at hand.

Knowledge about computer applications goes through a similar four-stage life cycle identified by Birkinshaw and Sheehan (2002) as it becomes accessible to more and more people. These four stages are creation, mobilization, diffusion and commoditization. In the *creation* stage, knowledge about a computer application is implicitly conceived as the application is being designed and implemented. Knowledge in this stage is often volatile since the application may still undergo several design iterations before it is released. In the *mobilization* stage, as the application is getting closer to be released, the developers may begin to explicitly codify the knowledge by writing manuals that provide instructions explaining each task that can be performed using the application. While knowledge in this stage takes on a concrete form, its content is often controlled by developers and made available only to a trusted community typically consisting of beta testers and early adopters. In the *diffusion* stage, as more users begin to use the application, they may begin to share knowledge about the application over the Internet via blogs, forums, or social networks. Knowledge in this stage propagates at a much accelerated pace since it can be created by and is available to anyone. Finally, in the *commoditization* stage, the knowledge about the application may be so thoroughly diffused that it has become common sense. At this stage, many users no longer need to rely on instructions when they try to perform tasks using the application.

One of the most effective forms of codified computer knowledge is **online screenshot instruction**, which we define as any webpage or web video where the textual contents are instructions and visual contents are screenshots. Figure 1 shows some examples of online screenshot instructions teaching users how to configure display properties. Instructions containing screenshots have been found to help users learn and perform task faster than do text-only instructions (Harrison). Compared to printed forms, online screenshot instructions are more comprehensive and accessible since they can be published by anyone about any application and read by anyone. Software developers regularly publish screenshot instructions about their software products to complement built-in documentation. Institutions such as companies and schools often post screenshot instructions to teach their members how to perform important tasks ranging from installing anti-virus software to configuring email clients. Perhaps the most prolific creators of screenshot instructions are altruistic expert volunteers who frequently offer tutorials and share tips about software applications on forums or blogs for the benefits of all.

Not only do screenshot instructions help users in general, but also they provide stronger benefits to children and seniors in particular. For example, the International Children’s Digital Library (<http://www.childrenslibrary.org/>) gives screenshot instructions to teach children how to use various features on its site such as searching for a book with screenshots. The senior health website created by the National Institutes of Health (<http://nihseniorhealth.gov/>) offers screenshot instructions as both webpages and videos to teach seniors how to find health-related information on the site. Screenshots can also cross language barriers. It is possible to follow each step illustrated in the instruction to press the series of correct buttons to perform a task.



**Figure 1: Examples of online screenshot instructions in a pdf file (left), a webpage (center) and an online video (right).**

# Needs for Research

We estimate that screenshot instructions available online are numbered in tens of millions. These screenshot instructions represent a tremendous amount of collective knowledge about computer applications. It can be a great research opportunity on how to harness such knowledge. However, existing information systems, in particular text-based search engines, pose several limitations that impede the mobilization and the diffusion in the knowledge life cycle of screenshot instructions. We have identified five such limitations as the motivating factors for the needs for research:

**1. Screenshot instructions are scattered all over the Web.** To the best of our knowledge, there has not been a systematic effort to aggregate online screenshot instructions and build a centralized index. As a result, finding relevant screenshot instructions can be a frustrating experience for many users. Users may need to visit several technical support sites and browse through the directory before they can find what they want. Even if they use a search engine, they may retrieve many useless pages that do not offer any instruction and need to painstakingly sift through them. Therefore, there is a need for research on how to construct a centralized, searchable knowledge repository of screenshot instructions.

**2. Keyword queries are difficult to formulate.** To search screenshot instructions using a conventional keyword-based search engine, users must come up with the right keywords to indicate both the context (which screenshot) and topic (what instruction) they desire to find in a webpage or a video. The context such as the type of operating system, the name of the application, and the name of the window may require several words to clearly specify. The topic may require even more words to describe. The result is a long list of words, which is not only mentally and physically tedious for users to generate and type but also prone to ambiguity (e.g., does the word “setup” refer to a program or a topical action?). Therefore, there is a need for research for a better querying method to reduce users’ input efforts.

**3. Text search does not capture screenshot relevance.** Current full-text approach to document search relies on words as the primary key to index documents. However, since screenshot instructions contain both images and text, using only words as primary keys will only capture the text content partially. A retrieved document that matches the text queries perfectly may not necessarily contain the right screenshot. Therefore, there is a need for research to identify suitable visual features to index screenshot instructions.

**4. Text excerpts in search results are not informative.** A typical search result returned by a search engine is a list of links accompanied by short excerpts. As users browse through the list, they rely on these excerpts to decide which links are likely to be relevant with respect to their search objectives and worth exploring. However, current excerpts are text-oriented and provide no evidence regarding the visual relevancy of the search results. It can be difficult for users to judge whether some results are really relevant. Therefore, there is a need for research for methods to generate more informative excerpts to simplify relevancy judgment.

**5. Following instructions while interacting with a program is difficult.** As users try to follow the steps outlined in an instruction, they often need to switch back and forth between the application and the browser periodically, which can be a tedious exercise. The experience can be even more frustrating for videos because of the constant need to pause and play the video between steps. Moreover, as users perform an action and see the next screen, there is no automatic way to scroll to the part of the page or fast-forward to the segment in the video corresponding to that screen. Because of these deficiencies, even if we can build a search engine to retrieve relevant screenshot instructions for users, the benefits cannot be fully realized. Therefore, there is a need for research to provide effective supports for viewing screenshot instructions.

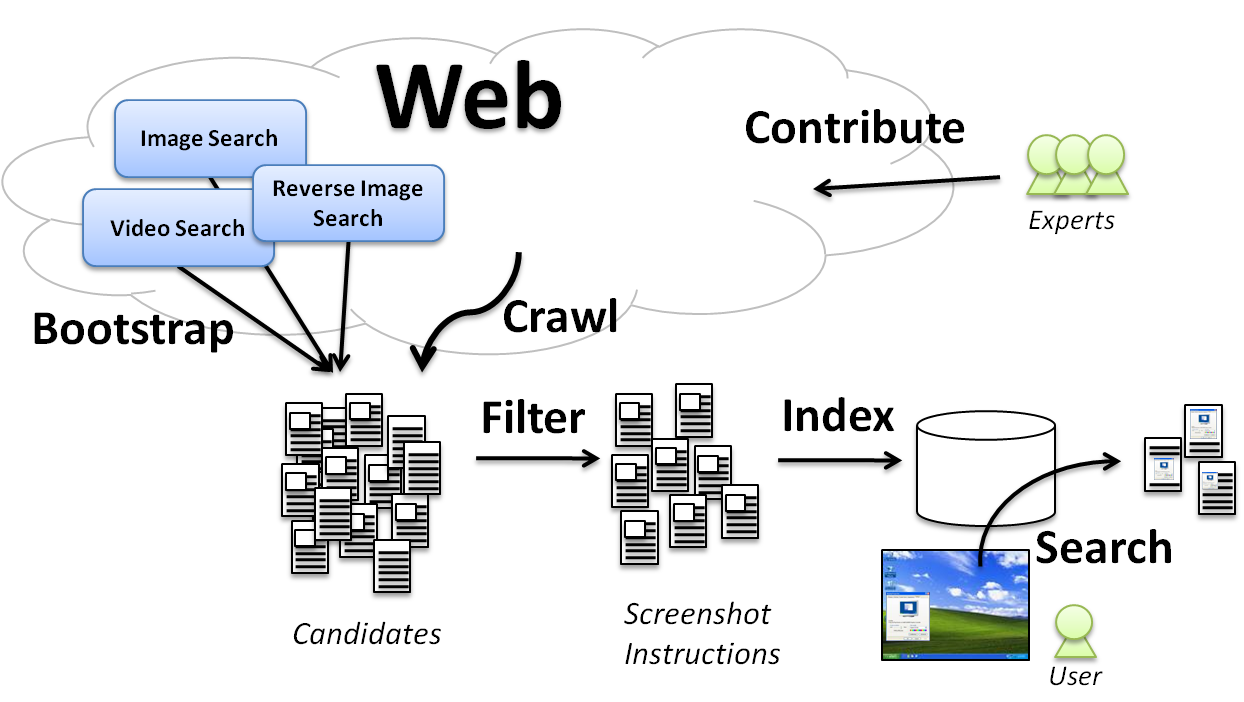
# Main Research Activities

We will focus on the process and the technology involved in the knowledge life cycle of screenshot instructions. Our research goals are:

1. To simplify the **process** in which users acquire knowledge from online screenshot instruction, and
2. To build the **technology** necessary to implement the simplified process.

## Simplifying the Knowledge Acquisition Process

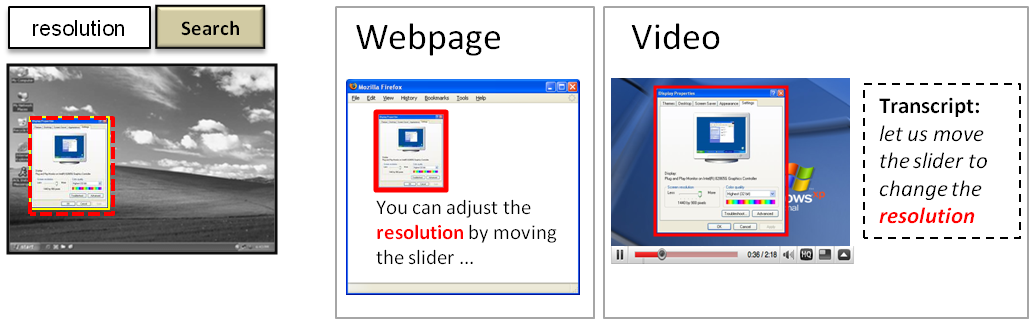
Given a novel task, a user typically goes through a three-step process to search for a relevant online screenshot instruction and learn from the instruction how to perform the task. These three-step process includes (1) *query formulation*, where the user formulates a query and submits it to the system, (2) *result reviewing*, where the user reviews a list of candidate instructions and selects the most relevant one, and (3) *learning*, where the user reads and follows the selected instruction while using the program to accomplish the task. We will propose research activities to improve each step in this process.

**Figure 2: Examples of online screenshot instructions in a pdf file (left), a webpage (middle) and an online video (right).**

### Simplifying Query Formulation

We propose to improve query formulation by allowing users to formulate a *hybrid* query by incorporating a screenshot as an additional search term (See Figure 2). Users can capture the screenshot of an arbitrary application window simply by stretching out a rectangle around it on the desktop screen. The captured screenshot can be submitted along with keywords to a search engine. This hybrid query including both a screenshot and keywords can provide several usability advantages over keyword-only queries:

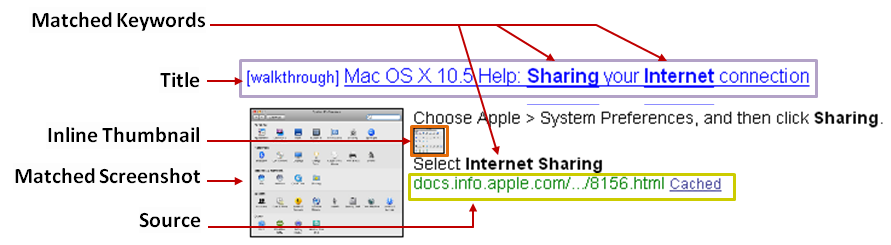
* **Less input effort.** Since many of the identifying features such as the title and the visual layout are contained in a screenshot, users no longer need to enter keywords to describe the application to identify the context. Users only need to enter keywords to describe the desired topic, which requires less input effort.
* **High learnability.** The standard input method of specifying a capturing region is analogous to drawing a rectangle in a typical graphic editor. This method is already familiar to many users and is easy to learn. In contrast, learning what keywords are needed to adequately describe an application for the purpose of searching its screenshots can be more difficult.
* **Consistency across applications and platform.** Screen capturing is a common feature supported by almost every operating system. Also, screenshots are simply pixel data that can be obtained directly from the screen buffer without interfacing the underlying application. Thus, users can apply this method consistently regardless of the application and the operating system.
* **Context-topic separation.** Users will be able to use a screenshot to indicate the context and keywords to specify the desired topic. They do not need to worry about the ambiguity resulted from using only keywords to describe both the context and the topic.

**Figure 3:** Query formulation can be made easier by enabling users to take a screenshot to indicate the context and enter keywords to specify the desired topic.

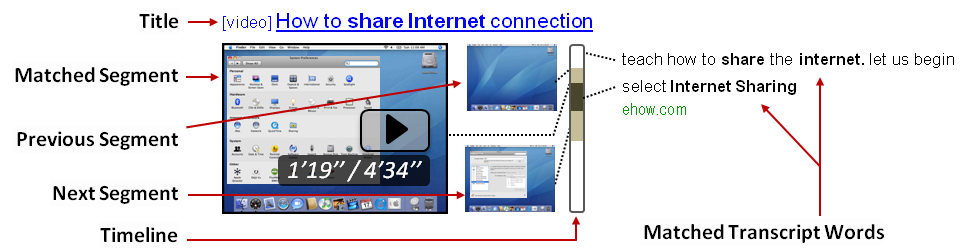
### Simplifying Result Reviewing

Suppose a search engine is able to retrieve a list of candidates matching the query given by a user. The user needs to review the list and identify those candidates that are indeed relevant to the query. An excerpt supplied by the search engine for each candidate is what the user relies on to make the identification.

We propose to improve the process of result reviewing by including screenshots in the excerpts. Since a query consists of a screenshot and keywords, an informative excerpt must be able to convey its relevancy to the query both visually and textually. Below shows a possible design of an informative excerpt for a web page result.



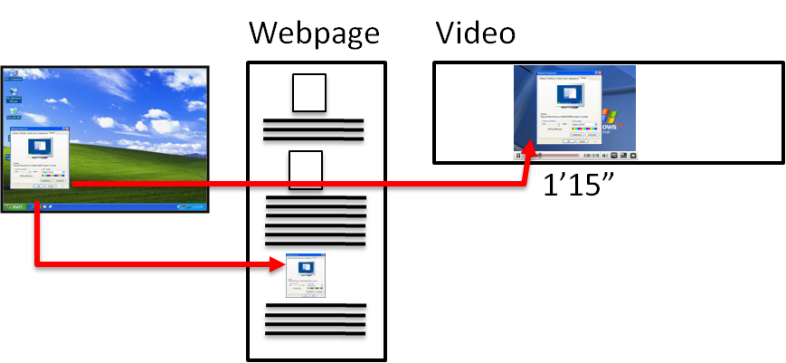
In this design, a thumbnail of the matched screenshot on the page is displayed for users to check if the matched screenshot is correct. The title and the text snippets surrounding highlighted keywords are shown for users to ensure the keywords are also matched. Finally, a tiny thumbnail of the matched screenshot is shown for users to visualize its layout relationship with the surrounding text snippets.



For a video search result, users also rely on preview to judge the relevancy of a candidate video before actually playing the video. Users may want to make sure the video not only contains the right screenshot but also mentions the desired topic. Thus, we will show the location of the matched screenshot in the video as the evidence of the presence of the right screenshot. We will highlight the occurrences of the topic words in the transcript. Moreover, to relate the matched words and screenshots, we will provide visualization to show whether the words are mentioned during, before, or after the matched screenshot.

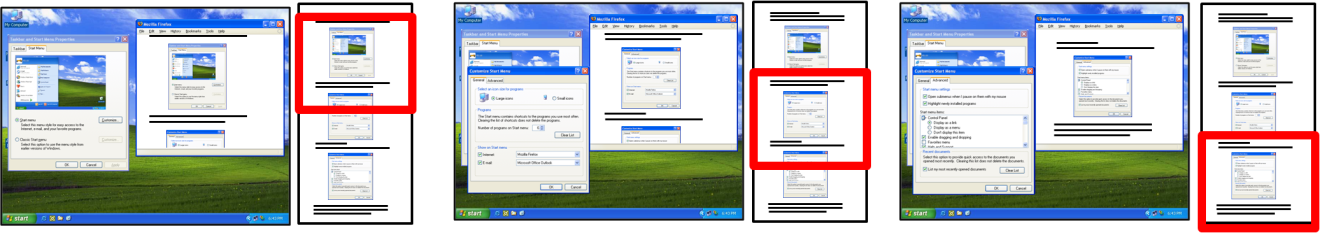
### Simplifying Learning

After reviewing the candidate list and identifying a relevant screenshot instruction, the user is now ready to follow the instruction and learn how to perform the task at hand. We propose to improve how users follow and learn from screenshot instructions by introducing two enhancements to current viewing tools:

****

**Figure 5: Visual seeking.**

* **Visual seeking** When users open an instruction page in a web browser or load an instruction video in a player, there may be multiple screenshots. Visual seeking allows users to find the relevant screenshot and advance to the screenshot automatically. To support this feature, we will pre-compact a local searchable index of screenshots for each page. If the number of screenshot is small, this index can be a forward index using linear search. Otherwise, the same inverted index scheme used for the global index will be applied for efficiency.



**Figure 5: Synchronized viewing**

* **Synchronized viewing** When using an application while following an instruction, synchronized viewing allows users to automatically scroll a page or advance a video to the next screenshot as users move to the next application screen. This feature will eliminate the need to switch back and forth between the application and the browser and the need to periodically pause the video player. To support synchronized viewing, we will develop an algorithm to automatically detect transitions between application screens. One a transition is detected, a screenshot of the next screen will be automatically sent to the server to find the next screenshot in the same page or the same video. Then, the enhanced viewer will automatically advance the content to the next screenshot.

### Designing for Seniors

The design improvements proposed above are mainly geared toward general users. However, seniors have needs and abilities different from the general user population. It is necessary to design a specialized version of the interface in order to accommodate their unique needs and abilities.

To design a specialized interface for seniors, we will involve the Old Adult Team (OAT) in a participatory design process. OAT is a group of seniors recruited from the computer training courses we have been giving at a local library. The original purpose of these training courses is to understand how older adults learn and use computers and the Internet. The creation of OAT is motivated by our desire to involve users early in the design process. We have invited advanced students who have completed the basic computer training courses to join the team. Currently, OAT consists of X seniors aged between A and B. Most of them are African Americans living in Prince George County, Maryland.

The goal of the participatory design is to understand the needs and abilities of seniors and to form a set of design requirements. In terms of needs, we will characterize the types of tasks seniors often need to perform. In terms of abilities, we will identify the cognitive and physical difficulties they encounter using the current search engines as well as using the standard version of our proposed interface. Once we have obtained these design requirements from OAT, we will adapt the interface accordingly and evaluate the produce with OAT.

## Addressing Technical Challenges

The proposed design improvements to simplify the process of searching online screenshot instructions pose two technical challenges: (1) how can we aggregate online screenshot instructions into a centralized collection, and (2) how can we index and search the collection by both visual and textual features in order to support hybrid queries. We will outline our research plan to address these two challenges.

### Aggregating Online Screenshot Instructions

We will address the challenge of aggregating online screenshot instructions in three steps. First, we will use existing search engines as bootstraps to collect a large pool of candidate web pages or videos that are likely to contain screenshot instructions. Then, we will develop an algorithm to determine whether each candidate is indeed a screenshot instruction by analyzing its content. Finally, given a set of screenshot instructions found, we will systematically crawl their source sites in order to collect other screenshot instructions hosted on the same sites.

In the first step, we will use three types of search engines as bootstraps:

* **Image Search** An image search engine such as Google Image Search (<http://images.google.com>) takes a set of words as search terms and returns a list of links to online images discovered on pages containing these words. To use an image search engine to collect candidate pages, we will manually compile a keyword list consisting of words sampled from the title bars of various computer application windows. Some examples of these keywords can be properties, preferences, option, settings, wizard, installation, network, sound, and keyboard. We will append to the list additional keywords that are commonly found in computer instructions such as tutorial, guide, and manual. Then, we will systematically submit different combinations of the keywords in the list to an image search engine. The rationale is that these keywords are likely to retrieve screenshot images of a variety of application windows. The pages containing these screenshots are likely to provide useful computer instructions related to these application windows, thus our candidates.
* **Video Search** A video search engine such as Google Video Search (<http://videos.google.com>) in principle works like an image search engine. It takes keywords as search terms and retrieve links to online videos that may be titled or tagged with related keywords. We will use the same set of keywords as above to retrieve candidates of videos offering computer instructions.
* **Reverse Image Search** A reverse image search engine such as TinEye (<http://tineye.com>) takes an image as the query and returns a list of links to online images visually similar to the query image. To use a reverse image search engine to collect candidate pages, we will create a list of query images by manually capturing the screenshots of a wide variety of common application windows across popular OS platforms (XP, Vista, and MacOS). We will submit each screenshot as the query to a reverse image search engine to retrieve a list of similar screenshot images and links to their source pages. These pages will be the candidate pages.

Next, once a significant number of candidates are obtained using existing search engines, we need to examine each candidate and remove false positives. While an image or video search engines can retrieve candidates containing textually relevant images, there is no guarantee every image is a computer screenshot. For example, the word “display properties” can retrieve screenshots of the display properties window as well as images of properties displayed by realtors. On the other hand, given screenshots as queries, a reverse image search engine can almost guarantee every retrieved candidate must contain a screenshot. But there is no guarantee the screenshot is part of an instruction; it can be from a software catalog.

To remove false positives, we will develop an algorithm that combines computer vision and text analysis. Given a candidate web page or video as input, the goal of the algorithm is to determine whether the text content is about instructions and whether the visual content contains computer screenshots. To determine whether the text is about computer instructions, we will consider word frequency features and build topical models. To determine whether the image is a screenshot, we will consider visual features such as the size, shape, color, and texture of the image and the presence of salient visual components such as title bars, buttons, and checkboxes. In addition, we will consider the structural relationship between textual and visual features. For example, if a web page exhibits the pattern of short sentences interspersed with screenshots, it can be a sign that the page contains step-by-step instructions. We will manually label a large sample of candidates as training examples and use them to train binary classifiers such as SVM.

Finally, after obtaining sizable and filtered collection of screenshot instructions, we will further expand the collection by crawling. We will treat the source sites of the screenshot instructions currently found and use a spider to systematically visit all the links in these seed sites in order to discover other screenshot instructions.

### Indexing and Searching by Visual and Textual Features

After aggregating a large collection of online screenshot instructions, we will need to deal with the technical challenge of actually indexing and searching them using both visual and textual features. We plan to proceed in two steps. First, we will identify suitable visual and textual features as primary keys for building an index offline. Second, we will learn how to incorporate other auxiliary features to rank each result by its relevancy to a query online.

A type of feature suitable for the primary key typically has the following properties: discriminative, robust to variations, can be quantized, and can be efficiently searched by similarity. We will experiment with four types of features that possess these properties for indexing screenshot instructions.

* **Word.** Many techniques have been established to use words effectively as primary keys for text document retrieval. For example, to make words discriminative, we can use a stop list to ignore common words. To make words robust to variations such as inflection, we can compute stemmed words. Words are naturally quantized and can be efficiently compared to find matching words.
* **Visual Word.** To index computer screenshots, we will need visual features with properties similar to words. Recently, many features have been proposed in the computer vision literature that can act as “visual words” for large-scale image retrieval applications. One notable example is the SIFT feature (Lowe). SIFT features are extracted from salient regions in an image and have been shown to be discriminative for the image in many visual analysis tasks such as object recognition. SIFT features are robust to variations in scale, translation, and rotation. For screenshots, robustness to scale and translation is particularly important because screenshots tend to be resized or cropped. Several clustering techniques such as K-mean clustering have been applied successfully to quantize SIFT features (Zisserman). Moreover, several efficient data-structures have been proposed to store SIFT features to allow fast lookup of matching features (Nister).
* **OCR Word.** In a computer screenshot, text can often be found in the title bar or the button of the application window in the screenshot. It is possible to extract such text using an optical character recognition (OCR) program and compute stemmed words as the primary key to index the screenshot. However, current OCR systems are optimized for analyzing text documents. We have found that their recognition accuracy is much lower when applied to screenshots because of the large amount of non-textual features such as icons and boxes also present in the screenshots. Therefore, instead of using raw strings extracted by OCR, we will compute 3-grams from the characters in these strings. For example, the word “system” might be incorrectly recognized as “system”. But when represented as a set of 3-grams over characters, these two terms are {sys, yst, ste, tem} and {sys, yst, ste, ten} respectively, which results in a 75% match, rather than a complete mismatch. These 3-grams can help discriminate the screenshots while providing robustness to variations caused by OCR errors. Like regular words, these 3-grams can be easily quantized and efficiently retrieved. Moreover, the recognition accuracy can be further improved by exploiting cross-modality redundancy. For example, we can give higher weights to the OCR words that also appear as regular words in the text.
* **Speech Word.** Since videos often have voice commentary, this commentary can be analyzed to extract the words spoken in the commentary. Given the inherent unreliability of speech recognition, we can take an approach similar to the one above for OCR words to generate primary keys.

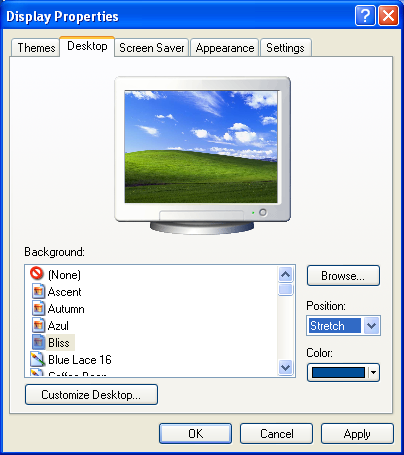
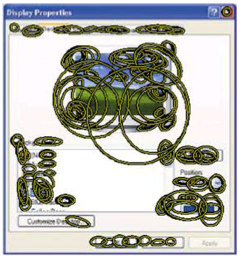
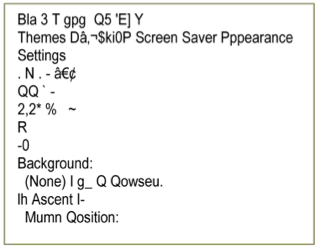
**  **

Figure 6: A typical screenshot (left) and the visual words (middle) and OCR words (right) extracted from it.

Given a hybrid query that consists of a screenshot and keywords, we can compute the four types of words as described above and lookup those documents or videos that share these words. Since there can be multiple documents or videos sharing the same words, it is important to rank them before presenting them to users.

Ranking must take into account both textual and visual relevancy of the query. Text-based search engines rank results only by textual relevancy to query keywords. On the other hand, content-based image retrieval systems rank results by visual relevancy to query images. However, neither ranking scheme is suitable here. The former may return instructions with the wrong screenshots, while the latter may find the right screenshots but on a wrong topic.

We identify the following auxiliary features that may be important for ranking:

* **Text features:** the number of matched keywords, the number of action words (e.g., click, open), and the number of steps.
* **Image features:** the similarity value of the matching screenshot, the size ratio between the query and the matching screenshot.
* **Video features:** the total length of the video, the number of matching video segments, and the length of each matching video segment.
* **Multimodal features:** the number of words per screenshot, the number of lines between the matching keyword and the matching screenshot, and the spatial relationship between words and screenshots (i.e., linear or grid).
* **Page features:** the number of words on the page, the number of other screenshots on the page.
* **Site features:** the number of other computer instructions hosted on the site, the popularity of the site.

While each of the features identified above may play a role in how users judge the relevancy of a result, we do not know which features are more important than the others. We will need to learn how to assign weights to features to reflect their relative importance. To learn these weights, we will apply the RankSVM technique developed by Jochims (2002). This learning technique is able to learn feature weights from a set of subjective ordering constraints inferred from user click-through data. Since we do not have any click-through data initially, we will recruit annotators to provide us with subjective ratings of unranked results for a set of test queries. The results will be shuffled so that the annotators would not be influenced by the ordering bias. A list of ordering constraints can be inferred from the rating data. These constraints than can be provided as input to the RankSVM learner to obtain a set of weights needed for ranking search results for our application.

# Supporting Research Activities

## Understanding the Knowledge Life Cycle

How do people create screenshot instructions? We will carry out a longitudinal study by following the knowledge life cycle of a particular software application. We will pay attention to the release of major software. We will monitor the Web and see when screenshot instructions about the application first start to appear. We will estimate the rate at which the screenshot instructions are created, which will give us a rough picture of the pace knowledge may diffuse. We will analyze the access counters (statistics) of a sample of the online screenshot instructions and estimate the users behaviors and see if we can see a tend.

## Linking Online Screenshot Instructions directly to Applications

We want to create a more open platform to encourage more people to contribute online screenshot instructions. There have been systems for contributing knowledge to supplement existing material are common on the web (e.g. WebNotes and Shiftspace), where URLs and HTML page structure provide robust attachment points for users to link any article or video they generate to an arbitrary web page. But similar systems for the desktop have previously required a formal API support or the access to the source code. Many would-be contributors are discouraged because they either do not have source code access or do not possess the necessary programming expertise. As a result, most screenshot instructions created by third-party users are published on the Web separated from the applications. We will explore the use of screenshots as hooks to link applications directly to instructions. People who wish to contribute instructions to an arbitrary application can simply take a screenshot of the application and type some brief instructional text. The screenshot-text pair can be uploaded to our system to be indexed and made available to anyone. On the users’ side, we will build a tool that listens for hotkeys such as F1. Whenever the hotkey is pressed, the tool will automatically capture the screenshot of the most salient application screen on the desktop and lookup the instructions relevant to the application screen.

# Deliverables

At the conclusion of the project, we will contribute the following deliverables:

* A search engine that supports hybrid queries for searching online screenshot instructions.
* A plug-in for web browsers and video players to provide visual seeking and synchronized browsing capabilities for viewing screenshot instructions.
* A specialized interface for seniors.
* An algorithm for detecting screenshot instructions.
* An algorithm for ranking screenshot instructions

# Evaluation

Each deliverable will undergo rigorous empirical evaluation. Wherever appropriate, comparative evaluation will be carried out against baseline systems including keyword-only and image-only search engines. Below we list the empirical questions we want to address and hypotheses we seek to test.

* **Hybrid queries.** How much simpler it is to specify hybrid queries for searching online screenshot instructions? Our hypotheses are (1) users will spend less time specifying queries, (2) users will enter fewer keywords, (3) users will enjoy higher success finding relevant screenshot instructions, and (4) users will favor hybrid queries over keyword queries in typical usability measures such as ease-of-use and learnability.
* **Enhanced viewer.** How much can our enhanced viewer improve users’ experience in following screenshot instructions? Our hypotheses are (1) users will spend less time following instructions to complete a task, (2) users will make fewer errors, and (3) users will prefer viewers with enhanced features to standard viewers.
* **Specialized interface for seniors.** How much can the specialized interface help seniors? Our hypothesis is that in all the usability metrics we measure, the specialized interface will achieve higher scores compared to those achieved by the standard interface.
* **Screenshot instruction detection algorithm.** What is the accuracy of our algorithm in detecting pages or videos that are screenshot instructions? Our hypothesis is that using both visual and textual features will achieve the highest accuracy.
* **Ranking algorithm.** What is the effectiveness of the ranking algorithm? Our hypothesis is that our algorithm will produce a ranking judged more favorably by users than those produced by baseline approaches.

# Appropriateness for the III Program

The proposed research project is appropriate for the Information Integration and Informatics Program for the following reasons:

* This project **focuses on the processes and technologies** involved in creating, managing, visualizing, and understanding a useful class of digital content–online screenshot instructions.
* This project **touches on all stages of the knowledge life cycle** by examining the role of online screenshot instructions in each stage.
* This project **involves multi-disciplinary collaboration** among qualified researchers of information technology, computer vision, and HCI where data (i.e., screenshot instructions) plays a central part.
* This project **is grounded in a useful application domain** that can benefit a diverse group of users in learning various computer-related tasks.

# Intellectual Merits

* We will understand how people use multimodal queries to search multimodal documents when both modalities are essential.
* What are the scientific questions that will be answered?
* TODO

# Broader Impacts

The proposed project has broader impacts in dissemination, diversity, research, and education.

In terms of dissemination, the search engine will be made accessible to anyone. This may potentially impact a broad range of users on the way they access and learn computers from online screenshot instructions. Since it does not require any special software or hardware installation, it can be easily scaled up. For disadvantaged people without personal computers or Internet access, they can visit local libraries to learn computers on their own using our system.

In terms of diversity, the project will make a strong effort to understand the needs of older adults and adapts the system to meet their needs. Many of the older adults involved in the design process are of minority backgrounds.

In terms of research, it will pave the way to study other kinds of graphically illustrated instructions that are not related to computers. Examples are steps to find health information, steps to repair things, and steps to cook.

In terms of education, it will offer graduate students opportunities to conduct multidisciplinary research spanning information science, computer vision, and HCI.

# Preliminary Studies

We have conducted three preliminary studies to evaluate the potential of the proposed project in terms of technological feasibility and usability.

For technological feasibility, we studied the idea using a small collection of screenshot instructions contained in electronic books. We were able to apply standard image-matching techniques to index more than 50,000 pages. A significant number of these pages do contain screenshots and can be considered as examples of screenshot instructions. We found multimodal indexing using regular words, visual words, and OCR words provided the best retrieval performance based on a small test set of 500 queries. We shared the result as a poster at SIGIR 2009. (TODO: Elaborate)

For usability, we recruited 15 subjects to try a Web-based simulator to experience taking screenshot and typing keywords to search for screenshot instructions. All the subjects we studied found the multimodal search method novel and easy to learn. We also measured statically significant reduction in task completion time in formulating multimodal queries than in formulating keyword-only queries. These findings point to the potential usability of a search system fully taking advantage of the multimodal nature of screenshot instructions. We have published the result as a part of a full paper publication at UIST 2009. (TODO: Elaborate)

While these preliminary studies were limited in scale (50,000 pages), in scope (only books), and in diversity (only young adults as subjects) due to resource constraints, they led to encouraging findings. Therefore, we are requesting funding to enable us to expand the project in these three dimensions. In scale, we wish to scale up to tens of millions of screenshot instructions. In scope, we wish to extend from books to web pages and videos. In diversity, we wish to involve seniors and minorities. Funding from III is crucial for fully realizing the potential broader impacts of the proposed project.

# Management Plan

The project team will consist of two co-PI’s, Prof. Larry S. Davis of the Department of Computer Science and Prof. Bo Xie of the College of Information Studies, one senior research personnel, Dr. Tom Yeh, and two graduate students. The makeup of the team reflects the project’s synergetic nature by bringing together researchers of information science and computer vision. With her expertise on information management and computer literacy, Prof. Xie will supervise the research activities aimed to improve the process in which users can access and learn from online screenshot instructions. Also, she will lead the participatory design with the Old Adult Team (OAT). Prof. Davis will supervise the research and development of the necessary technology components, in particular, the computer algorithms for indexing and searching screenshots in web pages and videos. Profs. Xie and Davis will be assisted by Dr. Yeh, who will lead the integration efforts between the two sides given his successful prior involvements in similar multi-disciplinary research collaborations.

We request funding to support two graduate students, one full-time and the other half-time. The full-time graduate student will be jointly supervised by Prof. Davis and Dr. Yeh. He or she will be responsible for researching and developing all the technical components described in the proposal. The part-time graduate student will be supervised by Prof. Xie. The student will devote half of the time to examine all issues related to the usability of the proposed information system. This collaboration will provide both students valuable learning experience conducting multi-disciplinary research.

As a team, we will conduct bi-weekly meeting. Dr. Yeh will organize reading groups on cross-disciplinary topics, attracting students from information science, computer vision, and HCI.

# Work Plan

TODO:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Fall 10** | **Spring 11** | **Fall 11** | **Spring 12** | **Fall 12** | **Spring 13** |
| **Aggregate** | R&D | Evaluate |  |  | Integrate & Deploy | Evaluate & Publish |
| **Index** | R&D | Evaluate |  |  |
| **Query** |  | R&D | Evaluate |  |
| **Rank** |  | R&D | Evaluate |  |
| **Preview** |  |  | R&D | Evaluate |
| **Read** |  |  | R&D | Evaluate |
| **Contribute** |  |  | R&D | Evaluate |