Employing Mobile Applications in Human-Machine Interaction in Visual Pattern Recognition Research

Amir Schur(✉) and Charles C. Tappert

Seidenberg School of CSIS, Pace University, 1 Martine Ave., White Plains, NY 10606, USA
amirschur@aol.com, ctappert@pace.edu

Abstract. This study is part of the first author's continued dissertation research in human-machine interaction in visual pattern recognition. Previous research focused on evaluating human-machine interaction using a flower recognition tool. Initial research showed that human interaction in color recognition improved accuracy significantly. We then looked more deeply into various automated color recognition algorithms and ways of combining them with human feedback. Described here is the process of upgrading the initial system into a new mobile application using Appinventor. After data collection, models were built for various color spaces. Sharing this experience may help other researchers incorporating a human-computer interaction component into their work.

Keywords: Human-computer interaction · Visual object recognition · Pattern classification · Feature extraction · Appinventor · Color space

1 Introduction

Initial research in comparing accuracy and time to accomplish visual pattern recognition tasks among humans, machines, and human-machine combinations showed favorable results for the human-machine combination. While maintaining a reasonable time to accomplish the task, human-machine interactions resulted in better accuracy compared to human activities alone or machine alone, especially when humans assisted in the color recognition process. In this flower species identification task, human activities were performed by amateurs and not experts [1]. Why human assistance is so valuable in the color recognition task is still being investigated. In addition, a different tool, or at least an upgraded one, was considered necessary in continuing this research, as well as investigating the various color spaces.

A tool upgrade to a mobile application was desired for ease-of-use and portability, as collecting flower pictures is required during data collection. The original Interactive Visual System (IVS) is a native java application running on a typical X86 machine with an option to install on a palm pilot [1].

Several development tools to accommodate mobile application development were investigated. These included NetBeans, Eclipse, and IntelliJ, as well as cross platform development tools such as Codename One where a one-time build is suitable for

C. Stephanidis (Ed.): HCII 2015 Posters, Part I, CCIS 528, pp. 696–699, 2015.
DOI: 10.1007/978-3-319-21380-4_117

various platforms such as Android, IOS, etc. Finally, Appinventor, created by Google, but now maintained by MIT (at appinventor.mit.edu) was investigated. This tool has gained enormous popularity in the past years, with 3 million users representing 195 countries worldwide.

Appinventor is not a typical application development tool. It has been simplified so that users do not need to master a typical programming language. It is advertised as the perfect tool for teaching basic software engineering, basic application development, and research. With the simplicity of the development process, the developer is allowed to focus more on the idea level rather than spending time developing an application, and thus not being dependent on the availability of an application developer.

As an upgrade to the original IVS, a data collection tool using Appinventor was created. The tool was utilized to perform a set of tasks and an evaluation of its performance evaluated. Due to performance issues the tool will be replaced with a web application using python with Flask framework, requiring a savvy developer to create a web application. Nevertheless, it should be mentioned that Appinventor is considered a good alternative to create a mobile application quickly.

2 Appinventor Data Collection Tool

Appinventor provides a block-based programming tool for creating an app. Users must design the screen layout with available tools/objects to display on the mobile screen. After labeling each object, activities must be defined using their building-block tool: what happens when a particular object is clicked, long clicked, etc. Users can then test the mobile application by connecting an android device to the website, or by running a simulator locally. When the user is satisfied, the application can be packaged into an android application package (apk) file and then immediately installed on the user's Android device. A simple mobile application can be built within an hour.

For human-machine interaction research, this opens a vast area of research and data collection possibilities. The mobile app can quickly collect GPS data, user activities, user behavior and responses, etc. There are various built-in sensor functionalities readily available, such as accelerometer, orientation sensor, proximity sensor, location sensor, bar code scanner and near field sensor. In this study, users chose the primary and secondary colors of petals, the primary color of the stamen, and provided a petal count.

The users took pictures of flowers via the relatively high-resolution camera built-in to mobile devices. These pictures are typically 3−4 MBs in size, compared to the lower resolution images 100−200 KBs obtained in previous data collections [1]. The Appinventor tool developed was named Interactive Visual System 2 (IVS2).

Another useful feature of Appinventor is the ability to save data in the cloud, in particular Google Drive, allowing easy access and analysis to the researcher. A native authentication method using OAuth 2.0 was implemented, allowing the team to store and view data in Google Drive.

The Appinventor mobile application tool was used to take pictures of various flower species, perform manual feature extraction, and upload the pictures and data to Google drive (Fig. 1).

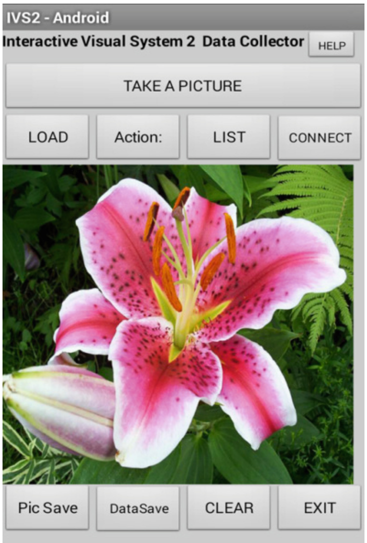

Fig. 1. IVS2 data collector android app

Below are issues with the application:

- The application has too many options for the user. The application should be more streamlined during the process of using it.
- The error handling is flawed. A JSON error report is presented with a vague error code.
- The application allows a user to load the data before loading the picture. This should be restricted so that the user cannot upload data without first taking a picture or opening an existing picture in the device.
- Because the uploaded information is not cleared from the mobile device and no history of previous posts is recorded, collection of redundant data is likely. Once uploaded, the information should be cleared automatically from the mobile device, or there needs to be a way to track which pictures on your phone have already been uploaded.
- A method should also be provided to save data locally.

Although all of these problems can be easily fixed within Appinventor, these functionalities have been moved into a web application for the testing phase, which will be performed after the data modeling phase. The web application uses Python with Flask framework and is more guided in the data collection process with a changing screen after each activity is completed with automated data saving capabilities. For the web application, however, the infrastructure that can host a web application is required and this requires good programming support.

The portability and speed of a mobile application makes it a good tool for human-machine interaction data collection. Human-machine research often involves software development activities and infrastructure, which is often not readily available.

Tools such as Appinventor can be utilized by a researcher who doesn't have a software engineering background, or a software developer readily available. The researcher can quickly build a simple to moderately complex mobile application using Appinventor. The learning curve for this framework is small (days) compared to learning and becoming proficient in a computer language (months).

3 Continued Research

Once the data collection process has been established to obtain training data set, the research process will be continued. Because color perception was an issue and we were only using the RGB color space in the past [1], finding the best color space in addition to using high resolution images is expected to increase the accuracy level. For example, we expect a higher level of accuracy using CIELab color space since it is considered to be a better model for human visual perception [2]. To be complete, however, the extracted features of various color aspects of the flowers will be translated into the dominant color spaces: HSI, XYZ and CIELab.

The next step will be to build a data model using IBM SPSS Modeler. We will build separate models for each color space and then compare the accuracy results. Once we have a solid model, we can continue our testing phase. We will build the test data set and perform testing to reach our conclusions. The results will be compared with previous research to develop a better understanding of what is the best method of combination of human and machines in the visual pattern recognition process.

References

1. Schur, A., Tappert, C.: Combining human and machine capabilities for improved accuracy and speed in visual recognition tasks. In: Stephanidis, C. (ed.) HCI 2014, Part I. CCIS, vol. 434, pp. 368–372. Springer, Heidelberg (2014)
2. Ugarriza, L.G., et al.: Automatic image segmentation by dynamic region growth and multi-resolution merging. IEEE Trans. Image Process. **18**(10), 2275–2288 (2009)