GUIDING AUTOMATION WITH PIXELS: A TECHNIQUE FOR PROGRAMMING IN THE USER INTERFACE

Richard Potter

Human-Computer Interaction Laboratory
Center for Automation Research
University of Maryland
College Park, MD 20742
Tel. (301) 405-2725
potter@cs.umd.edu

INTRODUCTION

Accessing data is a critical challenge for users who write programs to process data already stored in the computer. This *data access challenge* is particularly acute for end-user programming because the users' data often exists in applications like word processors, drawing editors, and spreadsheet applications whose internal workings are unknown to the users. Regardless of how easy their programming system is to use or how skilled they are at using it (whether it be C, PASCAL, keyboard macro, or programming by demonstration), the system is of no use if it can not access the data of interest. This challenge will be all the more frustrating to users when the data is clearly represented on the computer display but cannot be accessed.

This research shows how simple pattern matching on the pixel representations in the computer display buffer can significantly enhance a programming system's ability to access data. Because pixel representations are a low-level representation, the algorithms that make use of this representation are different than what one would expect for a given task. The video shows the operation of Triggers, a programming system developed to demonstrate that these algorithms exist, are understandable, accomplish useful tasks, and can be easily implemented.

Pixel-based data access seems to run counter to the widely acknowledged fact that high-level data and constructs improve programming productivity. Unfortunately, this fact has led to wide spread acceptance of a conclusion that, strictly speaking, does not logically follow: that programming with *low-level* data and operators is inherently inefficient. This research shows that the inherent visual nature of pixels, the high degree of regularity of modern graphical user interfaces, and the user's extensive familiarity with the display content make programming at the pixel level appropriate.

Triggers currently implements only two simple types of pixel-based data access. One is the *specific location match* where Triggers tests for a rectangular pixel pattern at specific location on the computer display. The other is the *pixel pattern search* which locates a rectangular pixel pattern within a rectangular region on the computer display. These two tests are enough to show the usefulness of pixel-based data access for automating meaningful tasks.

THE VIDEO

The video demonstrates how a user can program Triggers to automate the wrapping of a properly sized rounded rectangle around a preexisting text field in an *unmodified copy of MacDraw II*. MacDraw II conveniently places a gray bounding box around a selected field. Pixel pattern searches using pieces of this bounding box as the pattern give enough data access to determine the size and location of the text field. Triggers then simulates a series of keystrokes and mouse actions that create the rounded rectangle. Other examples from graphic and text domains are briefly shown.
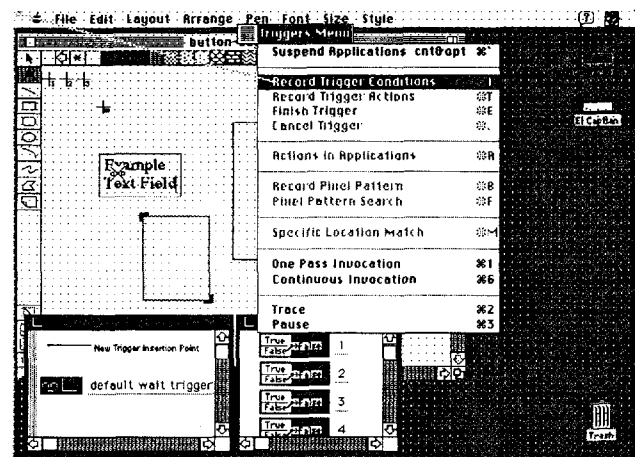
Triggers is part of a larger research effort to create programming systems that allow users to productively weave programming activity into their daily work. Not all data can be easily accessed with such simple pattern matching, but when the opportunity of regularity presents itself, users should be able to take advantage it.