

School of Computer Science and Informatics Cardiff University

Initial Plan Mobile application to assist Scrabble play

Author Timothy Sean Go McCarthy (C2056696)

> **Supervisor** Kirill Sidorov

Moderator

Xianfang Sun

1 Project Description

Scrabble is a word based game that is suitable for 2 to 4 players. It was designed in 1931 and is one of the best selling games of all time [1]. Players compete on a 15x15 board and accrue points by playing words using letter tiles from their rack.

Keeping track of the score in Scrabble is quite cumbersome. This is due to many factors such as bonus tiles, multiple words being created in one turn and bingo's. Having a mobile application to track the score of a game would make this process much easier. The players would just have to take a picture of the scrabble board each time a new turn is taken. This would also give players the option of revisiting old games they played and seeing the moves they made turn by turn on a digital representation of the board. Using the difference between the boards an algorithm can be used to calculate the score the turn player has gained.

To create the mobile application, I am planning to use React Native which is a JavaScript framework for creating iOS and Android applications. This project is going to rely heavily on computer vision to convert the picture of the Scrabble board to a digital representation. To attain this digital representation we must split the board into its 255 individual squares. To achieve this we can use the OpenCV library in Python and exchange information using a Flask web server (explained later on). After gathering the picture of each square, we now perform object classification on each of these squares. We must classify whether the square is blank or whether the square has a letter and which letter is being represented. To do this we can use a Convolutional neural network (CNN).

There are two possible solutions to this problem:

- We train a CNN using TensorFlow.js on a desktop and import the trained model into the application and run the trained CNN on the mobile device.
- We train a CNN using TensorFlow in Python and create a web server in Python using flask that contains the CNN. The app sends the picture of the board to the web server using RESTful api. The web server will calculate and then return a representation of the board and the updated score.

I believe both solutions have their upsides. The second approach means none of the calculations have to be computed on the mobile device. I would usually go for the first approach as modern mobile phones have reasonable computing power and shouldn't have too much of a problem running a trained CNN. However we are already creating a web server to exchange the data from the computer vision before the CNN and it will be easy to simply calculate the result of the CNN on the web server and the scoring algorithm as well.

2 Aims and Objectives

The aim of this project is to create a mobile application that can track the score of a Scrabble game for each player. This can be realised through the objectives below. Likely risks of these objectives is also considered.

- Create a computer vision algorithm that is able to manipulate the picture of the board in such a way that it is able to be split up into the 225 different squares of the Scrabble board. The biggest risk of this is an unreliable algorithm due to variance in the input data. The picture can be taken from various angles and in different levels of lighting.
- Train a CNN with the ability to classify a square of the scrabble board. The CNN should be able to classify whether the square is empty, whether it has a tile and if it does have a tile which letter the tile is representing. The largest risk for training the CNN is not having the quantity or quality of training data necessary. Missing either of these can lead to a CNN that is unreliable.
- Create an algorithm that is able to translate consecutive board states into Scrabble scores.
- Create an application where the user is able to create a new game and be prompted to take pictures of the board. After taking a picture of the board they should see an update of the score and a prompt to take a picture of the next board state.

3 Feasibility

Hardware is not a problem for this project as I own both an Android and iOS device to test the final application on. However, there are legal issues around creating a app so closely linked to a trademarked game like Scrabble.

We are using the Scrabble dictionaries to use for word validation. This can be used in the computer vision side and also to check if the player has played an invalid word. The Scrabble dictionaries are copyrighted so we will have to do a reference to them somewhere within the application.

The Scrabble logo must not appear anywhere in the application. This is due to copyright over the logo of Scrabble. In the same vein we cannot use the word Scrabble in name of the application. I will not be using the word Scrabble in the application in any form. If the application was every made into a commercial product, the copyright holders of Scrabble, could argue that it is not under nominative use because the app could be seen as a potential competitor to Scrabble.

4 Work Plan

Task	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12
Initial plan												
Background Research												
Computer vision/CNN: Creating web server												
Computer vision: Manipulating board												
Computer vision: Splitting into squares												
CNN: Creating training data												
CNN: Training												
CNN: Testing												
Mobile App: Testing interaction between app and web server												
Mobile App: Create basic interface												
Mobile App: Develop scoring algorithm												
Mobile App: Create ability to start and save games												
Final Report: Write first draft												
Final Report: Getting feedback + rewriting												

Above is a Gantt chart to track the progress of the project through the twelve weeks of working time. There are three deliverables Computer vision, CNN and Mobile App. These are the three areas of focus for the project and I have given myself three weeks to work on each one. The weeks highlighted in green is when I expect to complete the given deliverable. I will also be having a meeting with my supervisor every week.

Below are all the deliverables, the date they will be completed and what needs to be done to complete the deliverable:

- Computer vision: Week 4
 - Manipulating board Using OpenCV to detect the board using features. Manipulate the board to match a prepared image of the board.
 - Splitting into squares Adjust the angle and position board to fit a prepared 15x15 grid. Create an image of each square.
- Convolutional neural network and the web server: Week 7
 - Creating training data Create small dataset of images of letters. Manipulate these images using TensorFlow to create larger dataset.
 - Training/Testing Train, test and retrain until desirable accuracy is reached.
 - Creating web server Create a flask web server that when requested by the mobile application can compute the score and return a representation of the board.
- Mobile application: Week 10
 - Testing interaction between app and web server This will be developing and testing that the application is able to send and retrieve information from the web server.
 - Create basic interface Create ability to take photos in the application with simple instructions.

- Develop scoring algorithm Create basic scoring algorithm using the difference between board states following Scrabble rules.
- Create ability to start and save games This will include history of moves made.

The final report will be completed in the last 3 weeks.

5 References

Stephey, M.J. 2008. A brief history of Scrabble. TIME.
Available from: https://content.time.com/time/arts/article/0,8599,1867007,00.html