Audio-augmented Arboreality: Wildflowers and Language

Rahaman, H. and Johnston, M. and Champion, E.

Before colonisation, there were over 250 languages spoken in Australia. Today only thirteen Indigenous languages are still being taught to children (AIATSIS 2019). Language has an important part to play in cultural maintenance and 'closing the gap' in terms of First Peoples' cultural heritage, identity, and sense of belonging. In this work, we aim to develop an engaging and easy way to teach and learn the local Indigenous names of wildflowers using a mobile device. This paper presents the development of a phone application that runs on a local machine, recognises local wildflowers through its camera, and plays associated sounds and displays associated text in the Noongar language. The prototype mobile application has been developed with MobileNets model on the TensorFlow platform. The dataset is derived from Google searches, while the sound files are generated from label text by running an apple script. UI and interactivity have been developed by using Vuforia and the Unity game engine. Finally, the Android Studio is used to deploy the app. At this point in time, the prototype can only recognise ten local flowers, with 85%~99% of accuracy. We are working with a larger dataset towards developing the full application.

Keywords: wildflower, object detection, machine learning, TensorFlow, Noongar language.

1. Introduction

The United Nations declared 2019 to be the International Year of Indigenous Languages. This was an opportunity for Australia to consider the future for Indigenous languages. It is estimated that before colonisation there were over 250 languages spoken in Australia, whereas today only thirteen Indigenous languages are still being taught to Australian children (AIATSIS 2019). *Noongar* is the language spoken by the First People of Australia's south-west. It is a language that is "in rapid recovery (Centre 2019a), and is one of the languages currently taught in schools. The Noongar language is supported in schools by the WA Education Department but is offered in relatively few schools and more likely to be taught outside the metropolitan area. By expanding the number of Noongar language teachers and speakers, the number of schools teaching language can grow – an essential first step in reviving and preserving the Noongar language.

After colonisation in 1788, the Australian government had a program of assimilation for First Peoples that actively worked to destroy language and culture; Aboriginal and Torres Strait Islander people were forbidden to speak the language. Australian governments, businesses, and educational institutions are now undertaking a journey of reconciliation with ambitions to 'close the gap' of Aboriginal disadvantage by strengthening cultural identity. Language has a vital part to play in that journey as it has the potential to connect First Peoples and non-Indigenous people to First Peoples culture and to the land we share. Noongar people, like all First Peoples, have an intimate connection to their land and language helps to affirm that connection (SWALSC, n.d.). Language acquisition also contributes to Aboriginal children's engagement and success in school, but the benefits of teaching language extend beyond the classroom and First Peoples alone. Indigenous languages contribute to national Australian culture and identity (Purdie et al. 2008). Mühlhäusler and Damania (2004) explain the broader 'public good' that comes from maintaining Indigenous languages:

> "The idea that linguistic diversity should be preserved is not a sentimental tribute to an idealized past, but part of the promotion of sustainable, appropriate and empowering development. The problem of language death thus is a 'good' problem, in that solving it would mean solving many other urgent and interrelated problems at the same time". (p36)

In this paper, we present the development of a prototype mobile phone application that can help the user to learn the Noongar names of wildflowers in a fun and easy way; a step forward to teaching and learning the Noongar language in the broader community. Rather than presenting a glossary and quizzes, or a translation service for learning language (Estaff 2017); technologies such as augmented reality (AR), virtual learning environment (VLE), and mobile learning can offer a better environment in which learning activities can occur anytime and anywhere, and can benefit both learners and educators (Rahman, Yahaya, and Halim 2014; Kiat et al. 2016). These technologies are now being used in formal learning from primary school to university level (Akçayır and Akçayır 2017).

Artificial intelligence is the domain where machines are given the capability to learn new instances of data through learning, which is referred to as Machine Learning (ML). As a subset, Deep Learning goes into further accuracy and uses neural networking technology. Deep learning enables computational models (with multiple processing layers) to learn representations of complex situational data and can then make more precise decisions. Such techniques can significantly improve the methods of speech recognition, recognition of visual artefacts, detection of objects, and many other domains such as genomics (LeCun, Bengio, and Hinton 2015). Various data such as audio, image, human movement, which are challenging to analyse computationally without ML. Artists, Musicians, and creative practitioners are therefore using ML for practice, teaching-learning, and research (Fiebrink 2019). Learning analytics and decision making for more suitable, accepted, and better use is being promoted by using ML for personalised learning (Kurilovas 2018).

In this paper, we present the application of object recognition technique (using a transfer learning method to re-train an existing model, more in section 3.2) in the development of a mobile phone app prototype that recognises 3D objects (i.e. flowers), plays associated sounds (flower names in the Noongar language), and displays associated text (the flower name in Noongar language); thus helping the user to know the Noongar name of wildflowers. For example, if when using the app the phone's camera sees a ‘Banksia’ flower (Banksia attenuata) and the user taps on the screen, the phone will say the word “boolkaala and at the same time will display the name on the screen (figure 6) in real-time. In this way, the app shows the potential to be useful as an educational tool for people learning the Noongar language. Pronunciation of words can be challenging to master for beginner language speakers because Noongar pronunciation is markedly different from English. Hearing a Noongar Language speaker say the word correctly is an important first step to learning the language (Centre 2019b). Phones are portable and convenient devices that are easier to access than a dictionary or textbook. The phone app is likely to be an attractive learning tool for children in particular as it can turn learning flower names and related local terms into an engaging shared experience.

Education experts recognise that individual students have individual and diverse learning styles. Developmental psychologist, Howard Gardner, proposes that students have different ‘intelligences and therefore respond to different ways of learning (Pritchard 2017, 58). He identifies nine different types of intelligence, or learning styles, and describes different pedagogical approaches for each. Several of these learning styles can be supported by an app such as the one described: linguistic intelligence responds to literary resources such as reading, writing and speaking; spatial intelligence responds to pictures and diagrams; kinaesthietic intelligence responds to movement, touch and practical apparatus; and naturalistic intelligence enjoys engagement with the natural world (Pritchard 2017, 58; Gardner 2011). As a student moves around in nature, touching the plants and flowers, listening to the spoken words in the app, reading the words on the phone screen; several different learning styles can be simultaneously engaged.

Using visual object recognition techniques based on neural networks, the proposed solution explores a tailored visual database. From a software point of view, we have implemented the objects (i.e. wild flowers) with real-time recognition by using TensorFlow (https://www.tensorflow.org/), an open-source framework developed by Google. TensorFlow supports different computing platforms, mobile devices, IoT (internet of things) and embedded devices. Pattern recognition is one of TensorFlow's major application areas, other areas include object detection of models, video classification, and other related fields.

The main objective of our work is to build a prototype phone app, and that process began with ethical considerations about cultural awareness and respect. The project was discussed with the University Noongar Elder in Residence and a Noongar language teacher, with whom the authors have trusted relationships. Once the ethical approval is done with the university and the app has been developed to the point of reliability we will consult further with Noongar language experts to confirm the spelling, the correct use of words, and to record the spoken words with a Noongar language speaker who has been nominated by Noongar language experts. Once the app has been developed to the point of readiness for community use, it would be released and managed under the control of Elders and/or the Noongar Boodjar Language Centre.

The technical process starts with the development of a database, which includes the selection of objects, collecting photographs of the objects, and recording the phonation of that object (i.e. audio files). A key element of the work is to build a custom convolutional neural network (CNN) trained model by using TensorFlow. The model has been re-trained using new datasets (5519 images of ten flowers) with the MobileNets model on the TensorFlow platform. We have used Vuforia (https://www.ptc.com/en/products/vuforia) to manage camera feed behaviour, the Unity 3D game engine (https://unity.com/) to develop the prototype, and finally, the Android Studio to deploy the app (.apk file) to an android mobile phone.

The main contribution of this paper is to present a method of teaching and learning wildflower names using Noongar language, followed by the design development choices together with an initial implementation of the proposed prototype. Hence, the next section of this paper describes the background and related works. Section two presents the development of the dataset. The following section three describes the system overview and various steps in the development of the prototype. The early implementation for Android and basic performance evaluation is presented in section four, and finally, conclusions are drawn in section five.

2. Present works/studies/ related works and technologies

Augmented reality (AR) refers to the technology that overlays virtual objects (augmented components) into the real-world. These virtual objects then appear to co-exist in the same space as objects in the real-world (Azuma et al. 2001). During the 1990s, AR technology was developed as a training tool for air force pilots (Caudell and Mizell 1992). Currently, AR technologies are widely used in educational settings and research purposes because they no longer require expensive hardware or sophisticated equipment (Akçayır and Akçayır 2017), and thanks to the development of high-performance computers and mobile devices.

Previously AR technologies in education mostly relied on markers, images that are hard-coded into the application, which trigger some kind of action (Godwin-Jones 2016). TANGO (tag added learning objects) (Beaudin et al. 2007), HELLO project (Liu 2009), Zooburst (Mahadzir and Phung 2013) etc., are examples of such kind. Markers can be attached to any surfaces such as walls, embedded in books, and can be tagged to an object. Whenever a marker is recognised by an AR app that's come into view through the user's camera, an action is triggered, and information (such as text, image and sound) is displayed on the screen. This is how AR is being used by publishers to supplement and update information in textbooks or in other print materials (Hawkinson 2014). Marker-based AR had been used widely in helping teach vocabulary; learning characters in Chinese, facilitating English to Tamil translation (Rose and Bhuvaneswari 2016), learning Filipino and German vocabulary (Santos et al. 2016), assisting in pronouncing new English words (Solak and Cakir 2015), or generating flashcard interactions for English (Li, Chen, and Vorvoreanu 2015).

GPS supplied locational information with combined data supplied by other means from the handheld device (such as camera feed, gyroscope, compass and accelerometer) become another way to generate events from AR. The mobile game Pokémon Go by Niantic (July 2010) is one of the most successful examples to use such place-based technology and AR. Tapping a Pokémon switches the display on an AR view, and the virtual creature appears on the live scene (real-world view) which is captured through the player's camera. Pokémon GO was not designed for language learning. However, studies from Godwin (2016) shows that the game has been experimented and used by various educators to enhance language learning, especially for learning English. Mentira is another example that uses location-based mobile AR technology for learning Spanish language (Holden and Sykes 2013).

Computer vision, and object recognition, in particular, has made tremendous advances in the past few years. The PASCAL VOC Challenge (Everingham et al. 2009), and more recently the Large Scale Visual Recognition Challenge (ILSVRC) (Russakovsky et al. 2015) based on the ImageNet dataset (Deng et al. 2009a) have been widely used as benchmarks for numerous visualization-related problems in computer vision, including object classification. In 2012, a large, deep convolutional neural network achieved a top-5 error of 16.4% for the classification of images into 1000 possible categories (Krizhevsky, Sutskever, and Hinton 2012). In the following three years, various advances in deep convolutional neural networks lowered the error rate to 3.57% (Krizhevsky, Sutskever, and Hinton 2012; Simonyan and Zisserman 2014; Szegedy et al. 2015). While training large neural networks can be very time-consuming, the trained models can classify images very quickly, which makes them also suitable for consumer applications on smartphones.

Plant and flower identification by conventional means is difficult, time-consuming, and (due to the use of specific botanical terms) is often frustrating for novices and learners. This creates a hard-to-overcome hurdle for beginners interested in acquiring species knowledge. However, recently; ML, image processing and pattern recognition techniques, have been introduced into plant taxonomy which eventually paved the way to address limitations in beginners' identification abilities (Wäldchen et al. 2018).

Automated species identification mostly focuses on the development of feature detection, extraction, and encoding methods for computing characteristic feature vectors. Deep learning CNNs, on the other hand, automatically discover a statistically suitable image representation (similar to a feature vector) for a given problem. The fundamental concept of deep learning is a hierarchical image representation composed of building blocks with increasing complexity per layer. From a machine learning perspective, plant identification is a supervised classification problem, and a comparative study from Wäldchen et al. (2018) shows that CNN classification performance is far better in comparison to traditional and shallow learning approaches.

There is a sizable number of mobile applications available in the market today that can identity plants and flowers (BalconyGardenWeb 2020). These apps use different methods for identification. For example, Plantifier, Garden Tags, iNaturalist and FlowerChecker are based on identification by a broad community of garden and plant enthusiasts, and sometimes an international team of experts.

They allow users to upload a photo of an unknown plant and the community/expert then helps to identify that specific plant. The team tries to respond as soon as possible, but identification usually takes several minutes or hours. LeafSnap, Garden Answers, PictureThis, PlantSnap, PlantFinder etc. on the other hand, use visual recognition and AI technologies to help identify tree species from photographs. The user needs to be online, take photos and upload to the cloud-based backend service to get an immediate result. Some of these apps are based on paid service and offer their content translated into different languages.

For example, Google Lens (https://lens.google.com), using the Google search engine, can bring up relevant information related to objects it identifies using visual analysis based on a neural network, and it can also be used for flower/plant identification. Users need to take a photo, upload it, and Google will retrieve and display information related to the identified plant.

At the time of writing, we could not find any published resource that evaluates the performance and results of an image-based identification engine. There are cloud-based solutions for mobile devices, such as IBM Watson, AWS, Google etc. with vision APIs for image processing. However, these commercial services run on a subscription-based business model and may not be scalable for low budget developers and organisations. Most importantly, to get the service, a user needs to connect to the cloud for every request in real-time this requires an internet connection and mobile data. There are also native solutions for running ML models on Android and iOS devices, but they are rarely cross-platform supported and out of the scope of our present study. Instead, we will focus on how image identification can help people learn wildflowers with their indigenous names. The following sections describe the development of the prototype app.

3. System overview and development

The main objective of the project is to develop a prototype mobile app that runs locally and helps the user to learn the Noongar names of wildflowers while identifying them, it shows the Noongar name in the text, and speaks the Noongar name (i.e. plays the associated pronunciation). Here image processing with machine learning can do the heavy lifting for us.

In general, there are three levels of image processing that machine learning is used for. Firstly, image classification, where it can tell generically what is in the image. Then there is object detection where it can draw a boundary box around the image. And, lastly, there is image masking where it retrieves an exact outline of the object in the image. In this paper, we have tried to find a solution based on an image classification method. Our prototype application, which can run on a local machine, is built on the MobileNets model on the TensorFlow platform, with the Unity 3D game engine and the Augmented Reality application Vuforia (figure 1).

There are several game engines which support both 2D and 3D interaction. Unity 3D game engine has been selected because of its cross-platform support. The engine offers a primary scripting API in C# and supports a free licence for personal use with less than $100,000 annual income. Besides, the plug-in (TensoFlow#, explained later) which has been used for developing the app, only works with Unity. There are several augmented reality software development kits (SDK) for mobile devices that enable the creation of augmented reality applications and support real-time object tracking through camera feeds. These SDKs can be proprietary such as Vuforia, Wikitude, Maxst etc. or free and open-source such as EasyAR, AR.js, ARKit, ARCore etc. Vuforia has been selected here for its cross-platform support and long reputation in the industry. Unity 3D can directly develop and deploy an app (*.apk) to an android device. However, to avoid any potential compatibility issues and for smooth deployment, the Android Studio has been used.

Fig. 1. The system framework of the prototype

3.1 Preparing the datasets

Flower classification is a major concern not only for the domain of Botanical research but also for Image Processing through Machine Learning (Gavai et al. 2017) because of both the number of species and their similarities. There are more than 12,500 flowering species in Western Australia (Geographic 2017), and people often get confused because of their similarities. Remembering local names consequently is a big challenge. Having a fast and accurate flower classifier may, therefore, help the general public, people, students, as well as professionals, to identify and remember both the local Indigenous and non-Indigenous name of a flower.

However, there are some significant challenges in flower classification, such as their complex background as an image, and numerous similarities between different species of flowers. Most importantly, we cannot just depend on a single feature, such as shape, colour and texture to distinguish each flower. Due to the shades of colour, scale, shape or even the user's viewpoint, the system may be confused and falsely recognise or identify the same flower as a different species.

Until now, an expert observer or trained professional was needed to observe the growth habit and habitat, botanical structure and other features of a flower, then compare those features with other flowers, to ultimately determine the flower type and identity (Gavai et al. 2017). A suitable alternative has, however, become possible due to the development of Machine Learning. A large dataset (images) therefore is required to train the machine to retrieve the correct image and associated sound. To develop this working prototype, we have selected ten species of local wildflowers (Solanum/Bush tomatoes, Tjunguri/Lily, Yoont djet/Daisy, Berrung/Grevillea, Biara/Banksia, Kara/Orchids, Kurulbrang/Kangaroo Paw, Kwowdjard/Bottlebrush, Mindaleny/Wattle and Pulgur/Hakea) and collected a total of 5519 images from Google Image Search. Images are harvested through a python script and saved locally under separate folders (figure 2). Automatically harvested images are then manually authenticated and validated for any discrepancy, low resolution, wrong labelled image and several species in the same photo.

Fig. 2. Collected images of Banksia/Boolkaala from Google's search page.

3.2 Re-train the ML model

TensorFlow is a framework originally developed by the Google Machine Intelligence research organization for conducting machine learning and deep neural networks (DNN) research. However, the system can be applied to other various domains (Vishnu, Siegel, and Daily 2016). The main functionalities of the framework in the computer vision field are based on specialized convolutional neural networks (CNN) stored into visual databases, called *models* (Mulfari, Longo Minnolo, and Puliafito 2017). Specifically, the object recognition process leverages a pre-trained *model* (currently Inception V4), based on a dedicated CNN built by Google (Szegedy et al. 2016). This *model* has been trained on more than a million images from ImageNet database (http://www.image-net.org/) and can classify visual data into 1000 object categories, such as appliance, bird, food, plant, vehicle and many animals (Deng et al. 2009b).

'Transfer learning' is a method of machine learning that uses the existing knowledge learned from one environment, applied to solve a new problem in a different environment that has some relationship with the previous one. Bearing in mind the Inception's architecture, its final layer can be *'re-trained'* to allow the neural network to recognize new objects (such as flowers in our case) while leaving the lower layers intact. Following the guidelines available on the TensorFlow website (Codelabs 2017) and (Wei and Katariya 2020), we have organized our visual datasets and *re-trained* the model based on the pre-trained 'MobileNets' model.

Graph variables are not maintained in TensorFlow. Whenever the system runs a TF Python session, it rebuilds the graph from scratch. In order to use a pre-trained graph after retraining, it needs to be saved (or frozen). Saving a graph will save the graph nodes and edge weights at a certain "checkpoint", and it can be restored and used later, at the time point it has been saved (Hanson 2018). However, in order to properly import the saved graph into Unity, we must save it in *.bytes format.

The procedure has been launched on a single desktop environment on python (anaconda) with TensorFlow (ver. 1.7). The *re-training tool* writes out a version of the existing model while adding a final layer *re-trained* to out categories in a *.pb file (Protocol Buffers or protobuf graph file), while labels are stored into a text file (*.txt). The hardware platform was an Apple iMac, processor 3.2GHz Intel i5, memory 16GB.

3.3 Preparing the sound/audio files

This model is re-trained on the given dataset, which has six classes (six folders for each flower). The re-training process extracts all the class labels that are stored in the label *.txt file. Every time an image is classified, we want the app to speak/play the word. The OSX operating system has a built-in text to speech conversion tool/application. An apple script has been prepared to open the image label text file and convert each line to speech, and then save the audio file to another folder. After running this script, it has produced all the audio files that match the names of each label in the text file (figure 3).

Fig. 3. An Apple script that produces an audio file from label text.

3.4 Setting up ML-Agent TensorFlowSharp plugin for Unity

In late 2017, Unity 3D introduced Machine Learning (ML)-Agents, an open-source Unity plug-in that allows users to train and use “intelligent agents” in Unity applications. The TensorFlow sharp plug-in from GitHub (https://github.com/Unity-Technologies/ml-agents) was first downloaded and then imported as assets into the Unity environment.

However, in order to properly import the saved graph into Unity, the *.pb file needs to be converted in *.bytes file format. Simply changing the extension of the file worked for us. In the player settings, it needs to change ‘enable_Tensorflow’ in the scripting define symbols input, and the script runtime version must change to .Net 4.x equivalent. After this step, the graph file (*.bytes file), image label text file (*.txt) and sound files (*.aiff) were imported into Unity as assets.

A C# script was created to classify, process and call various objects (figure 4.a). Whenever we feed an image into the model, it must go through some specific processing. At a high level, this script loads the label file and reads it into a list of strings. Every time the screen is 'clicked' (as user input); a processed image gets called, which creates a tensor from an image. The input tensor gets fed to the model and gives an output tensor containing all the probabilities of each label appearing in the image. Additionally, the input image must get scaled and cropped. That is why a texture tools script has been applied. The TF Sharp also uses a decode JPEG function that transforms the image before it gets sent to the model.

For a simple GUI visual appearance, a little drop-down box for the message has been used. A script for mask behaviour handles the movement (figure 4.b). And the camera feed behaviour retrieves the pixels of the camera feed from the Vuforia.

Fig. 4. Screenshot of the C# script and Unity working scene development.

4. Implementation and testing

Exporting the app to an android device first, it needs to change the build settings in unity. While switching the platform to Android, it needs to make sure that the current scene is added to the build, specify a bundle identifier and uncheck Android TV. Later, the build was exported as *.apk and the Android studio was used to deploy the app to a mobile phone (Samsung S10).

The trained model has been tested both inside the MobileNets platform and in the outdoor environment. We got a correctly classified flower with an accuracy value between 85% and 99% using MobileNets (figure 5). Similar accuracy was also found in the outdoor environment/settings (real-world situation) (figure 6). However, the resulting value changes from case to case, as there is some arbitrariness in the training process. Additionally, if we could provide more images to the datasets, we would expect a higher precision rate.

Fig. 5. Result showing testing accuracy of the retrained model (graph).

Fig. 6. Outdoor testing of the prototype

It is worth mentioning, the rankings provided here are based on the model trained dataset and do not equate to accurate outputs for the model. It has been discussed earlier (section 3.1) that, flower identification does not depend on a single feature but rather a combination of shapes, colours and textures. Furthermore, shades of colours, scale, lighting, shapes and camera viewpoint may often confuse the recognition system and lead false positives results.

5. Discussion & Conclusion

This paper aimed to develop an engaging and easy way to teach and learn the local Indigenous names of wildflowers using a mobile device. In this, the paper has presented a workflow of developing mobile phone application that runs on a local machine, recognises local wildflowers through its camera, plays associated sounds (local Indigenous name) and displays associated text (local/Indigenous name in Noongar language). There are a couple of plant and flower detection applications and commercial object recognition services such as Amazon, Microsoft and IBM in the market. However, the challenge here was to support a similar function/service running on a local device without an internet connection. Developing a dataset of local wildflowers, re-training the pre-trained model and playing audio of the flower names are the other significant challenges. The prototype mobile application has been developed to a stage that it can recognise ten wildflowers and has been developed with MobileNets model on the TensorFlow platform. While using customised datasets with associated sound files, the UI and interactivity have been developed by using Vuforia and Unity game engine. Finally, the app has been deployed to a mobile device through the Android Studio.

The app was demonstrated to two Noongar women who are actively engaged with language teaching and revival. Ingrid Cumming is the Noongar cultural advisor to Curtin University, and is the creator of another Noongar language app. She said this about the wildflower app:

> Nidga Kwop! This is good. The digitisation of Noongar language and knowledge is another popular format to continue kadadjiny koorl – or the journey of knowledge. This app will be a great tool and resource for people to connect and learn more about Noongar language. The interactive component as well – people can go out on country and explore, search and find, and learn – its a great concept. Having the Noongar voice would be great – and a mix of young and old fellas speaking.

Merinda Hansen is a Noongar language teacher in a Perth suburban primary school:

> Mooditj! It's going to be an amazing app. I think its something that we really need because so many people want to hear the Noongar names for plants – and I don't know all the plants. For people to get out there and see it on their app – just point a phone – technology is an amazing thing, but to have something in Noongar is even greater.

Identifying flowers is challenging because shape, colour and texture; as well as shades of colour, scale, lighting, viewpoint, nearby leaves etc. play a significant role and may confuse the system. This false positive can recognise or identify the same flower differently. More images in the database for an improved model and testing with various lighting conditions can help to get a better sample of the true reliability of identification for the model. Augmented Reality, ML and associated technologies are rapidly improving, and new tools are evolving. Instead of using Vuforia (because of its high pricing), in future, we are interested in using open source alternatives such as AR.js with Unity Machine Learning Agents SDK (ML-Agents). Besides, we are interested in the inclusion of voice/sound from a native speaker, the English name of the flowers side-by-side of their local/native name, adding more species and improvement of the GUI in the next version of the prototype.

The Noongar country is located in Western Australia which is also sometimes referred to as “The Wildflower State”. Thousands of tourists and local people drive many hours into the countryside, often beyond an internet connection, to celebrate the spectacular Western Australian wildflower season each Spring. Further development of this app would provide an opportunity for the wider community to engage with the language, the land and the culture of one of Australia’s First Peoples. Adapting the app for other First Nations languages is also a possibility.

Noongar language is intimately connected to the land and the natural environment and so overcoming the technical challenges of using the app in the less predictable natural landscape is essential to its success as a teaching and learning resource for Noongar language and other First Nations languages. Unlike the built environment of the modern world, with its geometric and uniform shapes, the natural world rarely produces any object in a uniform way. Each flower, tree and animal is unique and might be discovered across a diverse range of landscapes. Developing the app to recognise each unique shape and object will require considerable work. If successful, use of the app can be extended to include many more objects, or nouns, that will extend its capacity beyond the flowers featured in the prototype. An app such as this cannot replace more traditional methods for teaching language because features such as sentence structure, verbs and adjectives are not recognised. However, given the wide use of mobile phones by school children and adults alike, it can prove useful as a teaching and learning resource.

Ideally, the development of this prototype into a widely accessible and easily used app could see it delivered into the hands of millions of mobile phone users as a powerful step towards improving the health of Indigenous languages and nurturing not only Indigenous identity, but also Australian identity. To encourage active participation in extending the knowledge base for this phone application, we believe incorporating crowd-sourcing is worthy of future consideration.

References

AIATSIS. 2019. "Indigenous Australian Languages : Celebrating 2019 International Year of Indigenous Languages." Australian Institute of Aboriginal and Torres Strait Islander Studies, Accessed 23 March. https://aiatsis.gov.au/explore/articles/indigenous-australian-languages.

Akçayır, Murat, and Gökçe Akçayır. 2017. "Advantages and challenges associated with augmented reality for education: A systematic review of the literature." *Educational Research Review* 20:1-11. DOI: https://doi.org/10.1016/j.edurev.2016.11.002.

Azuma, Ronald, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. 2001. "Recent advances in augmented reality." *IEEE Computer Graphics Applications* 21 (6):34-47. DOI: http://doi.org/10.1109/38.963459.

BalconyGardenWeb. 2020. "Best Plant Identification Apps 2020." Balcony Garden Web, Accessed 27 Oct 2020. https://balconygardenweb.com/best-plant-identification-apps/.

Beaudin, Jennifer S, Stephen S Intille, Emmanuel Munguia Tapia, Randy Rockinson, and Margaret E Morris. 2007. Context-sensitive microlearning of foreign language vocabulary on a mobile device. Paper presented at the European conference on Ambient intelligence, Darmstadt, Germany, 7-10 November.

Caudell, T. P., and D. W. Mizell. 1992. Augmented reality: an application of heads-up display technology to manual manufacturing processes. Paper presented at the Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences, Kauai, HI, USA, 7-10 January.

Centre, Noongar Boodjar Language. 2020. "A Short History of the Noongar language." Noongar Boodjar Language Cultural Aboriginal Corporation, Accessed 25 Sept. https://noongarboodjar.com.au/history/.

———. 2020. “Spelling and Punctuation Guide.” Noongar Boodjar Language Cultural Aboriginal Corporation, Accessed 25 Sept. https://noongarboodjar.com.au/spelling-and-pronunciation/.

Codelabs, Google. 2020. “Tensorflow for poets.” Codelabs, Accessed 17 Feb 2020. https://codelabs.developers.google.com/codelabs/tensorflow-for-poets/#0.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009a. Imagenet: A large-scale hierarchical image database. Paper presented at the 2009 IEEE conference on computer vision and pattern recognition, Miami, FL, USA, 20-25 June.

———. 2009b. “Imagenet: A large-scale Hierarchical Image Database.” In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248-55. Miami, USA: IEEE.

Estaff, Remezcla. 2020. "From Náhuatl to Guaraní: 5 Apps to Help You Learn Indigenous Languages." REMEZCLA. https://remezcla.com/lists/culture/indigenous-language-learning-apps/.

Everingham, Mark, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. 2009. “The Pascal Visual Object Classes (VOC) Challenge.” *International Journal of Computer Vision* 88 (2):303-38. DOI: 10.1007/s11263-009-0275-4.

Fiebrink, Rebecca. 2019. “Machine Learning Education for Artists, Musicians, and Other Creative Practitioners.” *ACM Transactions on Computing Education* 19 (4):1-32. DOI: http://doi.org/10.1145/3294008.

Gardner, Howard. 2011. *Frames of mind: The theory of multiple intelligences*: Hachette, UK.

Gavai, N. R., Y. A. Jakhade, S. A. Tribhuvan, and R. Bhattad. 2017. “MobileNets for Flower Classification using TensorFlow.” In *2017 International Conference on Big Data, Iot and Data Science (Bid)*, 154-8. Pune, India: IEEE.

Geographic, Australian. 2020. “The wildflowers of Western Australia.” Accessed 27 Oct 2020. https://www.australiangeographic.com.au/topics/science-environment/2017/05/wildflowers-of-western-australia-3/.

Godwin-Jones, Robert. 2016. “Augmented reality and language learning: From annotated vocabulary to place-based mobile games.” *Language Learning & Technology* 20 (3):9-19. DOI: http://doi.org/10125/44475.

Hanson, Sara. 2018. “TensorFlow + Unity: How to set up a custom TensorFlow graph in Unity.” In.: Good Audience.

Hawkinson, Eric. 2014. Augmented reality enhanced materials design for language learning. Paper presented at the The Asian Conference on Technology in the Classroom 2014, Osaka, Japan, 17 April.

Holden, C, and J Sykes. 2013. “Place-based mobile games for pragmatics Learning.” In *Technology in interlanguage pragmatics research and teaching*, edited by Naoko Taguchi and Julie M. Sykes, 1-15. Philadelphia, USA: John Benjamins Publishing Company.

Kiat, Lee Boon, Mohamad Bilal Ali, Noor Dayana Abd Halim, and Halijah Binti Ibrahim. 2016. Augmented reality, virtual learning environment and mobile learning in education: a comparison. Paper presented at the 2016 IEEE Conference on e-Learning, e-Management and e-Services (IC3e), Langkawi, Malaysia, 10-12 October.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. Paper presented at the Advances in

Neural Information Processing Systems 25 (NIPS 2012), Nevada, USA, 3-8 December.
Kurilovas, Eugenijus. 2018. “Advanced machine learning approaches to personalise learning: learning analytics and decision making.” *Behaviour & Information Technology* 38 (4):410-21. DOI: http://doi.org/10.1080/0144929x.2018.1539517.
LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. 2015. “Deep learning.” *Nature* 521 (7553):436-44. DOI: 10.1038/nature14539.
Li, Shanshan, Yang Chen, and Mihaela Vorvoreanu. 2015. A pilot study exploring augmented reality to increase motivation of Chinese college students learning English. Paper presented at the ASEE Computers in Education Journal, Indianapolis, Indiana, 15-18 June.
Liu, Tsung-Yu. 2009. “A context-aware ubiquitous learning environment for language listening and speaking.” *Journal of Computer Assisted Learning* 25 (6):515-27. DOI: https://doi.org/10.1111/j.1365-2729.2009.00329.x.
Mahadzir, Nor Nashirah Nor, and Li Funn Phung. 2013. “The use of augmented reality pop-up book to increase motivation in English language learning for national primary school.” *Journal of Research Method in Education* 1 (1):26-38. DOI: http://doi.org/10.9790/7388-0112638.
Mühlhäusler, Peter, and Richard Damania. 2004. “Economic costs and benefits of Australian indigenous languages: Discussion paper.” In, edited by Australian Government Aboriginal and Torres Strait Islander Services. Canberra.
Mulfari, Davide, Antonino Longo Minnolo, and Antonio Puliafito. 2017. “Building TensorFlow Applications in Smart City Scenarios.” In *2017 IEEE International Conference on Smart Computing (SMARTCOMP)*, 1-5. Hong Kong, China: IEEE.
Pritchard, Alan. 2017. *Ways of learning: Learning theories for the classroom*: Routledge.
Purdie, Nola, Tracey Frigo, Clare Ozolins, Geoff Noblett, Nick Thieberger, and Janet Sharp. 2008. “Indigenous languages programmes in Australian schools-A way forward.” In, edited by Kevin Lowe. ACT, Australia.
Rahman, Mohd Hishamuddin Abdul, Noraffandy Yahaya, and Noor Dayana Abd Halim. 2014. Virtual world for collaborative learning: A review. Paper presented at the 2014 international conference on teaching and learning in computing and engineering, Kuching, Malaysia, 11-13 April.
Rose, R Faustina Jeya, and G Bhuvaneswari. 2016. Word recognition incorporating augmented reality for linguistic e-conversion. Paper presented at the 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, 3-5 March.
Russakovsky, Olga, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, et al. 2015. “ImageNet Large Scale Visual Recognition Challenge.” *International Journal of Computer Vision* 115 (3):211-52. DOI: http://doi.org/10.1007/s11263-015-0816-y.
Santos, Marc Ericson C, Takafumi Taketomi, Goshiro Yamamoto, Ma Mercedes T Rodrigo, Christian Sandor, and Hirokazu Kato. 2016. “Augmented reality as multimedia: the case for situated vocabulary learning.” *Research Practice in Technology Enhanced Learning* 11 (1):4. DOI: https://doi.org/10.1186/s41039-016-0028-2.
Simonyan, Karen, and Andrew Zisserman. 2014. “Very deep convolutional networks for large-scale image recognition.” *arXiv preprint arXiv:1409.1556*.

Solak, Ekrem, and Recep Cakir. 2015. “Exploring the Effect of Materials Designed with Augmented Reality on Language Learners’ Vocabulary Learning.” *Journal of Educators Online* 12 (2):50-72.

Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition, Boston, MA, USA, 14 Nov.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. Paper presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27-30 June.

Vishnu, Abhinav, Charles Siegel, and Jeffrey Daily. 2016. “Distributed TensorFlow with MPI.” *arXiv preprint* 2 (arXiv:1603.02339).

Wäldchen, Jana, Michael Rzanny, Marco Seeland, and Patrick Mäder. 2018. "Automated plant species identification—Trends and future directions." *PLOS Computational Biology* 14 (4):e1005993. DOI: http://doi.org/10.1371/journal.pcbi.1005993.

Wei, Wei, and Yash Katariya. 2020. “Recognize Flowers with TensorFlow Lite on Android.” Codelabs, Accessed 27 Oct 2020. https://codelabs.developers.google.com/codelabs/recognize-flowers-with-tensorflow-on-android/#0.