Augmented Reality for Network Visualization

(Or: A Cool New Tool to Find Hidden Stuff!)

A Technical Paper prepared for SCTE•ISBE by

Joshua Seiden
Vice President
Comcast Innovation Labs
4100 E. Dry Creek Road, Littleton, CO
303 486-3619
Joshua_Seiden@comcast.com

Nishesh Shukla
Software Development Engineer
Comcast Innovation Labs
4100 E. Dry Creek Road, Littleton, CO
303 263-8121
Nishesh_Shukla@comcast.com

Table of Contents

List of Figures

1. Introduction

One of the higher-ranking intangibles on any field technician's wish list is the desire to spend more time fixing things, and less time finding them. In an industry with physical network assets measured in the hundreds of thousands of miles, in all imaginable terrains and environments, locating physical elements can be as challenging as their repair or replacement.

For many years, network technicians have used computer-aided design (CAD) maps to help them locate underground cables, aerial spans, taps, amplifiers, nodes, and other network elements. While helpful, CAD maps are two-dimensional, static and not always updated with current coordinates for the components listed – let alone Mother Nature's contributions, in the form of nightfall, snow-covered taps, shrubbery-occluded pedestals, or spans lost in heavy tree foliage!

Simultaneously, the landscape of Augmented Reality (AR) is emerging as a beneficial tool for just such circumstances. Not, as the reader may first connote, with special glasses, or helmets, or anything of the sort, but with a tool technicians use routinely in the field: A smart phone or a tablet. After extensive and ongoing laboratory reviews of various "smart glasses" and "AR helmets," designed to help technicians to access expert help in a "see what I can see" environment, our conclusion remains that such devices need a few more design cycles – and cost reductions – before they will represent a plausible field tool.

Instead, in 2019, Comcast Labs began to investigate the use of an AR modality as a network visualization tool that technicians can access with the same field tools (e.g. iPhones and iPads) that are part of their day-to-day lives. It applies Apple's open sourced ARKit framework to the integrated camera in an iOS device (iPhone or iPad), then draws upon graph database information, derived from latitude/longitude and GPS coordinates of plant elements, to present a near-real-time visual guide, regardless of time of day (or, especially, night) – there's a pedestal within 20 feet, but it's behind something tall, like a fence – as one of many examples. It gives technicians a sort of "x-ray vision" for the network, and moves the evolution of Proactive Network Management (PNM) from 2D to 3D. The tool, which we call "VON" for "Visualize Our Network," went into the field in early 2020, as a component within the tool suite technicians use daily.

This paper will describe the VON work to date, including the evolution of network visualization, how AR works to visualize the plant, development challenges, why graph theory was an important design component, the importance of "occlusion," lessons learned, and next steps (if only in the form of a "design wish list.") Note: "Occlusion," which will be mentioned frequently in this paper, is defined here as "the blocking of a view of part or all of something, but something else."

2. A Brief History of Plant Element Visualization t

Network visualization, in a classic two-dimensional, addresses-on-paper and rolled-up-maps sense, has long been a vitally important, if passive, component of proactive network maintenance (PNM). Many "cable old-timers" recount stories of entering the industry via a summer job – "here's a clickwheel, and here's how to use it" – to measure, verify and/or document the physical topologies of the network.

The state-of-the-state is now digitized, and revolves around spatial mapping in graph form. A Graph Database is a database that not only stores the elements of data, but also stores the relationships between those elements. The data is actually stored in the graph in the same way that they are connected in the "real world." Graph, in this sense, means "visibly connected," in the sense of showing the relationships between mapped elements: The amplifier goes to the line extender, which connects to the tap, then the splitter, and so on. Graph databases are the latest in what's been a long progression, happening in parallel with other PNM initiatives, and all in the pursuit of identifying and resolving plant issues before they impact customers. Graph databases will be discussed further in Section 3.

Most (but not all!) operators long ago mapped their network topologies into formats that can be reasonably easily translated into graph models. It was a natural evolution from computer-aided design (CAD), back when the state-of-the-art for network visualization were lists of addresses, which could be assembled into problem clusters.

Geographic Information Systems (GISs) came next, which yielded "dots on a map" informed by Global Positioning Data (GPS) coordinates. GIS/GPS advanced network visualization into tree-and-branch representations, which showed subscribing households connected to the branches. Since then, Google Maps emerged as a powerful visualization tool, but even in "satellite view" it's largely ineffective in seeing anything smaller than, say, a car in the driveway. Certainly not a tap; maybe a pedestal, depending on seasonality and camera angle at the time the image was collected.

The next step in network visualization, fueled by AR, takes matters one important, three-dimensional step further. Previously, the "dots on a map" showed the connection between, say, an amplifier and a tap, using each element's GPS-based latitude (X-axis) and longitude (Y-axis.) AR brings a new dimensions – a Z-axis – which adds visual depth and the camera attitude (pitch, yaw and roll) for first-person viewing perspective. This added depth allows the technician to do something that they've not previously had the capability of doing – visualizing multiple elements of the network, overlayed on the "real world" simultaneously. Not quite Superman-grade x-ray vision, but close enough to draw the parallel!

3. Realities & Challenges of Visualizing the Physical Plant with AR

As a painfully basic observation, Augmented Reality, as a technology modality, augments actual reality with informational overlays. And, as it turns out, overlaying the "this" onto the "that" – meaning most types of ancillary information designed to augment a reality – isn't the hard part of AR. The core technologies, like ARKit, are largely open-sourced, and substantial amounts of documentation exists online. This makes the actual "overlay" part of AR implementations reasonably straightforward.

The first reality when building an AR model of the network is the need to build a graph database of the network. If each element in the graph has latitude and longitude coordinates, and we understand the relationships of each element to one another, we can represent the network in 3D space. This brings us, again, to a discussion about the necessity of graph databases when developing AR-based plant visualization techniques for technicians.

Traditional relational databases rely on tables, where the data are organized into well-defined columns and data formats. This system works well when dealing with a modest number of relationships between a few different tables. But traditional relational databases are woefully underpowered to meet the needs of visualizing all of the components present in a modern broadband network.

A graph database is one that organizes data according to the elements of a graph – with its own "nodes" and "edges" that represent the relationships between the edges. Graph databases represent network connections in real time, which closely resemble how we interpret them naturally. As such they are ideal for informing network topologies and related maps in a visual way.

That said, there is the matter of the "Z-axis," mentioned in the previous section – another reality encountered when developing an AR-based plant visualization tool. Creating an AR session necessarily includes X, Y and Z coordinates: Latitude, longitude, and in this case, depth. Depth and attitude augmentation were a challenge, in part because ARKit was in beta when we began this effort, and partly because our use case required continuous depth detection, while in motion, to direct a tech to the desired destination. We added the Z-axis by using an ARKit feature called *plane detection,* and in the case of the Visualize Our Network AR effort, vertical plane detection.

Consider the technician looking for a pedestal in an established neighborhood, which is to say "in full foliage," because it's summertime. Or, that same neighborhood, buried under a foot of snow in February. Either way, the pedestal in question is behind a fence. To handle those frequent cases where a network element of interest is occluded by something – a fence, a tree, a building – we detect and notate the occlusion by marking it blue. This tells the tech that the object of interest is behind something vertical. We determine occlusion by detecting vertical planes in front of the object of interest.

Without vertical plane detection as a means of marking occlusions, it would be very hard, if not impossible, to pinpoint a specific location, in 3D. Figure 1 shows the VON app with Occlusion turned on, to indicate a vertical plane obstruction. Hence, any objects beyond the Occlusion would not be visible, such as pedestals, cable, and other elements of our plant.

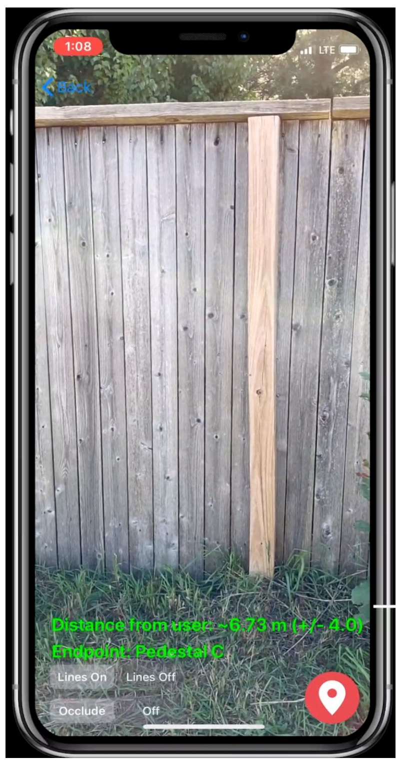

Figure 1 - The AR-based VON app indicating a vertical plane occlusion

Figure 2 (next page) shows the same vertical occlusion – the same fence – with occlusion turned off. This allows the technician to view the elements of our infrastructure, including the vertical occluded objects, but without noting them as such. This combination gives a visual depth to VON that previously didn't exist.

Figure 2 - The AR-based VON app with occlusion turned off

For underground plant, we applied a "snap-to-ground" focal point for the VON app, which, in essence, extended a "negative vertical plane" of -6 feet. As a result, by starting at a node, then going "down" the graph database to the first tap, then the second, and so on, we were able to make the assumption that if Node A is connected all the way down to those seven taps, there's likely a buried cable between them, which at best is observable as not being aerial, but rather, underground, indicated by a line/plane on the ground. Without a graph database, that job would've required a physical walk-out.

So: The hard part of AR-based plant visualization, in practice, isn't the AR. The hard part, perhaps not surprisingly, is assembling the information correctly: Latitude, longitude, depth, what kind of object, and whether it's occluded by another object. An early and persistent challenge was simple GPS inaccuracies, either from a faulty phone/app interpretation, or from incorrect GIS entries. Truth be told, we encountered at least one pedestal that was, according to its GPS coordinates, in the middle of a lake!

Another challenge: Point of view. While AR technologies continue to advance, there's still a lot of room for improvement. For instance, a marker representing a pedestal on the phone's or iPad's screen may appear to be three feet away on the screen – but actually, it's 30 feet away. The reason is the perspective: The phone is stitching a camera feed together with a virtual object. The virtual object has no explicit knowledge of what the camera is seeing. Rather, it exists in an empty world, with only an X (latitude), Y (longitude) and Z (depth) axis as a landing spot. If the perspective is off, while you're walking down an alley, looking for a plant element, it can *look* like you're close, when you're not. Resolving this took some iterations (understatement)! While tying a virtual object to a scene could be achieved with the use of model detection and logic code, to detect and identify scenes, such a plan would rapidly deplete the phone or iPad's battery.

A third challenge involved the previously mentioned matter of occlusions encountered along the way, and especially vertical occlusions – the fence, with the pedestal behind it. The VON, without scene detection of some kind of understanding of the surroundings, can't know that there's a "fence" in the way, only that there's a vertical plane occlusion. We resolved this by incorporating a way to adapt the vertical plane, to show / not show a vertical object in the way of the desired plant element. This discovery went a long way in making the user interface look realistic, and created a means for the technician to interpret depth. Without the Occlusion information, it doesn't make sense, because it lacks visual depth.

4. Findings and Next Steps:

The core technology enablers that made it possible to develop the VON application, as a PNM tool, were ARKit 3 Beta (at the time), GIS, and Graph Theory/Database. It is important to again note that the graph database aspect was crucial to VON as it described the relationship between each element of our infrastructure and how it all comes together. Graph theory, in combination with the GIS data that we had on our infrastructure, allowed for a graph database to exist and then be used by VON to visually represent all the different elements.

ARKit 3 was in a Beta state from Apple at the time we embarked on the VON tool. Because of the new-ness of the release, it allowed VON to tap into the potential for occlusion, and how it can beneficially alter the perspective of a user. Specifically, in terms of marking and presenting the visual depth of the geographical virtual objects, portrayed ontop of a digital camera feed. This is also known as "Markerless Augmented Reality," defined as AR that doesn't need prior knowledge of a user's environment in order to overlay 3D content into a scene and tie it to a fixed point in space. After the introduction of ARKit 3 in mid-2019, creating AR Sessions with virtual objects based on geographical data acquired from graph databases was straightforward. The challenge (still) is the available plant data and the GPS inaccuracies of sensors within our handheld devices.

Simply put, accuracy matters to visualing the physical plant with AR. It matters in particular as to how we place the virtual element, onto the digital (camera image) feed. Both the stored geographical data about our infrastructure elements, and the GPS circuitry present in handheld devices, need a reasonable degree of accuracy to best inform the VON effort. Currently, inaccuries can create a visual drift from the intended object of three to 12 feet. This is why a favored new feature for VON is the ability to re-tag a network element with correct location coordinates, quickly and easily, and as technicians use the app to locate objects. Better data = better accuracy = finding hidden elements faster.

To more thoroughly test the VON application, in real-world circumstances (no pun intended), it was "easter egged" into an existing suite of PNM apps used by Comcast's field technicians. In that sense, it's "out now," although truth be told, only a slight percentage of the field has yet to find it and put it to work. Also, it requires an iOS device that is optimized for ARKit 3 and above (iPad mini, iPhone 7 and above), which not all technicians have, as a function of routine device upgrades.

As with any new Augmented Reality application, the initial reactions run the gamut from "how cool" the application is, to an overall and genuine excitement about it. The excitement about it will drive more technicians to use the application, and provide feedback to improve it. As we gather this feedback, and continue to experiment with the "art of the possible," a few items tend to rise quickly to the list of desired feature additions for VON.

Most involve giving technicians more information about a task, by providing historical context. For instance, the ability to "hover" over a home, then "see" as an AR overlay its known technological history – CPE spectrum measurements, or full spectrum analyses from gateways, as two of several potential examples. From a plant element perspective, the ability to augment, say, an amplifier, with a visual overlay of its latest sweep trace measurement, would be a time-saving and useful troubleshooting tool for the field.

Another and more practical/tactical consideration is an easier, on-the-spot/in-the-moment way to correct incorrect database entries, as mentioned above, by visually "attaching" to a plant element, and thereby correcting bad data.

5. Conclusion

This paper described an ongoing effort within Comcast Labs, and active in the field with technicians, to use Augmented Reality for finding specific plant elements – and especially those that aren't plainly obvious, either because it's dark out, or because they're covered up by something else. Called "Visualize Our Network," or VON, it uses AR to add a 3D element to Proactive Network Management. It very specifically uses an iPad for the visual element, and not "AR glasses," which can get lost, or be unwieldy for people who already wear corrective eyewear.

Equipping technicians with this kind of 3D "find it" tool is both approachable and very well received, both for the on-the-job assistance, and the "cool factor." Using open-source elements like ARKit, and assuming an accurate plant database exists, with latitude and longitude coordinates for key plant elements, like nodes, amplifiers, and taps, you too can trick out your PNM tools with 3D plant visualization!

Abbreviations

AR	Augmented reality
ARKit	Augmented Reality Kit
CAD	Computer Aided Design
GIS	Geographic Information Systems
GPS	Global Positioning Satellite
PNM	Proactive Network Management
VON	Visualize Our Network

