

# An AR Application for Wheat Breeders

Kaitlyn Becker, Frederic Parke, and Bruce Gooch<sup>(✉)</sup>

Texas A&M University, College Station, USA  
kaitlyn.c.becker@gmail.com,  
parke@viz.tamu.edu,  
gooch@tamu.edu

**Abstract.** We report on an Augmented Reality interface to speed the workflow of food crop breeders. The goals of the interface are to make data collection in the field more efficient to accelerate the breeding cycle. For hardware combine the Recon Jet sports computer sunglasses and a Neutab N7 Android tablet. For software, we augment the Kansas State University open source Fieldbook application with barcode reading capability, a speech recognition interface, and information displays. We design and evaluate the application using a cohort of wheat breeders.

**Keywords:** Augmented reality · Agriculture · UIUX · Speech based interface · Plant breeders

## 1 Introduction

As the world's population continues to increase, food production works to keep up. The Food and Agricultural Organization of the United Nations (FAO) defines wheat as a staple food. The FAO also states that wheat along with rice and maize make up 60% of the world's food intake. However, we lose over half of the world's wheat to disease and pests. We need new varieties that are more resistant to disease and pests, which are tolerant of poor growing conditions, and that have higher germination rates and produce more grain per stalk. This ongoing research is imperative to keep up with the world's growing demand for food.

Many researchers today still rely on traditional log books for data collection. In scientific fields like chemistry and archeology, entomology and anthropology, the tried and true pen and paper method of note-taking is common. Accuracy and neatness are emphasized because the research is only as accurate as the data collected. As mobile technology becomes more widely available, developers are offering more efficient alternatives to traditional data collecting methods. On a computer, keeping notes organized is simple, and with a mobile computing device, this technology is as portable as a paper notebook.

Field Book is an open source application intended as a digital note-taking tool for wheat researchers. The goal is to provide researchers and growers with technology that is more efficient and accurate than traditional log books. A key goal of the initiative is to make information technology accessible by making tools inexpensive and intuitive.

In the field, crop scientists work in teams of three. One person visually identifies the plant to be measured. They visually grade the plant then speak aloud in a wheat grading markup language. The second person writes the plants condition in the Field Book. The third person navigates the group and tells the grader which plant to grade.

While Field Book supports digital data collection, it requires manual input. Researchers are forced to juggle between manipulating plants and recording data, slowing the process and requiring multiple people. We implemented hands-free field data collection without the need for an assistant. We assume that users do not have internet access while recording data in remote areas. As such, the data collection capabilities of the software have no internet dependency. We export data from the Android tablet in the lab.

*Field Book* currently allows users to record speech for later transcription. This feature is not commonly used by researchers because of the time it takes to later transcribe these notes (Trevor Rife, personal communication, May 19, 2015). With speech recognition, the user would have a written record that could be transmitted to colleagues immediately without the transcription time.

By utilizing speech-based software developed for multiple languages, the interface could also be made available to non-English speaking researchers. With the addition of speech synthesis, the app could read aloud (in the native language) instructions or descriptions to the user, making this app accessible to those with limited literacy. Since the system is most often used in bright sunlight (Trevor Rife, personal communication, May 19, 2015), read aloud functionality could be useful when screen readability is poor. Spoken text could also be added to the in-built tutorial.

The system augments the current Field Book application. It contains three components. The first is barcode recognition software to aid in visually identify the particular plant in a greenhouse or test plot. The second system is speech recognition software. It is important that the user not feels the need to check the accuracy of each entry. Therefore the third component is feedback in the form of an information display.

Since the advent of modern agriculture, field researchers have used paper to record their data. Survey books with waterproof pages made to fit in a pocket are often used. There are conventions as to how these pages are laid out and how the data is recorded.

Figure 1 shows an example of data layout. Traditionally, data in columns are on the left, and figures and sketches are on the right. The necessity for precision makes recording in a survey book a time consuming endeavor. Researchers have the option of designing and reproducing a form that is custom tailored to their specific project. This requires the researcher to know ahead of time what data they will be collecting and offers little flexibility.

Crop Scientists currently mark test plots with barcode markers that identify the gene line. Following barcode recognition, the display illustrates the name of the gene line. The grader then speaks aloud as usual. We used PocketSphinx application and recordings of Scientists speaking all the phrases in the plant grading markup language. We used PoketSphinx because it is free and can operate without internet access. The application records and recognizes speech and displays an icon based information display of the plant's grade. The grader can then move to the next position.

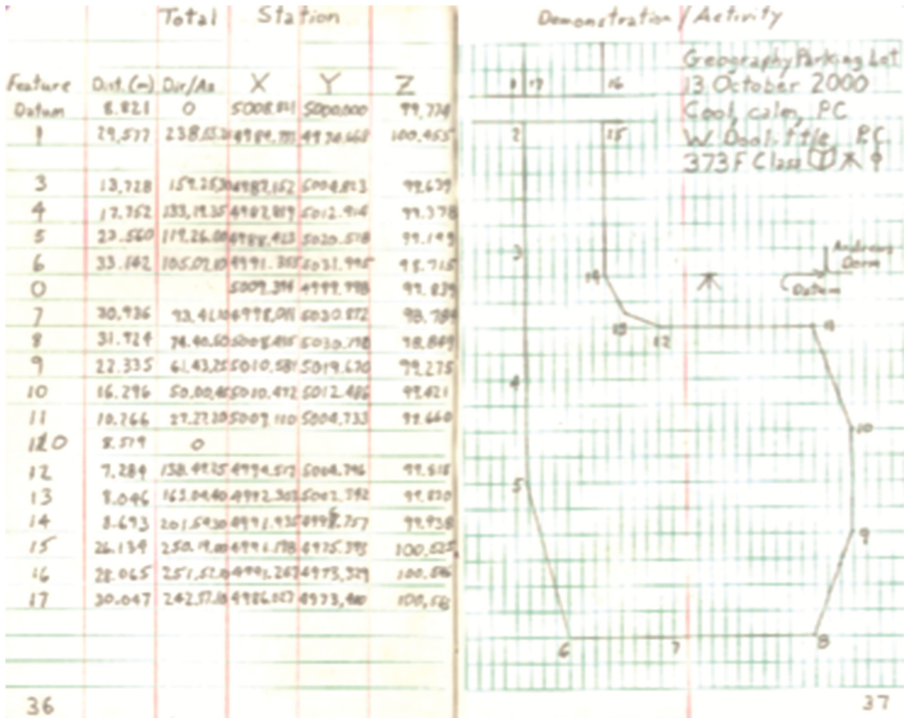


Fig. 1. An example of a paper journal currently used in field scouting.

## 2 Background

### 2.1 Mobile Applications for Data Collection

Users are becoming more comfortable using technology for tasks previously done on paper. According to a survey conducted by Princeton Survey Research Associates International, 50% of American adults own and use either a tablet or e-reading device (Zickuhr and Rainie). In recent years, researchers have begun adopting new technology in an attempt to make the process of data collection easier and more streamlined. With advances in mobile technology, digital tablets are becoming more useful as data recording media. They offer the portability of a logbook with the computational power of a computer. With a well written application, a user can edit, reorganize and customize their recorded data, as well as disseminate the gathered information efficiently.

Of the researchers who use a tablet as part of their data collection, many rely on Microsoft Excel. Excel offers a customizable grid and powerful mathematical functionality. Averages, totals and other survey data can be calculated and updated on the fly and can be translated into graphical format to help visualize information.

Excel is useful for data visualization and organization, but when it comes to mobile data collection, it proves unwieldy. Data cells and input keys are small and can be difficult to press accurately. The display is nearly impossible to see in bright lighting

conditions on tablets with inadequate glare reduction. The grid format for information is inefficient on large plots due to the standard serpentine order of collecting. It can also be difficult to maintain the correct position in a spreadsheet when it is necessary to skip cells that do not have data to be input.

Many applications have been developed in the past decade to ameliorate these limitations. Some focus on global data management, allowing collaborators to pool their collective knowledge. Applications like Magpi (DataDyne Group) and Epicollect ([EpiCollect.net](http://EpiCollect.net)) provide tools for creating mobile data collection forms and include functionality like GPS location and photo uploads. While they support data collection, their primary goal is organized dissemination of data.

Reference guides are particularly well-suited to mobile application development. Some, like Plant-o-matic (Ocotea Technologies, LLC) and the iBird Guide (Mitch Waite Group), present the user with a powerful search tool with which they can identify a specimen. Other references, like Project Noah (Networked Organisms), rely on the contributions of ‘citizen scientists’ to report and identify sightings of fauna across the globe. The application serves as a hub for a community of users to share their findings or seek help from fellow researchers.

## 2.2 Speech Recognition

Speech interfaces have been successfully incorporated into video games, office applications, art pieces and vehicle consoles. In these diverse settings, speech interfaces are beneficial for different reasons. For vehicle consoles, having hands free operation that doesn’t take the driver’s eyes off the road improves safety. In office applications, the most commonly used feature is dictation. The computer can transcribe the user’s thoughts as they speak them, thus allowing users who think faster than they type to capture their message more quickly and efficiently. The main benefit of a speech interface for a video game is a wider command base. On console games in particular, there are sometimes not enough buttons to encompass the commands. As such, most games that utilize a diverse number of commands (for example, World of Warcraft) have to use the computer keyboard. The addition of a speech interface allows for the use of far more commands without needing extra buttons or button combinations.

The process of speech recognition involves three main steps. The first is to sample an incoming analog waveform to a digital representation. Next, this digital data is divided into distinct units of sound called phonemes and pauses. Finally, the resulting phonemes are run through an algorithm to determine the resulting text. The algorithm used differs between various speech recognition software, and can have varying levels of complexity depending upon the needs of the system.

All speech recognizers use a dictionary. A dictionary consists of a list of all the words a recognizer can distinguish alongside the combination of phonemes that make up that word. It is possible for a single word to have multiple phoneme combinations just as it is possible for words in a language dictionary to have several definitions. Figure 1 shows an excerpt from one such dictionary.

A speech recognizer might also consult a grammar when parsing a phrase. A grammar tells the recognizer the context in which a word can be used. For example,

Fig. 2 shows the grammar written for the Field Book augmentation. Much like a simplified version of a language grammar, it lays out the rules for when and how certain words are used. From this, the recognizer can better detect what words are being spoken by comparing them within the context of the phrase.

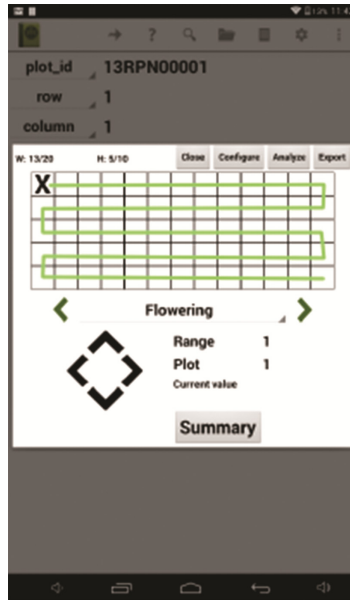
```
public <basicCmd> = <command>;
<command> = go forward | go back | <relocate> | <setTrait> | <query> | undo last | cancel;
<relocate> = <move> <location>;
<move> = go to | move to | i am at;
<location> = plot <number> | next plot | last plot;
<setTrait> = set <trait> to <number> <atLoc>;
<trait> = flower|height;
<atLoc> = at <location>;
<number> = <digits> <digits>* <digits>;
<tens> = ten|twenty|thirty|forty|fifty|sixty|seventy|eighty|ninety;
<teens> = eleven|twelve|thirteen|fourteen|fifteen|sixteen|seventeen|eighteen|nineteen;
<digits> = one|two|three|four|five|six|seven|eight|nine;
<query> = where am i | what row * | what column * | what plot * | what is <trait> set to;
```

**Fig. 2.** The grammar written for the *Field Book* augmentation. Much like a simplified version of a language grammar, it lays out the rules for when and how certain words are used. From this, the recognizer can better detect what words are being spoken by comparing them within the context of the phrase.

There are two main types of speech recognition: local and remote. In a local system, there is a dictionary file stored on the device that is referenced by the recognition algorithm. All computations are processed locally. These systems tend to be more accurate when dealing with a small dictionary and often have less sophisticated algorithms than remote systems. Our application only requires local speech recognition due to the small dictionary.

### 2.3 Field Book

Field Book is engineered specifically for field data collection. The application was developed with the needs of wheat researchers in mind. Users create a grid in Microsoft Excel, or other spreadsheet software, detailing the title or id (usually a number), row and column of each plot and import it to the tablet. Field Book then generates a map of the field. When creating a new data set, users specify the data points, referred to as ‘traits’, they will be collecting (i.e. flowering date, height, exertion), and the name of the researcher inputting the data. Field Book then allows the user to input the data per plot. Traditionally, researchers follow a serpentine pattern when collecting data; completing one row in ascending order, then following the next row in descending order. An example of this path is shown in Fig. 3. Field Book assumes this layout when progressing through a data set. Field Book’s visual design is high contrast with large text and large buttons to facilitate its intended use in bright sunlight. Inputting data is streamlined, making data collection faster and easier than traditional paper methods. Users can export their collected data in spreadsheet format, allowing them to make use of Excel’s mathematical capabilities.



**Fig. 3.** An example of a screen in the Field Book application.

Field Book maintains flexibility for users by allowing them to define the traits they wish to record for a specific crop. The user can create data points of many different types including numeric, date, text, photo and audio. This allows the user to fine-tune the process to suit the needs of different research projects.

### 3 Implementation

#### 3.1 Design Considerations

In designing an augmentation for Field Book, it was important to take into consideration the current implementation of the software. The application was designed for 7 inch tablets to maximize available screen space while keeping the device small enough to be easily portable. It was first developed for Android systems because the cost of these devices is lower and distributing the application is free and easy. This is in keeping with the goal of the “One handheld per breeder” movement. By keeping costs low, the application is available to more researchers. Android offers tablet models that are constructed to be durable, making them practical for field work.

A potential challenge faced by an outdoor speech recognition application is input sound quality. In practice, there will likely be environmental noise from wind, nearby roads or other interference. Distance of the user from the microphone (for speech recognition) or the speakers (for audio feedback) can reduce the user’s ability to properly communicate with the system. The user could hold the tablet closer to their head, but

this defeats the purpose of hands-free functionality. We used an inexpensive headset to provide quality audio input and output.

We assume that researchers will not have internet access while recording data in remote fields. As such, the data collection capabilities of the software have no internet dependency. Data dissemination is handled outside the application following data export.

The Field Book application is being introduced to both new and seasoned researchers. Prospective users run the gamut from avid technical gurus to traditional pen and paper enthusiasts. To account for this, developers have worked to make the interface as intuitive as possible, including an on-board tutorial.

We developed the speech augmentation for the Field Book application on a Neutab N7 Android tablet. The text and buttons added to the interface are large and high contrast to be seen easily in bright sunlight. The speech recognition and speech synthesis software used (PocketSphinx and Android Speech Synthesis, respectively) are available for free and operate locally without access to the internet.

### 3.2 Workflow Evaluation

We accompanied wheat researchers from Texas A&M on a survey of their project fields. On the trip, we noticed several inconveniences that could be ameliorated by a speech interface. First, the tablets require two hands to input data. This can be inconvenient for users who must also handle the plants or other equipment (like a measuring pole). The researcher also mentioned that the tablet touch screens tended to be less responsive in rain or mist, making data difficult to input. Inputting data also requires the user to look at the screen. It can be difficult to see the screen when the sun is very bright. With a speech system, input is handled via a microphone, minimizing the need to handle the device or view the screen.

We used this experience to lay out my concept for the interface. To ascertain the most intuitive setup, we framed the system as a kind of digital secretary. We asked the researchers to go over how they would relay their information if they had an assistant. Having one researcher call out information and one record it is common practice for large fields, as it speeds up the process. We made note of when the speaker would announce information, and when the recorder would ask for confirmation. In the case of a speech recognition system, particularly the less accurate local variety, it was necessary to assume this digital secretary was rather hard of hearing and would have to ask for confirmation more frequently than a human counterpart.

Researchers first state out loud plot number they are recording information. They then announce the trait to record and value of that trait. For example, in Texas there are five types of Wheat fungus. Plants are scored in six categories, no fungus or fungus type one through five. Plants with fungus are next scored with a percentage of disease from 2% to 100% in 20 point increments.

Researchers sometimes have to overwrite previous data as more information becomes available. For example, a researcher says that a particular plot must have flowered a day ago based on the appearance of the panicle (the loose, branching cluster of flowers found at the top of the plant) and the amount of pollen it has released. If the

researcher then scores a plant a few rows down that has released less pollen but clearly did not begin flowering today. The researcher records that the current plot flowered yesterday and edits the previous plot to say it flowered two days ago. Collecting data that requires estimation, like flowering date, is subjective and users must be able to go back and modify the data if they change their minds.

### 3.3 Audio Feedback

Since the user should not feel the need to look at the screen when using this interface, audio feedback becomes the primary way of communicating information. It is necessary for the user to know when the system is listening, when the system recognizes or fails to recognize a command, and when it requests confirmation. To accomplish this, the system provides audio feedback for each case. This system makes use of both speech and tonal feedback, depending on the situation.

The system is always listening, but only activates command detection when it is specifically addressed using the keyphrase “Field Book”. This prevents the system from trying to interpret everything it hears as commands. When the application detects the keyphrase, the system issues a notification sound to alert the user. Here, sound was employed rather than a spoken phrase to minimize the time between the user activating the system and speaking their command.

When the system recognizes a command, it issues a unique notification tone, then repeats the command it heard. For example, if the system recognizes the command ‘move forward’, plays the command recognized sound (a synthesized chime), then says “Moving forward to plot [next plot]”. In the case of any error, the system will first play an error notification sound (a descending pair of tones). If it fails to recognize a given command, it says “[command issued] is an invalid command”. If the command has a problem, for instance if the user attempts to move to a plot outside the map’s bounds, the system will speak to the specific problem, in this case “The maximum plot is [# of plots], cannot access plot [the plot the user attempted]”.

It is necessary for the researcher to receive audio feedback to ensure accuracy of data collected. For the trait command, the system announces the trait and value it is going to record, as well as the plot it is recording in. The system awaits feedback in the form of a ‘yes’ or a ‘no’ from the user prior to making changes in case the data is incorrect. The same occurs for the move command. Unlike other commands, the move forward and move backward commands do not await confirmation as they are used to move quickly between plots. The potential risk of falsely triggering these commands are combatted by their dissimilarity to other command phrases.

The user can request data such as the current row, column, plot, and trait values of the current plot using the commands detailed in the previous section.

### 3.4 What We Learned

In exploring this speech-based system, we worked out a series of do’s and do not’s for those considering the use of a similar system.



*Do expect a good speech-based control system to feel like a conversation.* Because this type of system requires the user to talk to it, and for the system to talk back, it inherits many of the characteristics of a traditional conversation. It can be more engaging to users than a passive point and click system. With a large enough command base, this type of system is also intuitive. The user can request a specific action, and the system will return the expected result.

*Do expect a hands-free experience.* Given the appropriate command base and audio feedback, the user does not have to physically interact with the system at all except for listening and speaking. This leaves the user free to use their hands and eyes elsewhere.

*Do the research for what type of recognizer your system will need.* Be aware of the requirements of the proposed system. Does it need a free or opensource software, or will it need a proprietary package? Know the priorities of speed, accuracy and flexibility. Will the developer define the grammar or use an open ended phonetic interpreter? A wider command base can mean less overall accuracy. Does the system require access to the internet? Internet dependent systems have access to more processing power and can handle larger dictionaries, but they can also be slower or less reliable depending on your connection.

*Do expect to have to update your system.* The first iteration (and likely the fifteenth) will not be perfect. Be prepared to make changes to the command base, the grammar, and the command interpreter many times to get it to a useable state.

*Don't expect the recognizer to be completely accurate.* Even the best recognizers have some degree of error. Always prepare your system for that eventuality and make sure it can handle miscommunications. It is far better to prepare for failures that never happen than to not be prepared for the one that does.

*Don't expect the recognizer to be instant.* Speech recognition takes time. The fastest systems still require a second or so to process. Speech controls are therefore not appropriate for situations requiring twitch controls such as in first person shooter games.

*Don't get discouraged.* Speech recognition is a conversation. It doesn't always go the way you'd expect. Be prepared to deal with the frustration of fine tuning the system to get it to a point where it is useable.

Overall, this system does what it was expected to do. It facilitates the use of the *Field Book* application as a hands-free tool. The user can move, set traits and "view" information without having to handle or look at the device.

## 4 Future Work

For future improvements of the system, I would like to continue to refine the accuracy of the recognizer. The incorporation of a remote recognizer could provide better translation while adding the requirement of continuous internet access. The system could also potentially incorporate user profiles, which would improve accuracy by training the system to a specific user's speech traits. I would also like to expand the commands the system is capable of handling, including the ability to navigate the main menu, load maps, and create new traits. More expansion possibilities include translation to other

languages, and dictation for notes. In an ideal setup, users could add customized commands through an intuitive interface rather than having to modify the code.

## 5 Conclusion

We introduced the AR Field Book application to both new and seasoned researchers. Prospective users run the gamut from avid technical gurus to traditional pen and paper enthusiasts. The AR application reduced the time needed to navigate menus and submenus. For example, to access a non-adjacent plot using the current version of Field Book, the user must expand a drop-down menu, open the map, wait for it to initialize, and then count the unlabeled cells to find the next test plot to grade. With the implementation of speech-based commands, the user can just say ‘Go to plot 285’.

Field Book currently allows users to record speech for later transcription. This feature is not used due to the costs in time or currency of transcription. With speech recognition, users have a written record to transmit to colleagues immediately.

## References

1. Allen, J., Hunnicutt, S., Carlson, R., Granstrom, B.: MITalk-79: the 1979 MIT Text-to-speech system. *Acoustical Society of America* 65.51 (1979): n. pag. AIPScitation. AIP Publishing LLC, 11 August 2005. Web. 29 September 2015
2. Davis, K.H.: Automatic recognition of spoken digits. *J. Acoust. Soc. Am.* **24**(6), 627–642 (1952)
3. Doolittle, W.E.: Recording Data. *Field Techniques*, 23 June 2015
4. Dudley, H., Riesz, R.R., Watkins, S.A.: A synthetic speaker. *J. Franklin Inst.* **227**(6), 739–764 (1939). Effective, Affordable, Reusable Speech-to-Text (EARS)
5. Flanagan, J.L.: *Speech Analysis Synthesis and Perception*, pp. 166–167. Springer, Heidelberg (1965). *Speech-to-Text (EARS)*. Electronic Frontier Foundation, n.d. Web, 05 November 2015. EpiCollect. Computer software. EpiCollect.net. Wellcome Trust, n.d. Web, 30 June 2015
6. Global Autonomous Language Exploitation. The Idiap Research Institute, 05 November 2015
7. GOOG-411 Team: Goodbye to an Old Friend: 1-800-GOOG-411. Web log post. Official
8. One Handheld Per Breeder. McKnight Foundation, n.d., 04 November 2015