An Online Handwriting Recognition Application in XScale

Jim Hong
Jing Zhu
CSE 237A - Fall 2006
Outline

- Motivation
- Types of Recognition
- Implementation
- Results
Motivation

- Mobile devices are becoming increasingly smaller (e.g., cell phones)
- Input with a small keypad is not user-friendly and efficient
- Utilizing a touch screen is more intuitive and easier to use
Types of Text Recognition

- On-line

- Off-line
Implementation

- GUI FrontEnd
- Recognition BackEnd
- Touch Screen
- Compromises
GUI library for ES

- Survey on GUI libraries for ES
  - with which we develop our OHR GUI
  - Qt/Embedded, GTK, MiniGUI, ...
- MiniGUI
  - compact, fast, light-weight
MiniGUI

- **Architecture**

- Abstraction Layer to deal with low level output and input

  - Graphics Abstraction Layer (GAL) - Graphics Engine
    - use frame buffer to display output
    - use MiniGUI’s fbcon graphics engine

  - Input Abstraction Layer (IAL) - Input Engine
    - originally intended to write an input engine for touchscreen
    - turn to Mouse
MiniGUI

- Get MiniGUI to run on host
- Port MiniGUI to XScale
  - Cross-compile for target on host
  - Make MiniGUI libraries NFS accessible on target
- Challenges
  - Multiple running mode
    - MiniGUI Thread, MiniGUI-Lite, MiniGUI-standalone
  - Numerous features
    - Must disable and enable some features to avoid problems with the configuration
Rosetta

- A handwriting recognition engine for the X Window System
- Open-Source
- Derived from Xmerlin (a simple, small footprint, single character recognition engine for X11)
- Recognizes characters by strokes, pixels, and line angles
Rosetta - Learning Process

- Asks user to write a set of letters that will be recognized
- Normalizes input
- Collects data from number of strokes, angles of lines, pixel count
- Creates a stroke database
Rosetta - Recognition Process

- Loads stroke database
- Asks user for written input
- Normalizes input
- Statistically analyze the input against the database to determine the possible recognized characters
- Generate the best recognized characters
Rosetta - Challenges

- Get rid of the dependence on Xlib in Rosetta
  - Rewrite some Xlib functions
    - the conversion between keysym (encoding of a symbol on a keycap on a keyboard) and string
- Integrate Rosetta with MiniGUI
Results

- Port MiniGUI & Rosetta to XScale
- Integrate Rosetta with MiniGUI on XScale
- Decent level of single character recognition with real-time response