Final Project Report

Dimitrios Galinos
University of Copenhagen
bst265@alumni.ku.dk

Gabriela Dvořáková
University of Copenhagen
kzh186@alumni.ku.dk

Denis Drobny
University of Copenhagen
htp440@alumni.ku.dk

Maurits Pallesen
University of Copenhagen
jrk712@alumni.ku.dk

Introduction

Learning a new writing system is challenging and takes a lot of practice. For example, Japanese and Chinese characters require not only memorization of complex stroke patterns, but also its correct ordering and directions. Relevant, meaningful activities that students find engaging helps build neural connections and long-term memory storage. We propose an application for mobile devices, that will use augmented reality for teaching writing characters in various writing systems. When using our application, the user will hold the mobile device over a paper and use the virtual directions to write.

Project idea and motivation

There is no shortage of applications and solutions focused on teaching languages, both written and spoken. To our knowledge, however, most (if not all) of these applications help you write the characters on the screen of the user device, which is rather different than writing them with pen on paper. We propose a mobile application that targets modern smartphones and tabloid devices and utilizes their camera in combination with augmented reality technologies to help the user learn to write languages directly on paper.

This is especially useful for written languages in which there is no latin alphabet such as Chinese and Japanese. These languages are the primary motivation for this application.

These written languages consist of thousands of characters representing different syllables, each consisting of a set of strokes. Words are then made up of one or more syllables. Learning to write these languages is difficult for multiple reasons, but primarily because there are so many characters to learn and to write each character properly one should draw the strokes in a certain order, and scale each stroke appropriately in comparison to the other strokes in the symbol.

The user would point their smartphone camera to a physical paper. The application would use augmented reality technology to virtually draw the symbols at the paper and guide the user through the drawing process of the symbols.

Basic User Interaction

1. The user points mobile device's camera at the piece of paper.
2. After camera starts tracking paper, the app responds by displaying a symbol to be drawn.
3. Then the symbol is decomposed into the set of strokes. In each step, one stroke is highlighted alongside with some guides, such as arrows or numbered tips of the stroke.
4. User starts drawing the stroke on the paper.
5. The aim of our app is to recognize when user is done drawing the stroke and automatically move to the next stroke until the whole symbol is finished.
6. When the symbol is complete, user can hear the correct pronunciation from the speaker.

Planned contributions

Our proposal's main contribution is creating an application and making it easily available to any individual with a smartphone, which simplifies and eases the procedure of learning foreign languages that do not use an Latin alphabet. In particular, step by step drawing of a selected language's symbols and their pronunciation.

Skills needed to accomplish the project

In order to successfully develop the application and test it, we primarily need skills in smartphone application development, as the application targets modern smartphones and tabloid devices. This includes skills in augmented reality development, user experience development and testing.

Furthermore, we need to research how the user best interacts with the application and how the application best engages the user and how it best helps the user learn the written languages.

Required software and resources and how to to acquire them

The goal of our project is to develop a mobile application, that would not require any additional hardware equipment. The core objective is to create an engaging user experience for learning graphically complex characters, that is accessible to a wide audience. Therefore the only hardware required for development and the subsequent use is a mobile device.

We have finalized our initial assessment and will indeed use react native to implement our application's front end. Moreover we have decided to use ViroReact and ViroMedia as our main AR libraries. Viro Media also offers basic support for computer vision tasks which we exploited. We are going to mainly target Android but we will also work on making

our app work in iOS since we are using react native and the application build can be easily exported for both operating systems.

Finally regarding device capabilities we require the smartphone to support ARCore (most smartphones that are not more than 2 year old support it, full list can be found here).

Success criteria

After careful thought, our team considered the challenges and motivations of our proposal and also took into account the learning outcome of the course reaching the following conclusion in regards to the success criteria:

- Finalize a proposal for a Mobile App that uses state of the art technology, takes full advantage of a mobile device's capabilities, be feasible and last but not least improve the users' life in some way.
- Explore and learn technologies that the group members have not used before, from simple Mobile interfaces to Augmented Reality development.
- Provide a minimum viable product app that will be able to capture the spirit and motivation of our finalized proposal.
- Evaluate and test our application. Research how the user interacts with our application and what can be improved.
- Provide a base for future development and possible release of a commercial application.

Related work section with more fleshed-out relevant work

Products which may seem similar to what we are trying achieve were already developed. There are two fields in which similar applications were already developed - apps for teaching how to draw objects using artificial reality (for example SketchAR [1]) and language learning mobile applications whose interaction with user is solely through screen of a mobile device (such as Learn Japanese! [2]).
Problem with the former type of products is that it is not well suited for language studying and it does not support educative elements such as character pronunciation, those apps aren't aware of the correct ordering of strokes of characters' which may be crucial and they do not dispose of suitable image library that would contain alphabets. Even though it is possible to somehow upload custom images, it makes the learning process much more timely and cumbersome.
On the other hand, the latter group of application is solves most of the problems described, however it relies on user drawing characters either using his finger or a stylus. It is clear that drawing by finger is movement which is very different experience than actual process of drawing using pen which may be not well suited, especially in the early stages of learning.
The drawback of the solution with stylus is that normally stylus works only with larger devices as tablets, which are in general more expensive than mobile phones and the stylus itself is usually a pricey item.
With our domain specific application targeting smart devices which doesn't need to contain any special pieces of premium high-end technology, we aim to both solve the problems described and target possibly very wide range of users of any age or social group.

Approach to the project

We are approaching the task at hand in a very collaborative manner. We have a team meeting at least once a week where we discuss how it it is going with everyone's tasks

[1] https://play.google.com/store/apps/details?id=ktech.sketchar&hl=sk
[2] https://apps.apple.com/us/app/learn-japanese-kanji/id1078107994

so we can help one another if there are any roadblocks in a specific task, or if some group members are busy with another course.
We have managed to set up a development environment that supports React Native, Viro React and Google Vision, and we are collaborating on the application through GitHub. This ensures transparency in the process such that everyone can follow along on each others tasks.
This setup should in theory allow all group members to run, develop and debug the application locally, however we have met some obstacles along the way.

Results and analysis

What we managed to achieve
We have developed an app that is in align with the requirements defined by the previously stated success criteria. We called the app *LeARn*. LeARn is a mobile application solution that uses AR to help people learn complicated characters from several writing systems, for example Japanese or South Korean, in an interactive and engaging way. We planned to support both Android and iOS platforms, however, due to the `Viro Media` library, the final product targets only Android devices and is written in React Native.
We made sure our solution offers a friendly interface that is easy to use and an appealing design. It starts off by letting the user to pick a language of choice and a character they wish to learn. We used augmented reality to place the 3D character on the paper plane, that navigates user's drawing stroke by stroke. After the drawing is finished, OCR techniques were applied in order to check the correctness of the drawn character. The sound button is available throughout the whole drawing time to hear the correct character pronunciation. If the drawing is recognized as successful, user can repeat the learning process with the same character or continue to learn another character. We have also gathered feedback to measure the success of the app from the user's perspective.

How we managed to achieve it
We have managed to create the application using most of the software stated in earlier sections. The primary tool has been *Viro React* in combination with *Viro Media*. Viro react is the library that allows easy augmented reality development, and Viro Media is the mobile application that let's you run and debug the application on any smart phone or tablet that supports augmented reality.

Viro React contains all the features we needed to make the application. We use its AR functionalities to place the augmented reality models of characters in the video captured by the phone or tablet. The user can drag these models around to place them manually on the paper they're writing on if the application does not place them in a satisfactory manner automatically.

Viro React also supports image recognition, which is the tool we use to recognize characters. We have taken 5 different images of the supported characters that Viro React uses as reference images. When it detects just one of the images, we accept the user written character. We use 5 reference images so the application can detect the characters with some variety.

The interface has been created using standard React Native.

Finally the 3D models were created using a free online 3D modeling tool called 3D Slash (www.3dslash.net). Our original motivation was to create 3D animations of the strokes but after many unsuccessful attempts with the Unity engine and Blender we understood it was more complicated and time consuming than what we could afford so we settled for the simpler transition of different 3D models.

Testing
After completing the development part of our application we wanted to do usability testing with real users and for that purpose we created the scenario where a user wants to learn four simple Japanese characters using our app. We used as testers friends, family and roommates. All of the testers were given a device were leARn was running without any instructions and they were requested to learn 4 japanese characters. After completing the test they were requested to fill this google form to provide us feedback. Regarding the testing results we got feedback from people with different technological backgrounds and the results speadsheet can be found here. A small conclusion of our test study would be that the users enjoyed the idea of working with AR for learning, a number of problems were recognized with our implementation though, which we were already aware of and could not perfect on time

Conclusions and future work

Reflections on a different approach

The biggest problems we had were linked to cross platform development of the application. We wanted to target both iOS and Android devices (as in our group there are people with both kinds of phones), that's why we chose developing in `React Native`. The most viable option for portable artificial reality seemed to be the `Viro Media` development platform.
However, it became source of many issues, since we later learned, that it is not being maintained on regular basis (at the moment of writing the latest update is one year old). Another problem was that the stable version of `Viro` is compliant with `React Native` of version 0.59, whereas the current stable version is 0.62. That may have been why we couldn't get to work certain dependencies with our app - such as a screenshot library `react-native-view-shot` which we wanted to use in order to speed up and improve character recognition process.
`Viro Media`'s debugging support also isn't great and lack of debugging was making our job quite cumbersome. Another problem was that even though the library has some documentation, there's not so many examples and in general the framework is not used very widely, therefore it's harder to find answers to some question that we had. Also we ran into some bug of the framework which haven't been yet fixed.
To sum it up, we believe it would haven been better to focus on one target operating system and work with native artificial reality framework for the OS.

How we rate/like our final product

If we consider the given time we had to develop the application and our experience with artificial reality, image recognition and mobile apps in general, we would rate our product as a success.
Of course there are some flaws which has to do something with usability of the app we weren't able to solve until the deadline - mostly character recognition which we were forced to do using built in `Viro Media` OCR functionality instead of desired `Google Vision` (why we were constrained to do it this way we elaborated in Section What we would do differently if we started over). Also we ended up focusing on Android devices as we had issues with `Viro Media`'s iOS support, so we failed to make a fully portable mobile application.
Other than that we believe we delivered a minimum viable produt of good quality which meets the goals we set ourselves in the project proposal.

Future work

There are two major areas regarding future work on this project. One is development to improve the application and the other is research on the applications ability to teach non

Latin alphabets.

As mentioned, we ran into obstacles that we did not manage to overcome in the development process such that leARn's optical character recognition did not end up as accurate and fast as we had hoped. We believe that switching from using Viro React's image recognition software to Google Vision optical character recognition would greatly increase both speed and accuracy in determining whether the user draws the correct character. The next development iteration should also add guidance to the user about which direction to draw each stroke in a character on top of which order to draw the strokes. There are also user interface issues we were not able to solve in the way we would like to due to the way Viro React works with React Native. For instance, the **scan** button is a floating button in the augmented reality world, whereas we would have liked it to be statically positioned on the screen. We had several similar problems that should be solved for optimal user experience.

Due to the mentioned development issues we had limited time to research the applications ability to teach the selected characters. While we did manage to have some people try out the app and tell us about their experience with it, a more thought through research needs to be done in order to know the real teaching capabilities of such an application. It would be relevant to see the difference in teaching capability under different usage scenarios, for instance when the user holds their phone in their hand and writes on a loose piece of paper in contrast to a user that has the paper held in place by a clipboard. We believe the best case scenario for learning through an application like leARn is using a clipboard for the paper and having a phone holder that can keep the phone in place in a position that makes it comfortable and easy for the user see where and what they should write. However, there is no way to know without doing the research.

Final Video

The link for our Final video can be found here