Assessing the Impact of using Robots in Education, or: How We Learned to Stop Worrying and Love the Chaos

Douglas Blank and Deepak Kumar
Bryn Mawr College
Institute for Personal Robots in Education
http://RobotEducation.org
Loving the Chaos
Overview

- Institute for Personal Robots in Education (IPRE)
- The Pyjama Project
- Loving the Chaos
- Future Work
Institute for Personal Robots in Education, IPRE

**Research Project**

- **Mission:** explore making CS education more fun and effective through the context of a *personal robot*

- **Goal:** Affect all levels, from middle school to graduate school

- **Initial Target:** CS1

- 3-year seed funding provided by MSR

- Joint effort hosted at Georgia Tech with Bryn Mawr College

- Now a 2-year CCLI funded by NSF

**Special ingredient and hypothesis:**

- A *personal robot for every student*
Personal Robot

- Every student gets their own robot
- Small enough to carry in backpack
- Cost about the price of a textbook
- Wireless, controlled from computer
- Interactive and easy to program
- Personalizable
- More than “just a robot”
IPRE Pilot Hardware Kit
Featuring Parallax’s Scribbler

- 6 Light sensors
- 7 IR sensors
- Stall sensor
- Speaker
- 5 LEDs
- 2 motors
- Bluetooth wireless
- Camera
- Gamepad
Using the Robot

Myro

USB Bluetooth Adapter
Start Python (IDLE)

Start Python.pyw

--- No Subprocess ---
from myro import *
init("com5")
setName("Fluffy")
for i in range(4):
    forward(.75, 3)
    turnLeft(1, .3)
    beep(.1, 440)
    speak("Turning...")
speak(getName() + " is done!")
from myro import *
init("com5")
while timeRemaining(60):
    pic = takePicture()
    sum, count = 0, 0
    for pixel in getPixels(pic):
        if getColor(pixel) == orange:
            sum += getX(pixel)
            count += 1
    if sum/count > getWidth(pic)/2:
        turnRight(1, .2)
    else:
        turnLeft(1, .2)
Available from Amazon.com, Kinkos, and lulu.com

Learning Computing with Robots

$17.95

Myro Software
Free, and open source
Runs on Linux, Mac, and Windows

$199.95
Underlying Motivations

1) Let the needs of the curriculum drive the design of the robot
2) Use tools that are easy to use, scale with experience
3) Treat the robot as a peripheral
4) Create an accessible, engaging environment for a new, diverse population of students
5) Computer Science is not just Programming
6) Make Computing a Social Activity
7) Make computing a medium for creativity
8) Performances vs. competitions
Doin' the Robot

Bryn Mawr CS110 Robot Dance
A CS1 Assignment: Exploring a Pyramid
Programming as a social activity
Connections to Biology and Psychology

Figure 3
Vehicles 2a and 2b in the vicinity of a source (circle with rays emanating from it). Vehicle 2b orients toward the source, 2a away from it.
Peacebots Picket Robotic Violence

What do robots do in the real world? They vacuum floors, work on assembly lines, assist with laparoscopic surgery and, as of last Saturday, march for peace.

The peacebots that demonstrated at the Franklin Institute on Oct. 20 were programmed by four students from Associate Professor of Computer Science Doug Blank’s introductory course in computer science, which uses
Graphics and Objects: Day 1
Vision and Image Processing
Advanced Robotics

- AI
- Neural Networks
- Vision
- Mapping
- Maze Following
Games and Robots

YouTube game videos available at cs.brynmawr.edu/games
Music and Robots

Sample music available at cs.brynmawr.edu/music

Saxophone
A saxophone or similar wind instrument (Saxofony in ChucK)

- `setStiffness(stiffness)`: set reed stiffness (0.0 <= stiffness <= 1.0)
- `setAperture(aperture)`: set reed aperture (0.0 <= aperture <= 1.0)
- `setPressure(pressure)`: set pressure / volume (0.0 <= pressure <= 1.0)
- `setVibrato(vibratoFreq, vibratoGain, noiseGain)`: set frequency and gain
  (0.0 <= vibratoFreq <= 0.0, 0.0 <= vibratoGain <= 1.0, 0.0 <= noiseGain <= 1.0)
- `setBellPosition(position)`: set blow position / lip stiffness (0.0 <= position)
- `startBlowing(strength)`: start blowing (0.0 <= strength <= 1.0)
- `stopBlowing(strength)`: stop blowing (0.0 <= strength <= 1.0)
- `setAttackRate(seconds)`: set rate of attack (sound’s beginning) in seconds

MoogSynthesizer
A Moog synthesizer (Moog in ChucK)

- `setFilterQ(floatValue)`: set filter’s Q value (0.0 <= floatValue <= 1.0)
- `setFilterSweepRate(rate)`: set filter sweep rate (0.0 <= rate <= 1.0)
- `setVibrato(freq, gain)`: set frequency and gain of vibrato (freq in Hertz, 0.0 <=
  freq <= 500, 0.0 <= gain <= 1.0)
- `setAfterTouch(afterTouch)`: set aftertouch (0.0 <= afterTouch <= 1.0)

Orchestrating
After you get familiar with a single instrument, then you migl

```
from myro import *
from myro.chuck import *

initChuck()

def playSaxophone():
    sax = Saxophone()
    sax.connect()
    sax.startBlowing(1)
    wait(1)
    sax.stopBlowing(1)

def playMandolin():
    mandolin = Mandolin()
    mandolin.connect()
    mandolinpluck(1)
    wait(1)

You can test each one of those independently by simply run

playSaxophone()

doTogether(playSaxophone, playMandolin)
```

StruckBar
Struck bar instruments (ModalBar in ChucK)
Contexts

- Robot is used for 8 weeks
- Music composition and performances (ChucK)
- Pixel-based media computation (Guzdial)
- Object-based graphics (Zelle)
- Game creation (Myro and Pygame)
Analysis

• Learned CS concepts through robots
• Robots made learning experience more hands-on, tangible, and exciting
• Most frustrating parts were dealing with robot hardware inconsistencies
• Viewed CS as a type of logic and problem solving; requiring patience & thought
• Discovered that CS and robots are applicable to the real world
Results

- Doubled the numbers of students taking the introductory course (5% to 10%)
- Over 1,000 students have taken variations of the course
- Numbers of students in upper-level courses has increased
- Python, Scheme, and C++ Languages
- Free textbook in English and Spanish
What's next?

• A new robot, called Gyro
  – Small, inexpensive
  – Play audio files from robot
  – Microphones
  – More expressive

• Pyjama Project
  – Rearchitecting of software
  – IDE
  – More languages
Problems in CS Education

• Some very nice environments (JES, DrScheme, DrJava, Alice, Scratch, etc.)
  – But they are language specific
• Some very interesting libraries (media, graphics, robotics, 3D, games, etc.)
  – But they are language specific
• We need a nice environment without having to replicate the functionality of these libraries for each language
Pyjama IDE

- Works with a variety of languages
- Libraries appear as native to current language
- Open source with plugins
Dynamic Language Runtime

- Open source project from Microsoft
- Built on top of the Common Language Runtime (CLR)
- DLR is what Java never had
  - Virtual Machine
  - Java Syntax
  - Compiler
  - 100's of languages, no sharing
The Pyjama Project

- Infrastructure for CS Education
- Change languages on the fly
- Can mix languages together
  - Python can call Ruby for example
- Write library once for all languages
- Plugins allow assessment and variations
Stop Worrying; Love the Chaos

- Grew out of the frustrations of trying to teach good software engineering practices in CS1
  - Commenting code, pre-conditions, variable naming, proper looping, etc.
- Students weren't ready for structure yet
- We weren't building on their knowledge
- Alienated students
Skills on which to Build

- Typing
- Text Editing
- Writing and Revisions
- Copy and paste
- Simple Recursion
def robotDance():
    # turn left at full for .3 seconds
    turnLeft(1, .3)
    # go forward at full for 2 seconds
    forward(1, 2)

robotDance()
robotDance()
robotDance()
robotDance()
On Proper Looping

- Have your robot perform a dance
- Ask the user if they would like to see another dance
- If yes, then repeat
done = False
while not done:
    robotDance()
    response = ask("Want to see the robot dance again?", ["Yes", "No"])
    if response == "No":
        done = True
def main():
    robotDance()
    response = ask("Want to see the robot
dance again?", ["Yes", "No"])
    if response == "Yes":
        main()
The Painter's Workshop
Medium versus the Message

flickr.com/photos/8166472@N03/2443176806/
Sapnya Cheryan of the University of Washington exposed male and female students to computer science workplaces that were decorated in both conventional and "stereotypical male" ways (Star trek posters, video game boxes and Coke cans). Women who saw the stereotypical workplaces had a lower opinion of the profession of computer science.

-Journal of Personality and Social Psychology, 97, 6
Adjusted Goals for CS1

- Creativity drives problem solving
  - Introducing concepts too early is confusing; let them need before they get lectured
- Importance of Debugging
- Show many contexts
- Attract, Retain, and Propel
- Stop worrying; love the chaos
Future Work

- Software: Release Pyjama
  - Python, Ruby, Scheme
  - A block-diagram language called "Wires"
- Curriculum: Portfolio-based Introduction to CS (Kumar and Xu, forthcoming)
- Hardware: Develop Gyro
Gyro Personal Robot

turnLeft(.5)
speak("Hello, AAAI Spring Symposium!")
playMusic("madonna.wav")
setFace("smile")
takePicture()
penDown("red")
This work was funded in part by NSF-0920539, Personal Robots for CS1: Next Steps for an Engaging Pedagogical Framework.
Loving the Chaos

- How to avoid bait and switch?
- How does this effect the rest of the curriculum?
- How do you grade open-ended assignments?