Exploring 3D Cybersecurity Visualization with the Microsoft HoloLens

Steve Beitzel[1], Josiah Dykstra[2], Paul Toliver[1], and Jason Youzwak[1(✉)]

[1] Vencore Labs, Basking Ridge, NJ, USA
{sbeitzel,ptoliver,jyouzwak}@vencorelabs.com
[2] Laboratory for Telecommunication Sciences, College Park, MD, USA
jdykstra@LTSnet.net

Abstract. We describe the novel use of the Microsoft HoloLens to assist human operators with computer network operations tasks. We created three applications to explore how the HoloLens may aid cybersecurity practitioners. First, we developed a 3D network visualizer that displays network topologies in varying levels of detail, ranging from a global perspective down to specific properties of individual nodes. The user navigates through the topology views using hand gestures while responding to simulated alarm conditions on specific nodes. Second, we developed an application that simulates a "capture the flag" exercise. Third, we developed an application to test network connectivity. We discuss the benefits, challenges, and lessons learned from developing mixed-reality applications for computer network operations. We also discuss ideas for further development in this area.

Keywords: Cyber security · Network security · Mixed Reality

1 Introduction

The goal of this work is to investigate the feasibility of using Mixed Reality (MR) devices to assist the day-to-day work of network operators. Network operators, who monitor and defend computer networks, are often required to perform several simultaneous tasks that require focused concentration, while also handling interruptions due to emergent high-priority tasks. This places high cognitive load on the operator. One of our primary research goals is to explore ways of incorporating MR devices into the network operations workflow to improve the user experience. In future research, we also plan to perform additional evaluation on how it impacts stress and cognitive load.

In previous research [1], we explored the use of Android-based Augmented Reality (AR) devices with basic capabilities and performed experiments designed to demonstrate the effect of AR devices on user cognitive load. These experiments showed that users expressed a decrease in their cognitive load when using an AR device with limited capabilities to monitor for emergent alerts. In this subsequent effort, we familiarized ourselves and experimented with the HoloLens [2], Microsoft's hardware and accompanying software for Mixed Reality. Compared to the previous AR devices, the HoloLens has more advanced features such as a stereoscopic 3D optical head-mounted

D. Nicholson (ed.), *Advances in Human Factors in Cybersecurity*,
Advances in Intelligent Systems and Computing 593, DOI 10.1007/978-3-319-60585-2_19

display, gaze tracking, spatial mapping, hand gesture navigation, advanced voice commands, and spatial sound. Since the release of the HoloLens, there has been increasing research into its use for a range of visualization applications, including molecular structures and architectural forms [3], augmented reality assisted surgery [4], and using biometric feedback to encourage focused concentration [5]. The work described in this paper is targeted at exploring the possibilities afforded by HoloLens capabilities within the context of computer network operations (CNO).

In this paper we describe the capabilities of the HoloLens and lessons learned in the development process we used to create applications for it. We built several prototype applications that explore CNO activities mapped into 3D environments, including tools for network visualization and network monitoring. We discuss the advantages and limitations of using the HoloLens, and offer ideas for future work.

2 Approach

2.1 Mixed-Reality Headset

The HoloLens is a mixed-reality head-mounted device developed by Microsoft, and marketed for a wide range of applications including gaming, design and engineering, education and training, and data visualization. In Fig. 1 we show the HoloLens device and internal components. The HoloLens contains an Intel 32-bit processor, a custom-built Microsoft Holographic Processing Unit (HPU 1.0), 2 GB RAM, 64 GB flash memory, and network connectivity via Wi-Fi 802.11ac [6]. Using projection-based smart-glasses that utilize optical waveguide technology, 2D and 3D images can be displayed on the HoloLens, overlaid on top of the user's field of view. Depth-sensing and 2D cameras enable spatial mapping and image sensing of the user's environment, allowing the HoloLens to track the user's gaze and place virtual 3D objects at known positions relative to real-world surfaces. A pair of speakers integrated into the headset enables binaural audio to simulate effects such as spatial sounds within the user's environment. Finally, an integrated noise-cancelling microphone enables control of applications via voice commands.

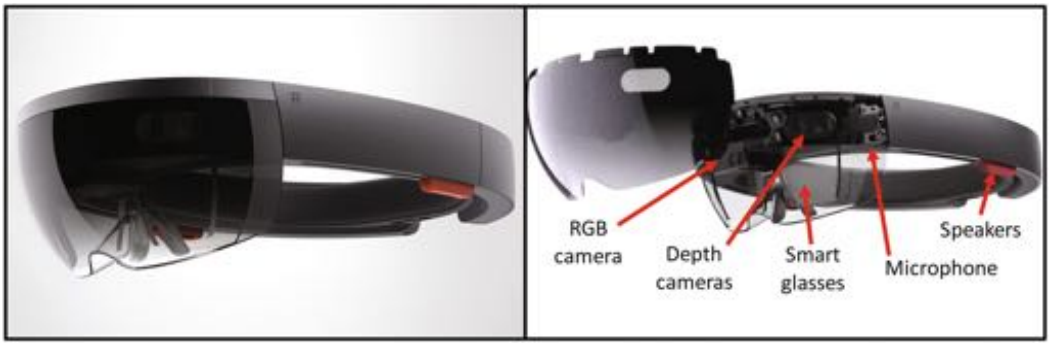

Fig. 1. HoloLens device and major components [2, 7]

2.2 Mixed-Reality Application Software

The HoloLens runs a 32-bit version of Windows Holographic 10 (currently version 14393.693 as of March, 2017) that supports Universal Windows Platform (UWP) apps. The HoloLens supports 2D apps, which are experienced as 2D projections within the user's field of view (e.g. web browser pane on wall), as well as full stereoscopic 3D apps, which fully immerse the user in a rich 3D experience. 3D apps are the focus of this work.

Holographic apps utilize Windows Holographic APIs, which provide a range of building blocks for interfacing with the HoloLens device itself including: (i) world coordinate system, (ii) tracking of user's gaze, (iii) gesture input, (iv) voice input, (v) spatial sound, and (vi) spatial mapping of the user's environment. These building blocks are completely integrated into Unity [8], which greatly simplifies 3D app development. In addition, Microsoft developed the Unity HoloToolkit [9], which provides additional components for developers including: (i) 3D cursors, (ii) display of spatial mapping, (iii) gesture-based object placement, (iv) object scaling and rotation, (v) linking of objects to particular spatial sounds, (vi) and spatial anchoring for coordinate system registration.

3 App Development Process

3.1 Development Workflow

In Fig. 2 below, we illustrate the workflow we used in developing several custom Holographic apps for the HoloLens. We utilized the Unity game engine as our core development platform. Unity manages integration and linking of the component assets used within an app project, including C# scripts, 3D objects, text data, images, audio, and others. In addition, Unity provides its own built-in components for 3D primitives

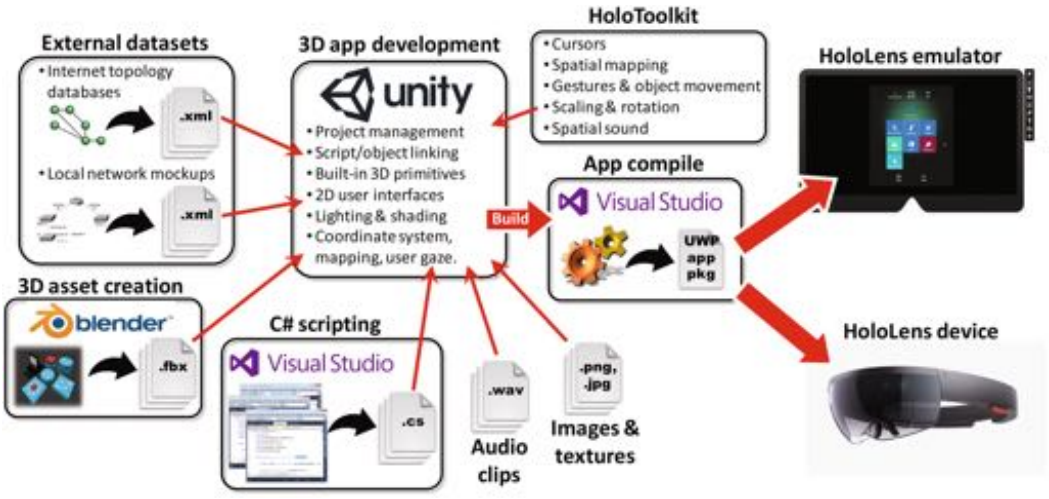

Fig. 2. Workflow we followed for developing HoloLens Apps.

(e.g. spheres, cubes, cylinders, 3D text) and libraries for building 2D user interfaces for use within the 3D world.

We also used other tools and data sources to create several external assets that we imported into our Unity projects. For example, databases describing the global and local network topologies that we display in our network visualizer 3D app are stored in XML files defined in a GraphML [10] format. We used Blender [11], an open-source 3D software tool, to create more complex 3D objects beyond simple spheres and cubes, such as iconography for network elements (routers, machines, etc.). We also developed several custom C# scripts using the Visual Studio editor [12] to interface with the HoloLens device, dynamically create 3D scenes, and enable remote network connectivity. Finally, we imported image files in several formats (e.g. PNG, JPG) for texturing raw 3D objects, and we imported audio clips to allow for specific spatial sound effects.

4 Results

We developed several applications to demonstrate alternate approaches to network security operations leveraging the HoloLens. In particular, we explored using the 3D stereoscopic display to provide a novel method of visualizing network status and the use of hand gestures for navigation. Over the next several sections, we describe applications demonstrating support for 3D network visualization, notional network security training exercises, monitoring of network captures, and testing network connectivity to the HoloLens.

4.1 3D Network Visualizer App

The 3D network visualizer prototype app displays network topologies in two levels of detail, first from a global perspective, showing all networked assets at a high-level; and second, the local area network topology surrounding a user-selected node from the global view. The app simulates intermittent random network "alerts", representing emergent problems that require attention, to draw the user's attention to specific nodes. The user can then gesture-select the node(s) in question to zoom into a more detailed view of the local topology.

The example global network topologies used by the app are based on GraphML databases available from the Internet Topology Zoo [13]. The app geographically displays these topologies on an Earth sphere in the global view. The app parses each GraphML database and dynamically constructs the corresponding 3D network graph within Unity, with spheres representing nodes and links interconnecting the nodes. In Fig. 3 below, we show an example that includes topologies based on the AT&T North America, British Telecommunications (BT) Latin America, BT Europe, and BT Asia-Pacific databases. Two different Earth images are used to illustrate daytime and nighttime texturing based on NASA images [14]. Using the HoloLens' gesture-based input capabilities, the user can set the globe in the network visualizer app to rotate automatically or to respond to real-time control by the user. The globe can also be positioned within the spatially-mapped environment and have its size scaled dynamically using similar gesture-based controls.

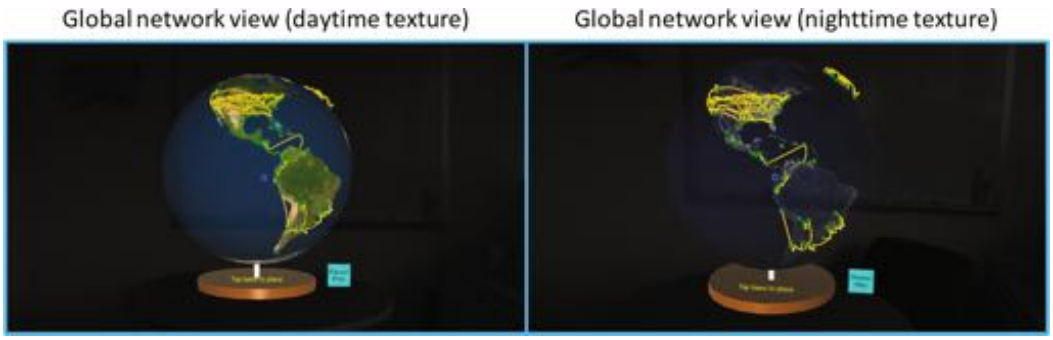

Fig. 3. Network Visualizer App in global view

To simulate alerts, the app selects random nodes from the global network topology, with random intervals between each alert. The app signifies an alert by changing the affected node's color from green to red and displaying a flag with the node's name as seen in Fig. 4. When the user gazes to the flag and gesture-selects it, the view of the visualizer dynamically changes from a global view to a local network view associated with that particular node.

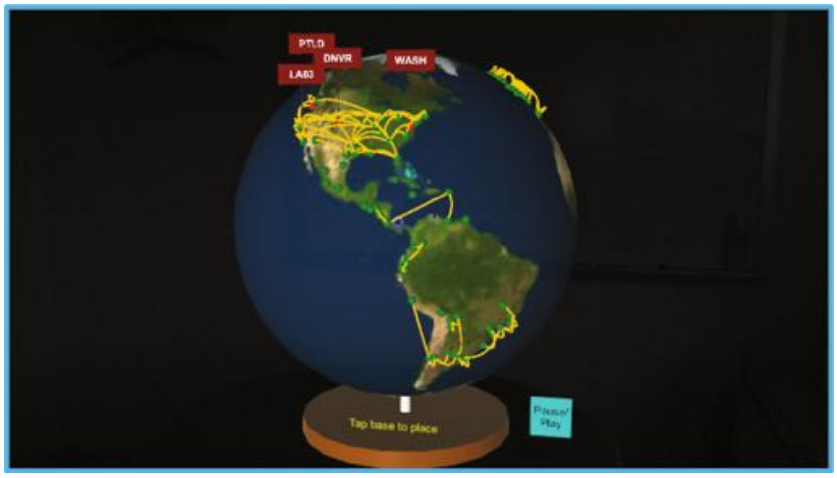

Fig. 4. 3D Network Visualizer App with simulated network alerts in global view

The local topologies displayed in the network visualizer are mockups of networks and associated network elements similar to what might be representative of a point-of-presence [15]. The app defines local networks using a format similar to the Internet Topology Zoo GraphML databases. We constructed 3D iconography for a variety of network component types, including routers, switches, and computers. Each local network topology is imported and dynamically constructed upon global node selection. In Fig. 5 below, we show three different examples of local topologies, each including various 3D network elements as well as their interconnecting links. In like fashion to the global topology, the user can rotate and scale through local network topologies through gesture-based user control.

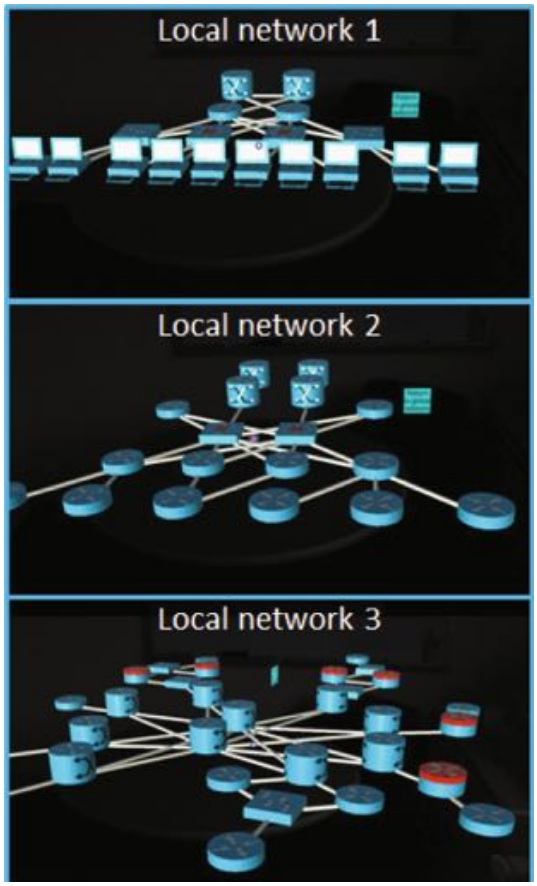

Fig. 5. 3D Network Visualizer App for different local network views.

The HoloLens, tracks the user's gaze and highlights individual links or network elements as the user views the local network topology. When the user manipulates an element with gesture-selection, the HoloLens displays mockups of different network diagnostic windows on top of individual 3D network objects. In Fig. 6, we show some of these, including a mockup command terminal window to illustrate potential HoloLens-based interaction when diagnosing problems on network switches and routers. We also created a

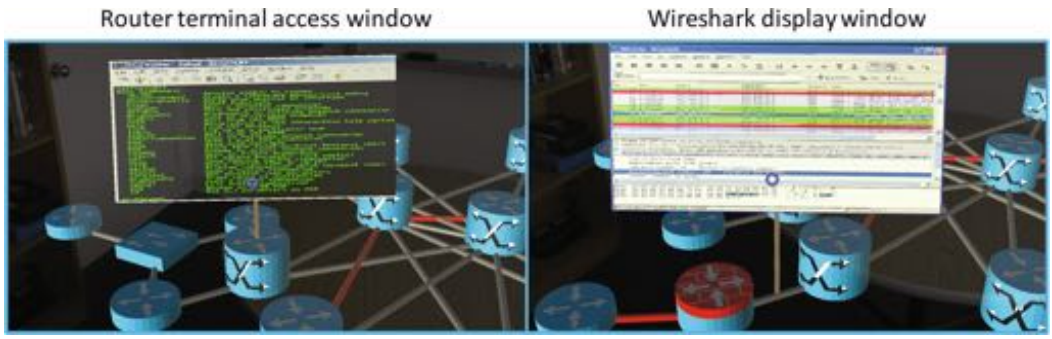

Fig. 6. 3D Network Visualizer App with mockups of gesture-selected display windows for network elements (terminal access window) and network links (Wireshark display)

mockup Wireshark display to illustrate envisioned user interactions when analyzing traffic on network links.

4.2 Capture the Flag via 3D Objects

We built a HoloLens "Capture the Flag" application to simulate a network security exercise. The application requires the user to locate a set of encrypted text files scattered randomly across a simulated file system. The contents of one file decrypt the contents of the next file in a sequence, which continues until all files have been decrypted. While performing the main task of locating and decrypting files, the user also needs to respond to simulated network alerts that occur at random intervals. This app embodies the test scenario we developed for our previous work in this area [1], adapted for use on the HoloLens.

The user navigates through items in a file browser window using HoloLens hand gestures. When the user locates a "flag" (denoted by the filename beginning with the string "flag"), the app creates a colored 3D lock representing that flag displays it in the user's world as shown in Fig. 7. The first flag, "flag 0", is initially unlocked, with the remaining flags locked. Using hand gestures, the user moves "flag 0" onto "flag 1", which unlocks "flag 1". When "flag 1" is unlocked, it yields a 3D key of matching color that unlocks "flag 2" and so on.

Fig. 7. Capture the Flag App displaying a file browser navigation window and Alert Management area

While performing the main task of searching for and unlocking colored flags, the user is interrupted periodically by visual and audible alerts. The user responds to alerts by shifting their gaze to the area of their world where the alerts are displayed. Once selected, the user dismisses the alert by using a hand gesture to apply a simulated countermeasure that clears the alert as shown in Fig. 7.

In previous research [1], user input came from a computer keyboard, a computer monitor was used as the main display, and alerts were displayed on Android-based AR glasses with limited capabilities. To respond to an alert, the user manually typed the IP address of the affected node into a control terminal. In the HoloLens app, the display is seamlessly overlaid on to the user's field of view and all input is supplied through hand

gestures. The user interface was also designed to be more intuitive: when an alert occurs, a fire is visually and audibly indicated, and the user applies a countermeasure by tapping on the network object to 'put out the fire'.

4.3 Network Feed App (from Remote System)

The Network Feed application allows the user to view a real-time network capture obtained from a remote system, conceptually similar to a typical Wireshark session. The HoloLens user can enter the network location of the remote system, connect, and view the text output in a scrolling window.

On the HoloLens, the user enters the network address and port of a host running the network capture service. Once connected, the network capture service sends a live stream to the HoloLens containing the network traffic captured on the remote system. The Network Feed App displays the capture results in real-time in a scrolling window on the HoloLens displays, as shown in Fig. 8. To narrow the focus of the capture, the user can apply some basic filters (UDP, TCP, and ICMP).

Fig. 8. Network monitoring window running on HoloLens

The network capture service on the remote host runs the tshark [16] command in the background and publishes the output to HoloLens clients connected via a network connection. HoloLens client apps also use this communication channel to command and control the program to apply various network filters.

4.4 2D Network Connectivity Tester App

Universal Windows Platform (UWP) applications can run on PC-based versions of Windows 10 and the HoloLens (as well as several other platforms). As mentioned previously, developers build UWP applications in Visual Studio without the need for Unity or DirectX libraries. UWP applications execute in the "main world" of the

HoloLens, and a user can have several UWP apps active at a time. We chose to develop a simple UWP app to explore the user experience of 2D UWP apps running alongside other 2D UWP applications.

The Network Connectivity Tester is a simple application that listens on a range of TCP ports and displays a message when an incoming connection is received, as shown in Fig. 9. This can be used for debugging a network connection, and to determine if there are any port restrictions on the HoloLens.

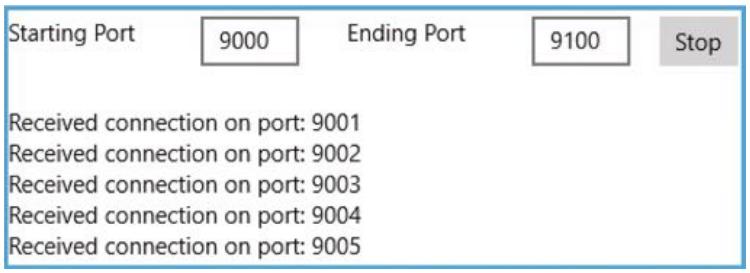

Fig. 9. Network Connectivity Tester

5 Discussion

From a human factors and ergonomics standpoint, the HoloLens headset represents a significant advance over previous augmented reality products. However, the current version does have some limitations that may impact a user's experience, depending upon the application and personal user preferences. In this section, we summarize our opinions of the key capabilities and limitations we observed based on our experiences with the HoloLens to date.

Key capabilities:

- Full-featured MR device with no physical tethering constraints.
- Smart-glasses provide good image displays for 2D and 3D stereoscopic apps.
- Multiple capabilities integrated in headset including spatial mapping, spatial sound, gesture & voice control, and wireless communications (both Wi-Fi & Bluetooth).
- High quality spatial mapping of the environment and stabilization of 3D objects enabled by depth cameras and Microsoft's custom Holographic Processing Unit (HPU).
- Extensive development support, including development tools, library toolkits, examples, tutorials, and active developer forums.
- Powerful testing and debugging tools, including the Emulator and Device Portal.
- Unity provides a framework that allows 3D stereographic apps to be created relatively quickly (compared to developing graphics routines directly in DirectX).

Potential device limitations:

- Relatively large, heavy device, particularly on the front side.
- Relatively narrow field of view (~33°–45°).
- Requires calibration of inter-pupil distance for best display.

- Constrained range for 3D objects and spatial mapping (closest: ~1 m, furthest: ~5 m).
- Gesture-based interface can be slow and limiting in some cases where a physical keyboard would be more efficient.
- No official support for creating custom gestures, although a third-party toolkit does exist [17].

6 Future Work

As part of our future work with the HoloLens, we plan to extend the 3D Network Visualizer app described in Sect. 4.1 beyond its present mockup form. Specific areas for improvements include incorporating more detailed local topology diagrams based on realistic networks. We also plan to extend the diagnostic pop-up windows (currently displayed as static image mockups) to allow for interactive display of realistic network data, such as traffic link analysis. In addition to network traffic, we plan to consider other types of network data, such as reported events and alarm conditions.

We will also be exploring how to extend network data visualization beyond the current 2D diagnostic displays to an immersive 3D world. Options here include dynamically updating the 3D network diagrams based on changing network conditions, such as link traffic volume. This could be represented, for example, by changing physical link size or including additional 3D features. Another option is to utilize other visualization techniques beyond topology diagrams, such as connectivity ring graphs or connection flow diagrams that have previously been explored for 2D network dashboards [18], and extending them to the immersive 3D environment. Finally, we plan to explore how the HoloLens' capability for sharing a common spatial reference frame between multiple users could be applied in the context of collaborative network operations tasks.

In addition to exploring technical improvements, we also plan to perform an evaluation on how the use of the HoloLens impacts stress and cognitive load in performing network operations tasks compared to working in an environment without MR devices.

7 Conclusions

We have presented new concepts for assisting computer network operations tasks through the use of the HoloLens mixed-reality device. We described a range of 3D applications built for testing different user scenarios and summarized the software development process involved in prototyping such apps. In contrast to the 2D limits of traditional computer displays, apps running on the HoloLens device enable both physical and virtual objects to co-exist within the real world environment while allowing the user to move around and interact with the mixed-reality 3D scene. Our prototype apps demonstrated how the HoloLens can augment the network operator's experience by visualizing network topology and status conditions in a 3D space, for example. Plans for future work are focused on extending these apps towards increasingly realistic network scenarios and sources of data, such as network traffic flows.

Acknowledgements. This material is based upon work supported by the U.S. Government under contract HR98230-13-D-0055. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

References

1. Beitzel, S., Dykstra, J., Huver, S., Kaplan, M., Loushine, M.: Youzwak, J: Cognitive performance impact of augmented reality for network operations tasks. In: Nicholson, D. (ed.) Advances in Intelligent Systems, pp. 139–152. Springer, Cham (2016)
2. Microsoft HoloLens. https://www.microsoft.com/microsoft-hololens/en-us
3. Hockett, P., Ingleby, T.: Augmented reality with HoloLens: experiential architectures embedded in the real world. arXiv preprint, arXiv:1610.04281 (2016)
4. Cui, N., Kharel, P., Gruev, V.: Augmented reality with Microsoft HoloLens holograms for near infrared fluorescence based image guided surgery. In: SPIE, vol. 10049. SPIE Publications (2017)
5. Ideals – Focus. https://devpost.com/software/ideals-67c1kl
6. Microsoft HoloLens Hardware Details. http://www.microsoft.com/microsoft-hololens/en-us/buy
7. HoloLens Development - Tony Labs. http://www.tonylabs.com/hololens-development/
8. Unity - Microsoft Windows – HoloLens. https://unity3d.com/partners/microsoft/hololens
9. HoloToolkit. https://github.com/Microsoft/HoloToolkit-Unity
10. The GraphML File Format. http://graphml.graphdrawing.org/
11. Blender. https://www.blender.org/
12. Visual Studio. https://www.visualstudio.com/
13. The Internet Topology Zoo. http://topology-zoo.org/
14. NASA Visible Earth. http://visibleearth.nasa.gov/
15. Service Provider IP System Test. http://www.cisco.com/systemtest/spip2b/index.htm
16. Tshark - The Wireshark Network Analyzer. https://www.wireshark.org/docs/man-pages/tshark.html
17. Gesture Recognition Toolkit. https://github.com/nickgillian/grt
18. Chen, S., Merkle, F., Schaefer, H., Guo, C., Ai, H., Yuan, X., Ertl, T.: VAST 2013 mini challenge 3: AnNetTe collaboration oriented visualization of network data. In: IEEE VIS 2013 (2013)