RESEARCH FRONT

CSIRO PUBLISHING

Journal of Southern Hemisphere Earth Systems Science, 2020, **70**, 143–150
https://doi.org/10.1071/ES19046

Content enhancement with augmented reality and machine learning

Justin Freeman

Bureau of Meteorology, GPO Box 1289, Melbourne, Vic 3001, Australia.
Email: justin.freeman@bom.gov.au

Abstract. Content enhancement of real-world environments is demonstrated through the combination of machine learning methods with augmented reality displays. Advances in machine learning methods and neural network architectures have facilitated fast and accurate object and image detection, recognition and classification, as well as providing machine translation, natural language processing and neural network approaches for environmental forecasting and prediction. These methods equip computers with a means of interpreting the natural environment. Augmented reality is the embedding of computer-generated assets within the real-world environment. Here I demonstrate, through the development of four sample mobile applications, how machine learning and augmented reality may be combined to create localised, context aware and user-centric environmental information delivery channels. The sample mobile applications demonstrate augmented reality content enhancement of static real-world objects to deliver additional environmental and contextual information, language translation to facilitate accessibility of forecast information and a location aware rain event augmented reality notification application that leverages a nowcasting neural network.

Keywords: augmented reality, content enhancement, environmental information delivery channels, machine learning, mobile device computing, neural networks, user centred rain nowcasting, weather forecast situational awareness.

Received 21 March 2020, accepted 10 August 2020, published online 1 October 2020

1 Introduction

Augmented reality is the merging of real-world environments with virtual content, providing interactive experiences and scene enhancements (Azuma 1997; Azuma *et al.* 2001). Through computer vision methods, such as object detection and image recognition, the features present within the real-world environment become accessible to computer interpretation, analysis and modification. The combination of this information with augmented reality allows the creation of user centric experiences that are location aware, context sensitive and interactive.

Modern machine learning methods have made significant advances in the field of computer vision, including image classification, object detection and tracking, and scene segmentation, and these advances have equipped computers with the means to interpret the real world environment. Wider applications of machine learning have advanced the fields of natural language processing, machine translation and understanding. Further, hardware architecture developments have provided a platform for the deployment of machine learning models directly onto consumer level mobile devices, with no degradation in performance.

The following work combines machine learning with augmented reality to explore the ways in which the real-world environment can be enhanced with digital assets to create new channels of environmental information delivery that is user centric, location and scene aware, and interactive. Four mobile augmented reality applications were developed to explore and demonstrate these elements. Within the applications, the delivery of environmental information through augmented reality overlays is demonstrated through four conceptualised applications. Each application showcases an approach to delivering user focussed environmental information. Machine learning models provide the mechanisms for scene awareness and the creation of user focussed environmental information.

Through this approach, the applications demonstrate the enhancement of real-world static content with additional digital information, the augmented reality enhancement of real-world static assets with animation and video streams, and the generation and delivery of user-centric, spatially localised and accurate environmental forecasts delivered through augmented reality channels.

2 Method

The augmented reality applications were built using ARKit (https://developer.apple.com/augmented-reality/). ARKit provides a framework for real-world and virtual content tracking on mobile devices. ARKit combines mobile device features such as motion tracking, camera scene capture and scene processing as well as augmented reality content delivery. Within ARKit, real-world tracking is accomplished using a technique called visual-inertial odometry.

Visual-inertial odometry combines motion sensor data with computer vision analysis of camera imagery to determine and track a device’s position and orientation in the real-world space

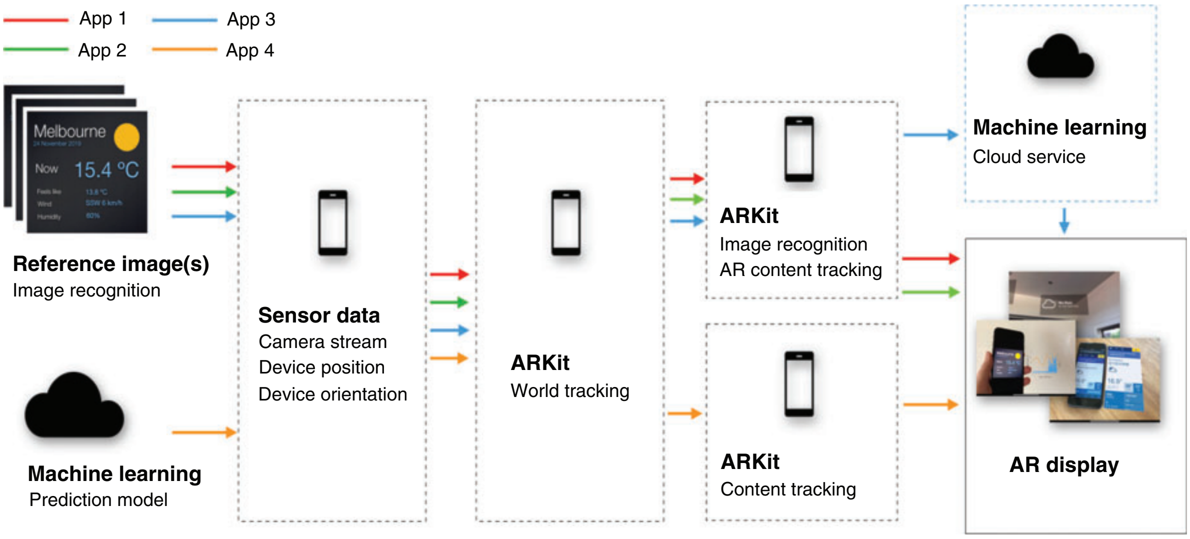

Fig. 1. Architecture diagram for the four demonstration applications. The processing chain utilised by each application is shown using the associated colour coded arrow.

(Li and Mourikis 2013). Computer vision analysis of the real time camera imagery stream enables the registration of scene features and the construction of a geometric description of the surrounding environment. Within this digital representation, augmented reality content can be registered and integrated with the camera imagery stream, creating the perception of computer generated components being seamlessly embedded with the real-world environment.

The architecture of the four sample applications is given in Fig. 1. These applications leverage several technologies to enable context aware delivery of environmental information through augmented reality channels. Three of the four demonstration applications rely on the detection, recognition and registration of image assets within the real-time camera stream. During the process of tracking, the detection of any of these reference images results in ARKit establishing a set of anchor points describing the location and geometry of the image within the real-world. Further processing by the application allows the creation and display of associated augmented reality assets, that are aligned with the real-world anchor positions. In the fourth application, the likelihood of rain is predicted using a machine learning approach and provides the basis for augmented reality content that is generated relative to the mobile device's location and orientation in the real-world. More detailed methods are provided in the following sections.

2.1 *App 1: image recognition augmented with digital content* – image2info

This application enhances real-world static content with additional digital information. This digital content is presented as an augmented reality overlay that is integrated within the real-world environment. The application detects the presence of a reference image within the users environment via the real-time device camera stream (https://developer.apple.com/videos/play/wwdc2018/610/, https://developer.apple.com/videos/play/wwdc2019/228/). Successful recognition triggers the delivery of the augmented reality content as described by the red pathway in Fig. 1.

Here, the known feature is a reference image asset that contains a weather forecast infographic, as shown in Fig. 2. Detection of the reference image within the real-time camera stream results in ARKit generating a set of reference coordinates, or anchor points, that describe the real-word coordinates of the recognised image. These anchor points are tracked and computer generated virtual content is displayed relative to the real-world position of the recognised image.

In this example, the weather forecast infographic is augmented with additional related information that displays the temperature and rainfall predictions for the next 24 hours, as shown in the augmented reality overlay in Fig. 2. Device rotations and translations result in corresponding transformations of the augmented reality asset relative to the detected asset's location, giving the appearance that the virtual content is embedded within the real world.

This application demonstrates an augmented reality pathway for the delivery of content enhancement. Through this method, user-centric information services may be delivered. Extensions to the application include the delivery of alerts and warnings, which can be dynamically generated, delivered and inserted within the user's augmented reality view. Such an approach will eliminate the need for users to manually search for this information on a website, or piece together information from multiple sources, and enables an information delivery pathway for situational awareness that is integrated with the user's current environment.

2.2 *App 2: image recognition augmented with digital content* – image2video

Within an augmented reality context, real-world static assets can be enhanced to enable the delivery of dynamical information. To demonstrate this concept, six reference images were embedded within the application. Each of these images, shown in Fig. 3, represent the first frame of an associated animation. The data visualisations were generated using data from the seasonal forecasting system (Alves *et al.* 2003), showing the sea surface anomaly and rainfall over land areas from summer 2010 to

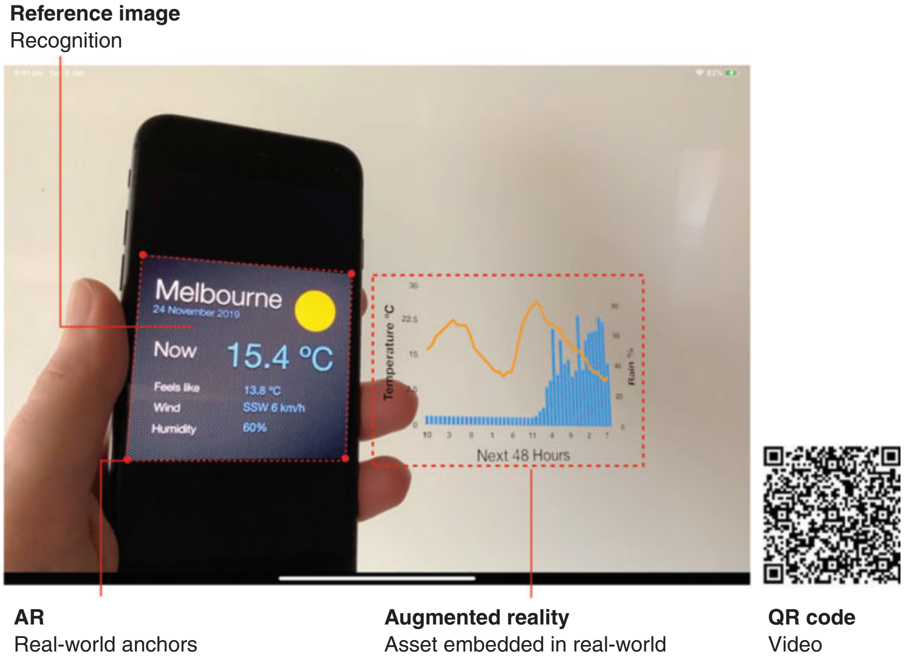

Fig. 2. A static weather forecast image is recognised by the application, and additional, related information is displayed within the augmented reality overlay, shown to left of the mobile device. The augmented reality overlay displays the temperature and rainfall forecast for the next 48 hours. Scanning the QR code will link to the demonstration video. This is also available via https://s3-ap-southeast-2.amazonaws.com/machinelogic.info/AR/img2info_480.mp4.

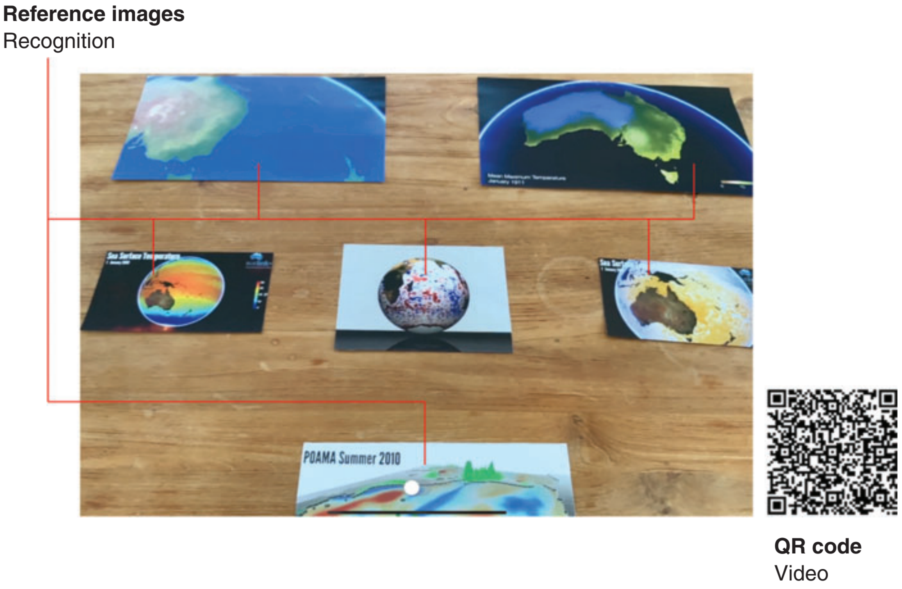

Fig. 3. Example showing the real-world static reference images that are recognised by the imageg2video application. Scanning the QR code will link to the demonstration video. Video is available via https://s3-ap-southeast-2.amazonaws.com/machinelogic.info/AR/ARimage2vid_480.mp4.

summer 2012, the Bluelink ocean forecasting system (Schiller *et al.* 2019) showing the sea surface temperature, sea surface velocity magnitude and the sea surface temperature anomaly for the period January 2006–June 2009, an oceanic particle advection model over the East Australia Current showing the change in position of 100 passive tracers over several days of advection due to sea surface currents, and the mean monthly maximum temperature for Australia from 2011 to 2013 (Jones *et al.* 2009).

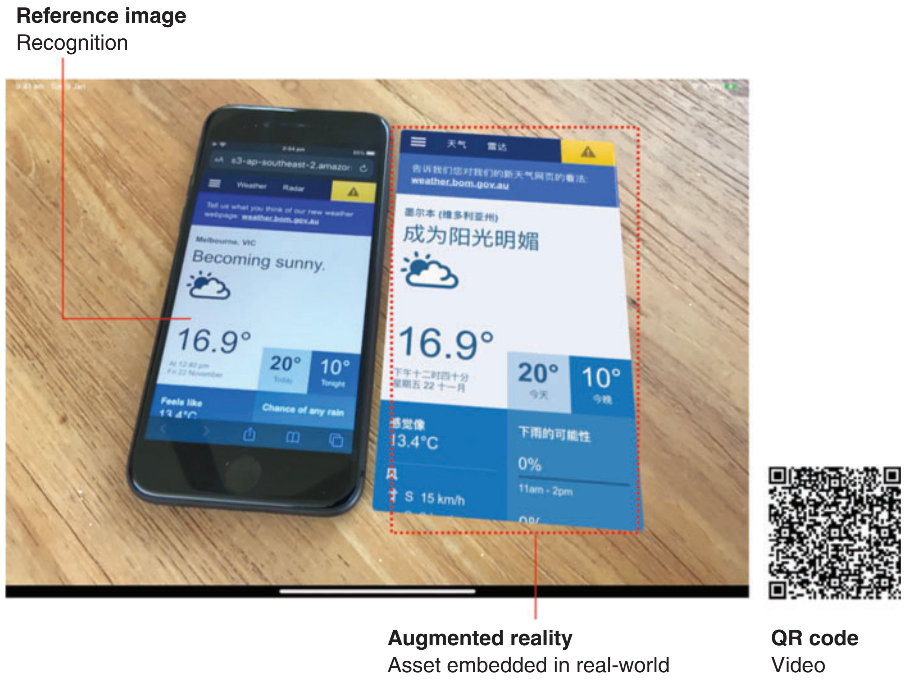

Fig. 4. Language to language translation showing the static reference image displayed on a mobile device and associated augmented reality display. In this example, the original English language webpage is translated to simplified Chinese. Scanning the QR code will link to the demonstration video. This is also available via https://s3-ap-southeast-2.amazonaws.com/machinelogic.info/AR/AR_webTranslate_480.mp4.

Within the application, the detection of any of the six reference images within the real-world environment triggers an augmented reality overlay on top of the location of the static asset, as shown in the application architecture diagram in Fig. 1. Within this augmented reality tracked overlay, the animation asset that is linked with the real-world reference image is dynamically embedded and played. As in the *image2info* application, device translations and rotations are tracked relative to the real world environment to ensure that the augmented reality content remains aligned with the location and orientation of the physical asset. Each augmented asset is independently tracked, and the animated overlay is tightly coupled to the real world asset. Although this application demonstrated the independent playback of each video asset, the application may be configured to deliver simultaneous playback and playback tracking across all video assets.

When delivering environmental forecast information, the temporal evolution of the forecast provides additional details about how the forecast will evolve over time. This level of detail is challenging to effectively communicate with static images. Common approaches employ discrete interval timelines such as textual descriptions that describe the future expected state, or a series of two-dimensional images that represent the temporal evolution of a forecast. Examples include weather forecasts describing the predicted conditions over a future time period, or the use of graphics and data visualisations showing the spatial distribution of environmental predictions. These representations are commonly employed in print media and electronic displays.

2.3 App 3: image recognition and machine translation – image2translate

An extension to the *image2information* and *image2video* samples, extends the concepts demonstrated there by now including a remote machine learning model to perform real time language translation (Fig. 4). As in the previous samples, a static reference image is embedded in the application, and detection of this static asset within the real-world environment triggers the workflow represented by the blue pathway in Fig. 1.

The static reference image in this case is a website which contains a seven day weather forecast for Melbourne, Australia. Post detection of the static asset, a cloud hosted process extracts the textual forecast elements form the web page, and for each element a language translation neural network is employed to translate the original English text to simplified Chinese. Machine translation was provided by a cloud hosted language-to-language translation deep neural network (https://aws.amazon.com/translate/). This service provides multiple language translation models that can be leveraged to expand the accessibility of environmental forecast information.

The translated elements are then used to generate a new webpage and this content is delivered back to the application. Within the application, a virtual overlay region is created and offset relative to the augmented reality tracked reference image anchors. A web view is created within this region and the translated web page is displayed. As in the previous applications, the real world position of the augmented reality content is tracked relative to the generated anchor points and the virtual content responds to view translations and rotations.

This sample application introduces an additional user interaction pathway in that the augmented reality display containing the translated webpage responds to user interactions. The content within this view behaves as a web browser, and responds to familiar user interactions such as drag to scroll as well as touch events to follow links. This was achieved through *hit testing*, where touches on the mobile device screen are translated into the augmented reality tracked world coordinate system (https://developer.apple.com/documentation/arkit/world_tracking/raycasting_and_hit-testing). Intersection testing is then conducted to determine if the intent of the touch event was to interact with the augmented reality content.

2.4 App 4: location aware augmented reality environmental overlays

Within this sample application, the delivery of accurate forecast information that is location and user aware is enabled through the combination of mobile device sensor data, a machine learning environmental prediction system and augmented reality overlays. The workflow of this application differs from the previous examples, in that a neural network model is developed to provide short term forecast predictions of rainfall locations and intensity. The application architecture is given in Fig. 1 by the orange workflow.

The machine learning model ingests radar reflectivity data that has been processed into rain rate estimates, and produces a 1 hour future forecast of how this data will evolve. The neural network was implemented using a Generative Adversarial Network (GAN) architecture (Goodfellow *et al.* 2014). Training off the GAN was performed using three months off radar observations for the Melbourne region. The associated generator and discriminator models within the GAN architecture are multi-scale fully convolutional neural networks. In this approach, the discriminator model aims to discern whether inputs to the network are members of the dataset as opposed to an instance that was output from the generator network. Each network is simultaneously trained such that the generator model learns to produce radar frames that are difficult for the discriminator model to classify, whilst the discriminator model learns to discriminate radar frames generated by the generator model.

Training of the GAN model followed the method given in Mathieu *et al.* (2016). The training and testing data contains the rainfall intensity, derived from radar reflectance observations and includes examples with and without rain presence. The data was split into 21 865 training samples and 2000 test samples. The dataset spanned observations from 19 August 2008 to 24 November 2018, with a temporal resolution of 6 minutes. The GAN model was trained for 600 000 steps. Following the GAN implementation of Mathieu *et al.* (2016), model assessment was made using a ‘sharpness’ measure which is based on the difference of gradients between the true frame and the predicted frame. Over the final 100 000 steps if the model training, the sharpness metric ranged between 11.4 and 16.5, suggesting that there is scope to improve the generator network performance.

The trained generator model was employed to produce rainfall location and intensity forecasts out to 1 hour from the last radar observation. The input for each prediction, generated by the network, was the previous four frames. For the first predicted frame, at $t = t_{b+1}$, the input was four observations located at t_{b-3}, t_{b-2}, t_{b-1} and t_b. Subsequent predictions at $t > t_{b+1}$ combine neural network generated outputs into the input sequence. Comparison between the neural network model and the ground truth radar observations, out to 30 min from the last observation, is shown in Fig. 5.

The neural network generated rain location and intensity predictions are incorporated into the sample application. The direction and intensity of rain, if present, is determined relative to the user’s current location, and the user is presented with rain presence, intensity and predicted time of arrival at their location as a series of augmented reality embedded information overlays. Mobile device sensors provide location services as well as accurate information describing the orientation of the device within the real world. These features were then combined with the GAN forecasts to deliver augmented reality assets that contain environmental information that is dynamically generated in response to the location of the device and the direction in which the device is oriented within the real-world.

Within these augmented reality overlays, weather information is displayed with the intent of informing the user about conditions which are currently active at locations near them, to provide information about weather conditions which are developing and to deliver information about changes in conditions that will impact the user in the near future at their current location. An example of the application showing the augmented reality overlays and samples of the generated forecast information is given in Fig. 6. In this example, the mobile device is facing in a westerly direction, and the augmented reality content alerts the user that there is currently light rain 3 km away from their current location and moderate rain 20 km away. For these cases, the GAN model is predicting that these conditions will intersect with the user’s location in 5 and 20 min respectively.

This demonstration application may be extended to deliver environmental alerts that are relevant to the user’s current location, such as severe weather warnings, changes in conditions which may impact the user as well as general forecast alerts and information (Fig. 6).

3 Discussion

The embedding of environmental information within augmented reality displays enables enhanced content delivery that leverages the local, real-world surroundings and preferences of the user. The sample applications demonstrate four approaches to enhancing content with augmented reality overlays and embeddings. Continued development of these concepts coupled with the ongoing developments in user localised environmental forecasting and information services will provide a feature rich and user-centric platform from which enhanced content delivery services can be developed. The conveniences of augmented reality approaches to information delivery and display have the potential to radically alter the ways in which environmental information are delivered and presented to data consumers.

The sample applications were created to demonstrate information delivery streams that are tightly integrated with the real-world environment of the user. Through augmented reality, localised and personalised information is delivered in an efficient and intuitive manner, enhancing the delivery of contextual,

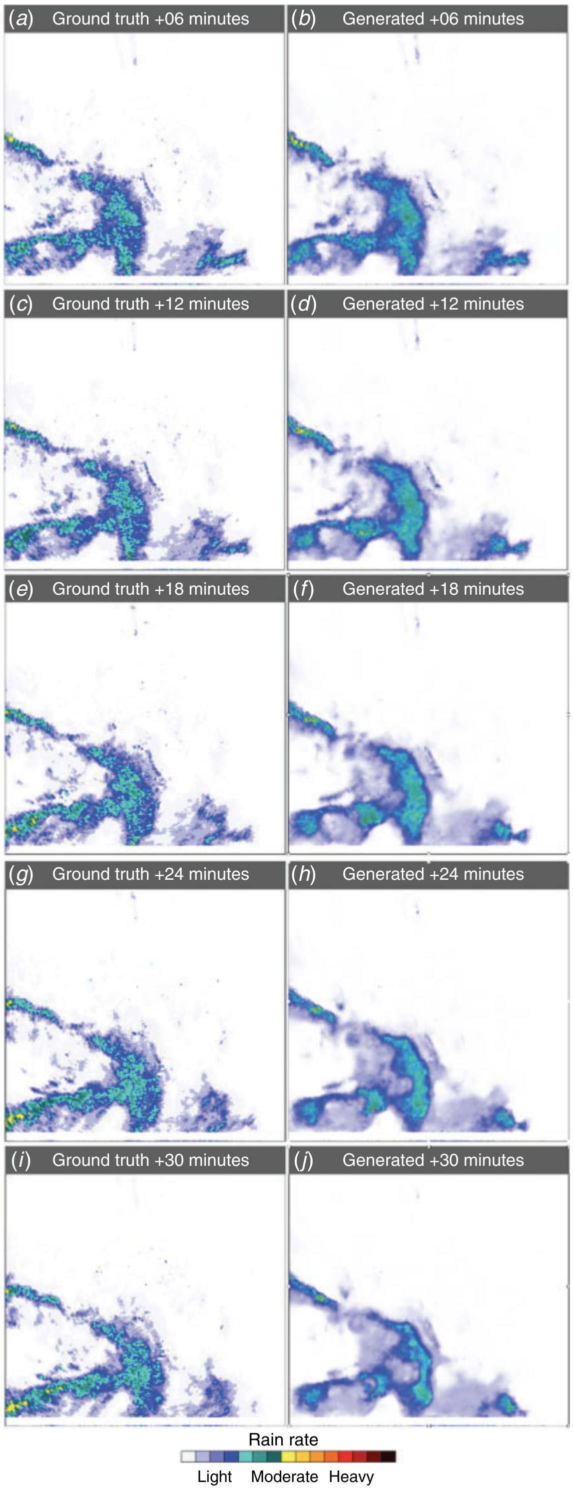
(a) Ground truth +06 minutes
(b) Generated +06 minutes
(c) Ground truth +12 minutes
(d) Generated +12 minutes
(e) Ground truth +18 minutes
(f) Generated +18 minutes
(g) Ground truth +24 minutes
(h) Generated +24 minutes
(i) Ground truth +30 minutes
(j) Generated +30 minutes
Rain rate
Light
Moderate
Heavy

Fig. 5. Generative neural network rainfall nowcast. Images on left are observations, images on right are neural network predictions. This series of snapshots shows rainfall location and intensity estimates out to +30 minutes.

user-centric information. Augmented reality, coupled with machine learning enables systems to understand, interpret and respond appropriately to the real-world environment.

Although the applications developed here perform well on the example tasks, the performance of these models will degrade as more features are inserted into the detection and recognition pipeline. The *image2info*, *image2video* and *image2translate* applications rely on the recognition of a small number of reference images to trigger the augmented reality workflows. These images are unique and no incorrect or failed detections were encountered during development and testing. However, as the number of reference images increases or the diversity amongst the reference images decreases, recognition performance will degrade. This can be addressed through the direct deployment of customised, fine-tuned and bespoke machine learning models within the mobile application.

To demonstrate this, the pre-trained Inception v3 network (Szegedy 2016) was deployed within a mobile application and image recognition performance testing was undertaken. The ImageNet (Russakovsky *et al.* 2015) pre-trained Inception v3 model was converted to *CoreML* (https://developer.apple.com/documentation/coreml) format, which is a mobile ready machine learning model framework, and implemented within a sample application. Frames from the real-time camera stream of the mobile device were fed into the Inception v3 model, which performed a classification operation on the image contents. From these classifications, the reference image can be identified and the augmented reality workflow of the application may then proceed.

One challenge with this approach is the large memory footprint of deep neural network models, such as Inception v3. To reduce the application deployment size, the precision of the weights was systematically reduced from 32 bit floating point precision to 4 bit floating point precision using a linear quantisation method. Linear quantisation discretises the network weights into the range $[a,b]$ where $a = \min(w)$ and $b = \max(w)$ and w is the full precision weight values of the trained network (https://apple.github.io/coremltools/generated/coremltools.models.neural_network.quantization_utils.html). The full precision weights consume 82 MB of memory, and through linear quantisation, this was reduced to 41 MB for 16 bit precision, 21 MB for 8 bit precision and 10 MB for 4 bit precision.

To investigate the performance of each of these precision representations of the Inception v3 weights, the output from the final Inception module was dimensionally reduced to a vector of length 2048 by applying a global average pooling operation and the L^2 norm of this vector was calculated, where $L^2 = \|x\| = \sqrt{x_1^2 + \cdots + x_n^2}$. The resulting model size and L^2 norm is shown in Fig. 7. Quantisation of the model weights to 8 bit precision results in a model that is 3.9 times smaller in size and performs at a comparable accuracy to the full precision model. Further quantisation of the precision below 8 bit results in significant model performance degradation. Other mobile optimised machine learning models are available such as MobileNet (Howard *et al.* 2017) or YOLO (Redmon and Farhadi 2017), which exhibit fast and accurate performance on mobile devices. The incorporation of the machine learning model within the application provides several advantages including faster model response times and delivers a strong user privacy environment. Sensitive user-centric information, such as location or details about their local environment, such as camera image information, are neither stored nor transmitted over a network to a remote server.

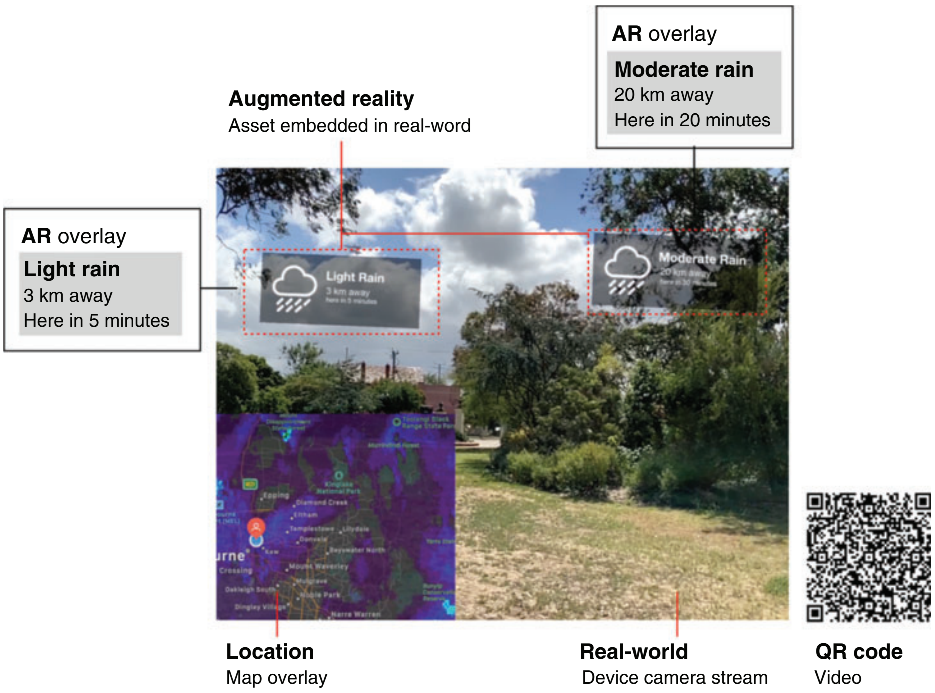

Fig. 6. Example augmented reality overlaying displaying rain arrival information at the user's location. Scanning the QR code will link to the demonstration video. Video is also available via https://s3-ap-southeast-2.amazonaws.com/machinelogic.info/AR/RainAR_ipad_trim_480.mp4.

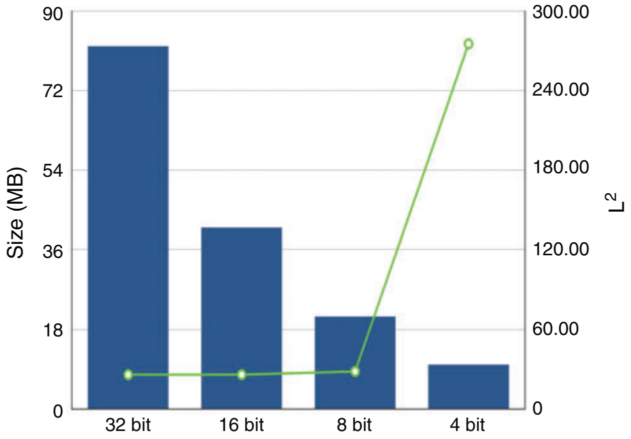

Fig. 7. Linear quantisation of the Inception v3 pre-trained weights. The L^2 norm (green line) was calculated over the final pooling layer prior to the dense network, where $L^2 = \|x\| = \sqrt{x_1^2 + \cdots + x_n^2}$.

Current mobile device warning and alerting services take advantage of many of the same hardware features as leveraged by the four demonstration applications. A common user alerting mechanism is the pushing of notifications to the user's mobile device. Whilst push-notifications provide user-centric information channels, the augmented reality demonstration applications here extend this notion by taking into consideration the current environment of the user, such as the direction in which the user's device is facing, as well as information about what objects and features are within view of the user.

The demonstration applications show the potential applications of augmented reality and machine learning in delivering environmental information in a seamless, interactive and user focussed manner. As augmented reality hardware and services develop, limitations, such as the need for user initiation of the application on a mobile device and interactions with the augmented reality displays through screens will improve.

Conflicts of interest statement

The author declares that there are no conflicts of interest.

Acknowledgements

This research did not receive any specific funding.

References

Alves, O., Wang, G., Zhong, A., Smith, N., Tseitkin, F., Warren, G., Schiller, A., Godfrey, S., and Meyers, G. (2003). POAMA: Bureau of Meteorology operational coupled model seasonal forecast system. In 'Science for drought. Proceedings of the National Drought Forum', Brisbane, April 2003. (Eds R. Stone and I. Partridge) pp. 49–56. (Queensland Department of Primary Industries: Brisbane)

Azuma, R. T. (1997). A survey of augmented reality. *Presence* **6**, 355–385. doi:10.1162/PRES.1997.6.4.355

Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., and MacIntyre, B. (2001). Recent advances in augmented reality. *Comput. Graph.* **25**, 1–15. doi:10.1109/38.963459

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. (2014). Generative adversarial networks. In 'Proceedings of the 27th International Conference on Neural Information Processing Systems, Vol. 2', pp. 2672–2680. Available at http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: efficient convolutional neural networks for mobile vision applications. Available at https://arxiv.org/abs/1704.04861

Jones, D., Wang, W., and Fawcett, R. (2009). High-quality spatial climate data-sets for Australia. *Aust. Meteorol. Ocean* **58**(4), 233–248. doi:10.22499/2.5804.003

Li, M., and Mourikis, A. I. (2013). High-precision, consistent EKF-based visual-inertial odometry. *Int. J. Robot. Res.* **32**(6), 690–711. doi:10.1177/0278364913481251

Mathieu, M., Couprie, C., and LeCun, Y. (2016). Deep multi-scale video prediction beyond mean square error. ICLR 2016. Available at https://arxiv.org/abs/1511.05440

Redmon, J., and Farhadi, A. (2017). YOLO9000: Better, Faster, Stronger. In '2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)', Honolulu, HI. pp. 6517–6525. doi:10.1109/CVPR.2017.690.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., and Bernstein, M. (2015). Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**, 1–42. doi:10.1007/S11263-015-0816-Y

Schiller, A., Brassington, G., Oke, P., Cahill, M., Divakaran, P., Entel, M., Freeman, J., Griffin, D., Herzfeld, M., Hoeke, R., Huang, X., Jones, E., King, E., Parker, B., Pitman, T., Rosebrock, U., Sweeney, J., Taylor, A., Thatcher, M., and Zhong, A. (2019). Bluelink ocean forecasting Australia: 15 years of operational ocean service delivery with societal, economic and environmental benefits. *J. Oper. Oceanogr.* **13**, 1–18. doi:10.1080/1755876X.2019.1685834

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In '2016 IEEE Conference on Computer Vision and Pattern Recognition', Las Vegas, NV, USA. pp. 2818–2826. doi:10.1109/CVPR.2016.308