Eulexia

A GLASS application
CSE 118: Phoebe Gao, Michelle Wu, Andrew Du
CSE 218: Busheng Lou, Sebastian Cheah
Topics

- Background & Motivation
- Challenges
- Design
- System Development
- Testing & Evaluation
- Collaboration & Future Work
Background & Motivation
What is Dyslexia?

A language-based learning disability that particularly impacts writing, reading, and spelling.

People with dyslexia require alternative learning methods:
- Audio versions of a text
Motivation

People Problem

40,000,000+
American adults have dyslexia
Dyslexic students often misspell words and need a spell checker to catch mistakes

Technical Problem

No existing ubiquitous solution to this problem
DCODIA: Aids in reading, not spell checking
Online spell checkers
Amazon Echo
Motivation

End Users
Dyslexic students who have trouble with spelling words correctly

Goal
Develop a Google Glass spellcheck application for dyslexic students to use in a classroom setting
Why Google Glass?

- Ubiquitous
- Simplicity of use
- Audio feature
Challenges
Challenges

Words Recognition - OCR
- Local OCR vs Internet OCR
- First failed using OpenCV/Tesseract - poor recognition results

Google Glass
- Dependent on mobile hotspots for internet
- Not well supported in Android libraries
- Slow development cycle
Design

MY HOBBY:

OK, GLASS, CHECK TOMORROW'S WEATHER.

OOH, SNOW!

OH MY GOD, IT'S SOMEHOW EVEN MORE ANNOYING THAN IF YOU HAD IT.

SAYING "OK, GLASS" BEFORE EVERYTHING WHILE WEARING REGULAR GLASSES.
Application Flow & UI Design

Word Recognition
Spell Check
Word Suggestions
  ■ Text-To-Speech
  ■ Save selected spelling
1. The Start

Tap to check spelling
2. Picture Time!

I’m going to buy a wayst traener for Black Fryday
3. Saving Picture

Loading...

Saving Image...

0% 0/100
4. Recognition & Spell Check

wayst

Tap to see suggestions
4. Recognition & Spell Check

traener

Tap to see suggestions
5. Suggestions

waist

Long press to hear spelling
5. Suggestions

waste

Long press to hear spelling
System Development
Architecture

Android Studio

- Google Glass App

Architecture must support the following:

- Take an image
- Process image for text
- Spellcheck input text
- Provide suggestions for misspelled words
- Audio feature for spelling out loud
- Save commonly misspelled words
Technology

Google Glass Explorer Edition
- Camera
- Audio speaker
- Voice and gesture commands

Optical Character Recognition (OCR)
- A9T9 - Current implementation requires an Internet connection

Spellcheck and Suggestions
- Jazzy Library
- Suggestions based on closest neighbors
- Built-in dictionary only supports English for now
  - Preprocess into memory or leave on disk
Google Glass Camera

5 Megapixels

OCR solution requires an internet connection

- 5 MB at least uncompressed
- Compression leads to degradation in quality

Camera will capture a small range of the user’s field of vision

- Provide a live camera preview when taking a picture (Warning: may lead to overheating issues)
OCR

Test case
- Image taken from google glass with printed text centered

Tesseract for Android (tess-two)
- Poor results
- Require more research into tuning and image morphology

a9t9’s Online OCR
- Based off of Microsoft’s OCR (Optical Character Recognition) engine
- Better results than tesseract
a9t9 is comparable to Abbyy Cloud SDK

<table>
<thead>
<tr>
<th>OCR Service</th>
<th>Result</th>
<th>Output (Excerpt)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abbyy Cloud SDK</td>
<td>Good</td>
<td>BEIJING — Jing Yuechen,</td>
</tr>
<tr>
<td>Google Docs OCR</td>
<td>Poor</td>
<td>EEN; — ing Yuechen,</td>
</tr>
<tr>
<td>OnlineOCR</td>
<td>Good</td>
<td>BEIJING - ding Yuechen,</td>
</tr>
<tr>
<td>i2 OCR</td>
<td>Poor</td>
<td>BI-'JtNG — Jing Yuechen,</td>
</tr>
<tr>
<td>FreeOnlineOCR</td>
<td>Fail</td>
<td>Bta t0 mt 21 the e</td>
</tr>
<tr>
<td>Tesseract Online</td>
<td>Fail</td>
<td>BEIJING — -1 ing Yuodzcn,</td>
</tr>
</tbody>
</table>
Results using mobile camera on wrinkled printed text

<table>
<thead>
<tr>
<th>OCR Service</th>
<th>Result</th>
<th>Output (Excerpt)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abbyy Cloud SDK</td>
<td>Poor</td>
<td>rely heavily on less-fettered</td>
</tr>
<tr>
<td>Google Docs OCR</td>
<td>Poor</td>
<td>rely muneinterne - anorameror</td>
</tr>
<tr>
<td>OnlineOCR</td>
<td>Poor</td>
<td>rely heavily on lass-fettereir</td>
</tr>
<tr>
<td>i2 OCR</td>
<td>Poor</td>
<td>rely hecavilyon less-fctt</td>
</tr>
<tr>
<td>FreeOnlineOCR</td>
<td>Fail</td>
<td>(no output)</td>
</tr>
<tr>
<td>Tesseract Online</td>
<td>Fail</td>
<td>7- \5g@llf&lt;. - nu-lain</td>
</tr>
</tbody>
</table>
Spellcheck & Suggestions

Jazzy library
- Must provide a valid dictionary file
- Depending on implementation, a corresponding dictionary file may be needed

Processing
- Preprocess by loading into memory
  - Lengthy startup time
- Read from disk during processing
  - Normal processing times increase
Features

- Live preview for taking pictures
- Uses OCR to check for misspelled words
- Offers word suggestions with an option for text-to-speech
Testing & Evaluation
Testing

title screen → live camera preview → misspelled words → spelling suggestions

- Separate module testing (White-box testing)
- Application user-based testing (Black-box testing)
Evaluation

- Great recognition for text
- Good recognition for neat handwriting
- Expected suggestion and text-to-speech

- Speed is the bottleneck
- Does not recognize messy handwriting
- Google Glass overheating
- Requires Internet
Collaboration & Future Work
Collaboration

- Split into smaller subgroups to work on different features
- We all worked together to brainstorm and troubleshoot when needed
- Biggest issue was coordinating our schedules, but we overcame it by constant communication online and pairing up with team members who have similar schedules
- Facebook Messenger, GitHub
Split work by function domains - traded off mini projects to keep it more interesting and have a better idea of how everything functions together

- OCR
- Spellcheck and Suggestions
- Record of the user’s commonly misspelled words
- User Interface
- Testing
Post Mortem

Risks
- Finding a reliable OCR engine (there is not much available for free)
- Potential overheating of Google Glass
- Different forms of dyslexia come with different needs

What went right
- Spellcheck and UI went smoothly

What went wrong
- TessTwo did not work

What we would change
- Spend less time trying to get TessTwo to work better since we had such a tight schedule
Lessons Learned

- Google Glass software and hardware limitations
- Importance of time management
- Working with limited resources (2 Google Glasses)
Future Work

User testing for usability and finding pain points

Preprocess Images

Update OCR library and related API to be more powerful

Handwriting can be detected correctly which is super important
  - Increase speed

Highlight the position of the misspelling words
  - Users can easily track their potential mistakes
Thank you!

Any questions?