Visionsketch: Gesture-based Language for End-user Computer Vision Programming

Jun Kato <i@junkato.jp>
The University of Tokyo

Problem and Motivation

Programming Language for Interactive Surfaces

An interactive touch surface is getting more and more popular as the primary input source for computers. Meanwhile, there is an increasing demand on the use of complex data types e.g. images. While textual or visual (symbolic) programming languages cannot handle such data nicely, I thought a new language on the surface can construct image processing programs through intuitive direct manipulation.

Background and Related Work

Programming with Visual Representations

Integrating visual representations into development environments has been successfully enhanced the programming experience. I made a new language and its environment from scratch rather than integration. They allow constructing domain-specific programs by direct manipulation just like Morphic (graphical user interface) and Excel (spreadsheet calculation).

Approach and Uniqueness

Gesture-based Language and its Integrated Development Environment

I propose Visionsketch, a gesture-based language where each code element is constructed through gestures on an image/video. With Visionsketch, the programmer can construct an image processing program by drawing lines and shapes instead of typing.

Language primitives

- **Color filter** / shapes: any / out: image
- **Linear-polar conversion** / shapes: a circle / out: image
- **Perspective warp** / shapes: a rectangle / out: image
- **Timelapse conversion** / shapes: a rectangle / out: image
- **Contour counting** / shapes: none / out: image, region count (Other primitives can be easily implemented as Java classes.)

More information available at http://junkato.jp/visionsketch/

Results and Contributions

Melting the Boundary between PL and UI

Since Visionsketch does not need traditional text-based programming, I expected that it can be used by a non-programmer and conducted a preliminary user study with her. The participant could construct a program to count the number of rotations of the coffee beans grinder. While Visionsketch does not cover all the programming language features (e.g. there’s no “if”), it is feasible for the practical use, melting the boundary between Programming Language and User Interface.