
Tufts University

Department of Computer Science



Comp 97: Senior Capstone (Fall)

InterLACE Worksheet Functional Design Document

Problem and Motivation (2)
Existing Solutions (2)
Sponsor's Goals (2)
User Experience Overview (3)
Goals (3)
Major Technical Components (5)
Evaluation (6)

Sam Heilbron, Jack Spiva, Samantha Welch

Sponsor: Dr. Ethan Danahy

Thursday, December 22, 2016

Samuel Guyer

Problem and Motivation

Many teachers continue to use paper worksheets because of how simple and straightforward they are. Class computers or tablets are often much more time consuming to set up and often end up wasting a lot more class time. However, teachers have no easy way to view and evaluate students' work on worksheets besides just going through them one by one. On top of this, worksheets in the classroom often offer no opportunity for collaborative work and students typically never see the work their peers do.

Our work will offer teachers an easy way to significantly enhance a low-tech classroom tool they already use. Teachers will be able to scan handwritten worksheets and easily view class data to determine what their students are struggling with instead of guessing based on observation or gut feeling. Additionally, students will be able to collaborate with one another more easily, using a simple, familiar interface instead of more complicated solutions like computers or tablets.

Existing Solutions

Most classrooms, especially those for younger students, focus on problems that have a single correct answer. The Tufts Center for Engineering Education and Outreach (CEEEO) aims to expand how students are taught, trying to create an environment in which the primary focus is how students get to an answer, allowing for students of all learning methods to thrive. This means that most problems are more design oriented with a range of possible answers. Since this methodology is fairly unique, there aren't many worksheets that focus on problems beyond right and wrong answers. As a result, most existing solutions are grading solutions, which examine whether or not a student is correct. Some of those solutions are listed below:

- Livescribe
- GradeCam

These solutions are sufficient at determining the correctness of a student's response. However, they fall short at improving collaboration between students and promoting a range of methods for determining a response.

Sponsor's Goals

Dr. Danahy oversees a variety of developmental projects around education technology and incorporating them into educational environments. One of these projects is the InterLACE project, or, Interactive Learning and Collaboration Environment. The InterLACE worksheet is one component of this entire environment.

We are working with a graduate student on Dr. Danahy's team. The graduate student at the CEEEO is completing the worksheet creation software for the project. We are responsible for the software to analyze the worksheets after they have been completed.

Ultimately, Dr. Danahy hopes that we can establish a suite of tools and analytics for different types of worksheets. Ideally, these tools can be applied to almost any kind of worksheet that can be created, but some may also be specific to different types of worksheets. This suite of tools will operate independently from other InterLACE projects as well as the work of the graduate student. However, we will be in constant communication with the graduate student so that if we create analytics features that work

for specific types of worksheets, he can incorporate those into his worksheet creation software.

User Experience Overview

Our solution is aimed to help both teachers and students in the classroom. With the feedback about students work, teachers can better verify classroom dynamics and conceptual understanding as well as promote collaboration in their classroom. With visualizations about classroom trends and their peer's ideas, students can improve their own ideas. Ultimately, most of the work (image processing and analyzing) is being done behind the scenes. Therefore, the main constraint of the solution is that teachers can easily understand the interface and navigate around it without any training. Below, we outline a few possible scenarios for both teachers and for students:

Scenario A: The teacher begins by creating a worksheet on the web app that is being created by the graduate student. The worksheet asks students to draw their vision of what the universe looks like. They are asked to label all components that they can. Students spend 5-10 minutes drawing their ideas and once they are done, they place their worksheet in the scanner. As worksheets are scanned, the raw image is displayed to the screen and visible to the class through the overhead projector. Students can easily see other student's ideas and can add to their own drawing. Upon adding their changes, they can tag their peers by writing their name on their drawing and then place their drawing in the scanner again, to update their image.

Scenario B: The teacher begins by creating a worksheet on the web app that is being created by the graduate student. The worksheet aims to improve students understanding of gravity. Students are asked to drop their pencils from various heights and to record both the height they dropped their pencil, and the time it took for the pencil to hit the ground. After completing the task, students scan their worksheets and a graph displays the relationship between distance and time. Students can easily visualize this relationship since the data collected by all the students in the classroom is more informative than the 4 data points that they collected themselves.

Goals

This project has a variety of interesting applications in the classroom. There are so many types of worksheets that can be created, and so much information that can be gleamed from these worksheets, that we have created a few levels of achievement for the goals we hope to accomplish. However, before any of these are possible, we must first create a way to process the images and extract the data from them. Therefore, our initial challenge will be to create an image-processing suite that can handle almost any type of handwritten data (number, word, image). We have broken this suite into three main components: image preprocessing, character prediction and character aggregation. These components are discussed in further detail in the *Major Technical Components* section below. Given the challenge that image processing creates, below are a few of our levels of achievement:

- Level One
 - Detect handwritten numbers – Find numbers on the worksheet and determine what value was written.
 - Detect handwritten words – Find characters on the worksheet and determine what word (or character) was written.
 - Simple database for class lists – This will make it easier to detect names at the top of the page since we'll be working with a smaller set of possible words.
 - Simple web app – An easy front end where visualizations of the worksheet analytics will be displayed after the worksheets are scanned.
 - Connect with the worksheet creation software created by the graduate student
- Level Two
 - Tagging - Recognize student names and custom tags. Think of this as similar to Twitter's tagging of people and ideas.
 - Classroom collaboration visualization – A direction graph outlining the student collaboration for a worksheet.
 - Quantity detection - Detect HOW much has been written on a worksheet. The quantity of the student's response is an interesting piece of data that can explain more about how they think (depending on whether they drew a picture, wrote a sentence or wrote a number).
 - Simple data comparison - Aggregate table, graph or heat map comparing data obtained by different students in the class. See *Scenario B* in the *User Experience Overview* section for an example.
 - Iterative analysis - Allow users to update worksheet and scan multiple times to track changes/updates (in conjunction with tagging)
- Level Three
 - Personal handwriting machine learning - Some sort of supervised/reinforcement learning to help analyze each student's handwriting individually. At the moment, all individuals are treated as the same person even though they all have different handwriting patterns.
 - Note: It may be a challenge to include active reinforcement learning since that requires knowing what they wanted to write. This would required extra interactions and that may not be feasible.
 - Control EV3 with handwritten instructions - Automatically program the EV3.
 - Check for spelling (and math) errors - Provide some sort of simple feedback when students spell something incorrectly or make a mistake on a problem.

Major Technical Components

Our solution can be broken up into image processing and worksheet analytics and visualization. These sections, all of whose code will be written in Python, is discussed below:

Image Processing

- *Image Preprocessing*

The goal of the preprocessing section is to remove as much noise as possible without distorting the image. We have begun to experiment with the effectiveness a Gaussian blur and opening and closing the image as well as static and adaptive thresholds. The preprocessing section will produce an inverted binary image where the white section is a single character (number, letter or symbol) and all else is black. OpenCV has been effective at manipulating images to extract data.

- *Machine Learning Predictive Models*

The goal of the predictive model section is to take a preprocessed image of a single character and predict the value of that character. We have begun to experiment with the effectiveness of various Machine Learning techniques, specifically a Support Vector Machine and Artificial Neural Network. Additionally, the use of a pool of dynamic classifiers, where multiple classifiers are used together to improve the effectiveness of classification, has been attempted and appears to carry with it significant success. The python sklearn library has been particularly helpful and we expect to use more.

- *Number / Letter Aggregate*

Optical character recognition, or OCR, is the conversion of images of text to machine-encoded text. Existing OCR libraries have proven to be extremely unreliable with handwritten characters and we expected them to be even worse with characters written by children. Therefore, we are defining our own classification section and then using the OCR libraries to map these characters to machine-encoded text. Pytesseract appears to be the most effective python OCR library though we have not begun to use it yet. This section will be responsible for determining if characters are spaced in a way that they should be treated as a single word or as individual characters (or numbers).

Worksheet Analytics and Visualization

Given that we are working in Python for the image-processing component, we have decided to use Python for our analytics and visualization to avoid any complications with integration between the two components. We are planning on creating a web application using the Django web framework.

Technical Overview

It appears the image processing section will be the most time intensive. Handwritten character recognition is an open problem that has yet to be completely solved. There are various solutions, which are extremely effective, but these are designed for adult handwritten characters and not those specifically designed for children.

Evaluation

Ultimately, our solution will be a success if it is used in classrooms and improves collaboration amongst students. However, this project can still be a success even if that doesn't completely happen. Building out an effective and efficient image processing system would be extremely helpful for creating a strong foundation for the project. Individual character classification accuracy and complete word accuracy are two metrics, which could be measured to evaluate the effectiveness of our solution. It's unclear what accuracy percentage would be possible but we will continue to be in touch with our sponsor to find a threshold that we expect to reach.