BE[KNOWN].iO

In-Programming Product Placement

A cursory exploration of leveraging RF technology for dynamic product identification as a means to advertise

Authored by: James S. Chambers
Version 4.0
Last Revision: 08.03.2018

Table of Contents

Executive Summary

The intent of this paper is to describe a novel advertising platform for video-media achieved through enhanced methods of organic product placement; this system will be referred to as "BeKnown.io" or simply "BeKnown". The creation of this platform is achieved through the development of a proprietary local real-time location system (RTLS) to spatially locate objects (promoted products) in three-dimensional-space and render them in 2D as a meta-overlay for movies and television. Objects are tagged with Radio Frequency IDs (RFIDs) which are coded with item attributes such as product name, physical dimensions, and a hyperlink to the product's original webpage. A camera-mounted reader-array employs triangulation or trilateration to determine the positions of these tags and receive the encoded data. Software logic uses the dimensional information received, coupled with the ranging-values computed via each reader to calculate the appropriate 2D simulated digital footprint of the product and create a clickable hotspot for each product in the frame. These frames are synced and overlaid with the matching traditional video layer captured by the camera. The output is a dynamic video file that allows viewers to click on products visible on-screen and be served product information instantly in any number of different screen configurations.

Time for an Evolution in Product Placement Methods

The methods employed for in-programming product placement (or embedded-branding as it is also referred), have been largely stagnant for decades. The practice of embedding sponsored products into scripts may be inorganic and distracting to viewers. At times products are overtly displayed in the frame for prolonged periods of time or placed blatantly to command the viewers' attention. A much-maligned example is seen in Figure 1 in which Coca-Cola-branded cups were an ever-present fixture on American Idol. At other times on-screen characters specifically state the brand so persistently or recklessly that the script loses authenticity.

Figure 1: Screenshot from American Idol which reflects the ever-present embedded branding of Coca-Cola

This method is undesirable not only because of the blatancy of the execution but also because it can only be used in a limited volume due to the intolerance viewers will develop from such overt placements. Using a 30minute sitcom as an example, one must recognize that the short nature of the content limits the number of opportunities by which advertisers can explicitly promote their products to the viewers without undermining the true content. Saturation of product placements can render a sitcom into a veritable infomercial.

The author of this paper argues a more organic and efficient technique can be achieved by removing the reliance on explicit cues. The foundation of this is a dynamic tagging approach across an unlimited number of on-screen items. The outputs of the tagging system (to be conveyed in proceeding sections) and the post-production processing create an interactive viewing experience in the form of hotspots by which consumers of the content can be exposed to a breadth of products in an unadulterated environment. The digesting of content remains traditional, that is, a person can view a movie or tv show as usual, however if his/her interest is piqued by an on-screen item, whether because a character is wearing it, performed a function with it, or merely exists in the background of a scene, the viewer can use a pointing device (mouse, touch) to engage the object to display product information. The advantage of this is two-fold; overt cues are removed, and there is no concern for product-placement saturation. The latter is particularly significant because it bestows the opportunity for countless advertisers to be known to the viewer. It should be noted that like any platform, the strength of the platform relies on the number of participants, this case is no different. Having few and sporadic tagged objects on screen undermines the viewers' capacity to engage, particularly since visual cues for hotspots will not be the default configuration. The audience should have the expectation that the majority items on-screen will be tagged.

Market for In-programming Product Placement

The market for in-programming product placement is substantial and growing according to PQ Media, a leader of global media econometrics. According the firm's 2018 industry report:

- Global branded entertainment marketing revenues rose 8.0% to $106.20 billion in 2017, fueled by the ninth consecutive year of double-digit growth in product placement and consumer content marketing revenues worldwide.

- Global product placement revenues increased 14.0% to $15.68 billion last year, with television being the largest channel at $10.50 billion and digital media rising the fastest, up 26.7%.

- Despite the number of TV placements declining, PQ Media expects product placement to grow at a similar pace throughout the 2018-22 period, as plot integrations at higher CPM levels become more common.

- Brands are aligning with rising social media stars via YouTube to use digital product placements to engage younger demos.

- Music video placements are also benefiting from YouTube and the trend toward products being placed in song lyrics, particularly in hip-hop.

- In the United States, overall branded entertainment marketing expanded 8.8% to $50.57 billion in 2017. Experiential marketing was the largest US platform, generating $35.17 billion, while product placement rose the fastest, up 13.7%.[1]

Leveraging Technology for Product Recognition

Inadequacies of Computer Vision as a Feasible Solution

It has been proposed that with the monumental advancements in object tracking, computer vision may be a viable software-based solution for product recognition within the embedded-branding ecosystem. This solution may require less hardware than a Real Time Location system by eliminating the need for tags, instead relying on complex algorithms and extensive processing power. A familiar example of such a technology is the Apple Facial Recognition system (Face ID) built for unlocking iPhone, with claims that the capability is more secure than its fingerprint predecessor. While impressive for its ability to recognize identifiable features of a specific face (even when facial hair is introduced or removed), the system is not designed for broader use cases such as random object recognition. Instead its accuracy is derived from comparing an active-user's face to the image of the original/programmed face in memory. In order to transcend the narrow constraints of face-recognition, the system would need a broader reference image-library. However, even this approach is limiting, as it's appropriate for use-cases with controlled environments and few objects; it is not well suited for more dynamic environments like a production set for a movie or television show.

Alternatively, one may argue that the self-driving vehicle industry has provided breakthroughs in object recognition, to such a degree that autonomous cars have been given federal clearance and are currently deployed across the country, successfully able to navigate complex traffic patterns and recognize and avoid objects at high -frame rates. The

[1] PRWEB. https://www.prweb.com/releases/2018/03/prweb15373577.htm

functionality of this is displayed in Figure 2; here the computer vision interpretations from a Tesla's systems are demonstrated. Each color of a feature represents a different classification, such as a roadway lane or a parked vehicle.

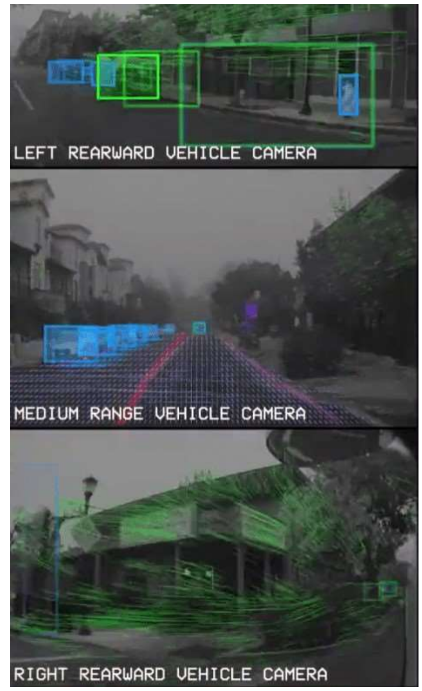


Figure 2: Example's Tesla's computer vision system which can detect lanes, road boundaries in-path objects, road lights, road signs, and other objects.

However, limitations do exist, particularly as it relates to volume and diversity of objects requiring identification (ie a self-driving system classifies objects into generic buckets, and thus a granular level of detail is not needed. For instance, a passing SUV would be classified as a car, rather than being identified as a Chevrolet, and more specifically, as a Chevrolet Tahoe). Self-driving cars are programmed to operate in a relatively narrow use-case (roadways) and thus this "object-labeling" and "scene-classification" approach works well, however when it comes to unique, non-controlled settings, the complexities of on-screen interactions and transformations that an object may experience would create unrecognizable occurrences; for instance a t-shirt, when being worn, will adopt more or less the form-factor of the person wearing it; when taken off and placed on table its form-factor is nearly unrecognizable from its prior state, any contextual references such as the face of the person wearing it or the shape are now disassociated.

There are a variety of tools that attempt to identify objects in a more specific fashion. Google, for example, has developed a feature in the company's Google Photo application. The tool accesses existing photos stored in a device, and, upon engaging the object recognition feature, attempts to identify objects within the photo, as explored in Figure 3. In this figure Sequences 1 and 2 showcase the application processing the image, which requires 1-3 seconds, afterwards its produces suggestions regarding what the image may contains. Sequences 3,4, and 5 show the outputs from three angles of the same vehicle (the three angles are distinct photos saved to the devices camera roll prior to launching the google Photo application). The outputs suggest the accuracy is volatile based on the quality and angle of the image. In sequence 3 the Mercedes sedan is mistaken for a BMW, and in figure 5 the sedan is mistaken for an SUV, however the application does successfully identify it as a Mercedes-branded SUV, likely because the badge is prominent from the front. Sequence 4 is the most successful, offering 3 possible sedan models.

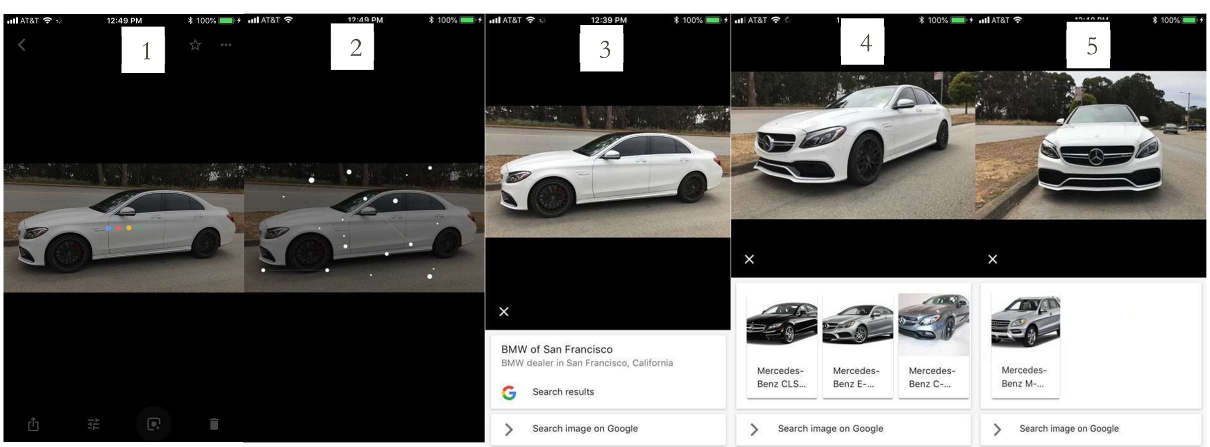


Figure 3: Sequence of screengrabs using Google Photo's image recognition feature. Sequences 1 and 2 show the application processing the image to determine the object. Sequence 3 promotes an ad for BMW, either because the application incorrectly identifies the image as a BMW or because the application cannot distinguish the vehicle as being a Mercedes and defaults to a luxury brand ad. Sequences 4 and 5 do suggest the application recognizes the vehicle as a Mercedes but in sequence 4 cannot determine the precise model so it offers 3 suggestions, while in sequence 5 is identifies the vehicle as an SUV.

The struggle with any of the technologies discussed above, and other software-based solutions, is that they are only as good as the algorithms that underpin them. They rely on deep learning, artificial intelligence and other refining techniques which ultimately produce levels of confidence when attempting to recognize objects. The thresholds for this confidence, despite improving over time, are still susceptible to failure because of physical transformations to the objects from such contributing variables as size, shape, color, angle-of-viewing, and others.

A correspondingly difficult issue for these technologies stems from the inherent similarities of products in today's consumer marketplace. As mentioned earlier, the ubiquity of on-screen tagging enhances the strength of the platform, as such, there may be similar items during the span of a show or movie that will have physical attributes that make them not only indistinguishable by the naked eye but equally as such algorithmically. Figure 4 demonstrates the similarities in a number of commonplace items; it should be apparent that identifying variables to determine the product (not necessarily limited to on-screen simultaneity of both products) is a challenge. The impact of misidentifying any of the objects is significant, as the brands are unrelated, price-points are disparate, and the revenue-model (as discussed in a proceeding section) would be inaccurate (CPC wrongly attributed to advertiser).

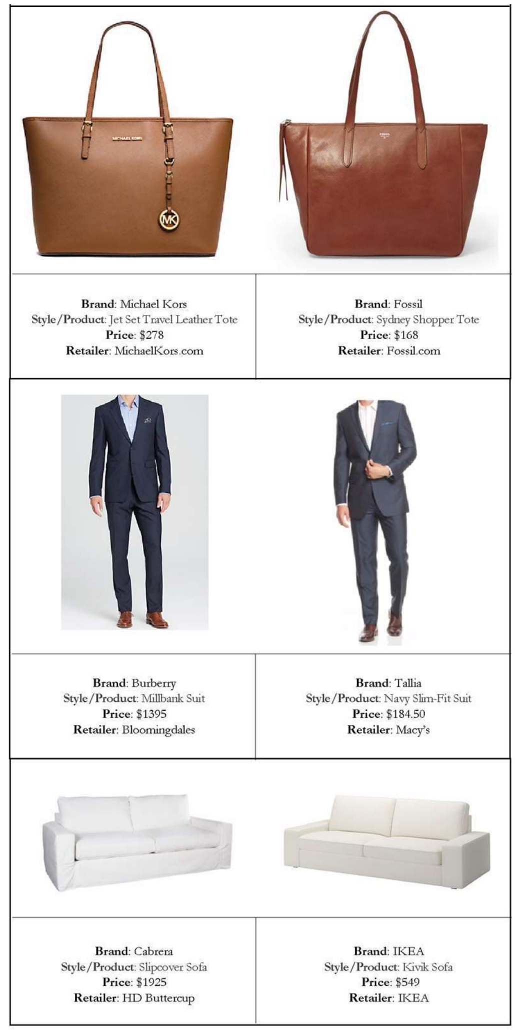


Figure 4: Examples of visually similar products but with disparate price points and distinct brands. Arguably these would be difficult, if not impossible for computer vision algorithms to distinguish.

To contextualize this concept, one can examine a show where the standard wardrobe is relatively consistent across characters; an example is the AMC show Mad Men, seen in Figure 5. Reflecting the male-dominated advertising industry in the 1960's, the cast of Mad Men often don suits (note that period pieces are not an ideal example because the on-screen props and wardrobe may be bespoke rather than being "off the shelf" and thus not available for mass consumption; the example is illustrative of the over-arching concept of similarities across multiple objects on-screen). While certain reference points can be used to provide recognition-characteristics, such as color of necktie and facial attributes, the effort and resource allocation needed to distinguish Brand A from Brand Z is inefficient and prone to error.

Figure 5: The cast of Mad Men; the image highlights the similarity of attire across characters and the challenge of identifying unique manufacturers or styles

To overcome this problem software must be partnered with hardware; as this paper proposes the hardware would take the form of wireless identification tags and readers to create a real-time location system capable of distinguishing each object regardless of the similarities between them. Variables that contribute to ambiguity for software-based systems such as lighting conditions, viewing angle, and other environmental conditions are eliminated because the mechanics of object-recognition have shifted from one reliant on vision to one dependent on communication (signal processing).

Using RFIDs in a Real-time Location System

Two technologies serve as the underpinning solution for object tracking in video as proposed in this paper: Real-Time Location Systems (RTLS) and Radio frequency identification (RFID).

Real-Time Location Systems

Real-time location systems enable people to find, track, manage, analyze and leverage information regarding where assets or people are located; this is performed by associating a tag, a small wireless device, with each person or asset. The tag is tracked using location sensors which typically have a known position. Perhaps the most common example is GPS (Global Positioning Systems) which use satellites that have a known position in orbit or cell phone towers from which a relative positioning can be ascertained from the towers' stationary placement. A number of different algorithms can be used to calculate the position of an object; similarly, a multitude of protocols can be used depending on the use case. RFID (which will be discussed in detail) is one viable protocol, however others exist including Bluetooth, Wi-Fi, GPS and cellular. Real-time location systems are implemented across industries, including healthcare to monitor the position of a patient or asset in a hospital, logistic environments to track packages throughout a warehouse, in manufacturing atmospheres to monitor parts across an assembly line, or in retail settings to track consumers' cart.

There are many considerations when implementing a location system for practical industrial use; however two critical components that traverse use-cases are beneficial to discuss here because these elements provide a fundamental understanding of the capabilities of a real-time location system. The first consideration is the ranging technique to estimate the distance between the tag and the reader. The most appropriate technique has yet to be decided for this proposal, however possible ranging methods include:

- Time of arrival: measured as the time it takes for the signal to arrive from the location sensor to the tag. The time to travel, also known as the propagation delay, can be converted into distance between the tag and sensor by multiplying it by the signals propagation speed. At high frequency, such as 2.4gHs, the signal travels at a speed approaching the speed of light.
- Angle of Arrival: the angle between the propagation direction of signal and some reference direction, which is known as orientation. The AOA uses direction sensitive antennas by the receivers to determine the direction and angle; the tag position is estimated by finding the intersection of different signal propagation paths.
- Time difference of arrival: Similar to TOA, TDOA measures the difference in transmissions times between signals but does not require the high-resolution synchronized clocks needed in TOA.
- Time of flight: TOF uses measured elapsed time for a transmission between a tag and a location sensor based on the estimated propagation speed of a typical signal; signals are sent with known departure times.
- Received Signal Strength Indicator (RSSI); this measurement gauges the power present in a received radio signal. As a signal leaves its source, it attenuates logarithmically, it however can be impacted by obstacles.[2]

The methods just described are used to calculate the range between a tag and an individual reader. However, a single reader is not sufficient to locate a tag in three-dimensional space. As the Diagram 6 demonstrates a minimum of three readers will be required. Using the illustration below as a reference, a range of 46 inches (arbitrarily chosen) may be returned from a tag (blue dot) to a reader (red dot), however there is an infinite number of points in space that are 46 inches from a tag as seen in the second frame (only 3 are shown, yet the entire circumference of the blue circle

[2] All ranging methods descriptions sourced from Malik, Ajay. RTLS for Dummies. Wilwy Publishing, Hoboken NJ. 2009 Print

represents possibilities that fulfill the 46 inch value). The introduction of a second reader reduces the possibilities to two intersections that meet the criteria from both readers. It is not until a third sensor is introduced that the placement can be ascertained because the intersection of the three sensors will occur at a single point. This estimation example demonstrates a trilateration technique; cell phone geolocation uses a similar method called triangulation in which degrees of an angle are used. Both techniques are common practice for location systems, and as such, the technology solution presented in this paper will employ a sensor array with a minimum of three readers.

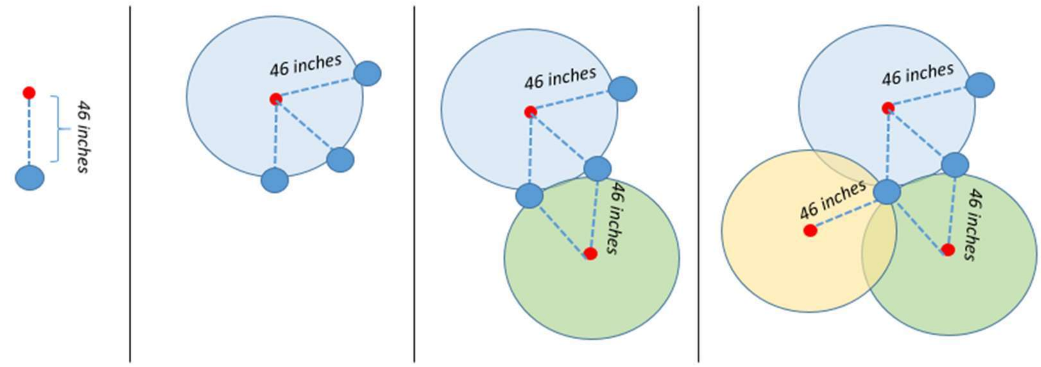


Figure 6: Trilateration illustrated; panel 1 represents distance between a tagged object (blue) and reader (red). The circle in the second panel shows all possible locations of the tagged object in relation to a single reader. Panel 3 demonstrates that the introduction of a second reader narrows the possible locations to two positions. The final panel presents the need for three readers in order to determine the true position of the tag.

The second consideration that warrants a deeper examination is the protocol by which an RTLS can operate. As mentioned, Bluetooth, Wi-Fi, GPS, cellular, and RFID are available protocols for use in a location system. Unlike the ranging technique which requires further evaluation to determine the best-fit solution, the author proposed RFID as the appropriate protocol for the rich-media use-case described throughout this paper. The selection stems from the protocol's existing use for asset identification (both as part of an RTLS and independent from it).

Radio Frequency IDs

Radio-frequency identification (RFID) is a method to transfer data wirelessly via radio waves; this is typically used for identifying and tracking purpose, and facilitated through the use of tags (which store digital information) attached to objects to be identified, and readers which receive information from the tag. Simply put, an RFID reader transmits an encoded radio signal to interrogate the tag; the RFID tag receives this message and then responds with its identification and other information. This may be only a unique tag serial number, or may be product-related information such as a stock number, lot or batch number, production date, or other specific information. Since tags have individual serial numbers, the RFID system can discriminate among several tags that might be within the range of the RFID reader and read them simultaneously.

RFID tags are used in many industries; Figure 7 showcases a variety of uses of RFIDs as well as their form-factors. Popular uses include toll collection and contactless payments, machine-readable travel documents, airport luggage logistics, employee access-management via badges, and pet identification via embedded microchips

Tags themselves are relatively basic constructs, containing at least two parts: an integrated circuit for storing and processing information, modulating and demodulating a radio-frequency (RF) signal, collecting DC power from the incident reader signal, and other specialized functions; and an antenna for receiving and transmitting the signal. The tag information is stored in a non-volatile memory. A battery may be the third component; presence of a battery classifies the tag as an "active-tag" allowing it to periodically transmit without requiring the responder to ping it. In addition, the battery provides significant increases in the distance by which the tag can transmit (up to 600 meters). A battery-less tag is classified as a passive tag and relies on the energy transmitted by the reader to power a responding signal.

Unlike tags, which can be disposable, readers are more permanent fixtures; they can be mobile, such as a handheld device, or fixed, like those mounted at particular thresholds on an assembly line. The former configuration can create interrogation zones that can track objects as they enter or leave a room, or pass through a perimeter.

While it can be argued that RFID functionality is similar to that of barcodes, tags unlike barcodes do not necessarily need to be within line of sight of the reader, and may be embedded in the tracked object. Moreover, RFID tags can be read hundreds at a time, bar codes can only be read one at a time using current devices. Further contributing to the flexibility of tags is their cost; some can be produced for pennies and can be manufactured using printers with metallic inks. In 2011, the cost of passive tags started at US$0.09 each; active tags for tracking containers, medical assets, or monitoring environmental conditions in data centers range from US$50 to US$100+. Battery-Assisted Passive (BAP) tags are priced at US$3–10.

Figure 7: Various form factors and uses of RFIDs; readers are also shown in some of the images

With a cursory understanding of RFIDs and RTLSs one can explore the amalgamation of these two technologies via a real-world example that is currently employed yet represents a unique use-case than the one presented throughout the paper. For the 2014 season, the NFL and RFID manufacturer, Zebra Technologies, outfitted 17 NFL stadiums with readers and players with RFID tags. The tags that are being worn by players are less than the circumference of a quarter and installed under the top cup of the shoulder pad of a player's uniform. Powered by a watch battery, they transmit a signal 25 times a second to readers mounted between the upper and lower decks of the stadium, allowing location mapping within 6 inches of the tag's true position. The output offers "next-gen" stats to fans & broadcasters, and give teams a higher level of analysis of player movements on the field. Precise location measurements, speed, and distance can be captured in real-time and compiled into a database allowing new experiences to be created from the information. While tangentially related to the proposal presented, this use-case serves to demonstrate how RTLS using an RFID protocol can successfully augment and enhance a traditional media experience.

Functionality/Operability

The preceding introduction to RTLS and RFIDs technologies and their capabilities provide the fundamental understanding for BeKnown's model. The following describes the resources and configuration required to enable successful execution within a production environment (television or movie), ultimately creating a dynamic tagging ecosystem for a novel advertising platform.

- An onsite (production set) human asset (tagging coordinator) will be required to program and test all RFIDs
- The tagging coordinator will use a computer system with proprietary software to program product attributes for each item. Diagram 8 reflects an illustrative example of the inputs the Tagging Coordinator would configure via the software represents. Note that the diagram represents the minimum inputs required; a broader data set of categorical inputs will be available (type, color, etc) to allow analysis of product attributes at an aggregate level.
    - The input for dimensions (HxWxD) is inputted for an object at its most-common-state. The example in the diagram is suitable for conveying the appropriate measurements of the tee shirt. A tee shirt's most-common state will be the form-factor when worn, thus the depth should be equivalent as if worn across the torso (equivalent to the depth of a chest), rather than flat (as if lying on a table).

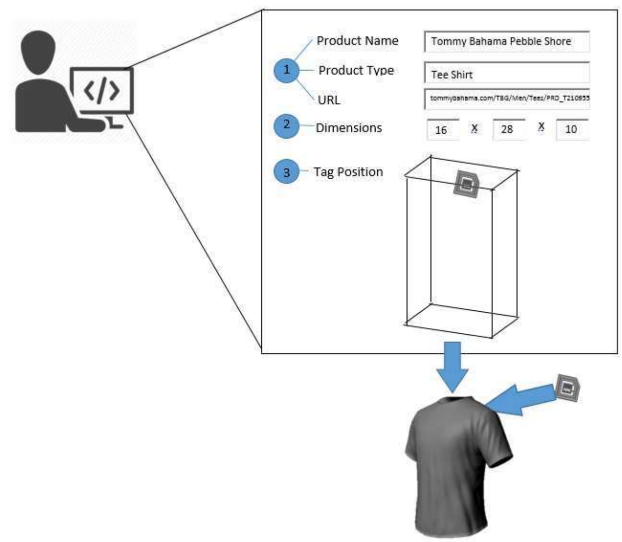


Figure 8: Label 1 reflects attributes inputted by the tagging coordinator and available for consumption by the end-user (viewer). Label 2 represents the physical measurements to product the footprint of label 3. Once the footprint is rendered, the coordinator can position the tag within this space to mirror its real-world placement on the product.

    - Once dimensions are inputted, the software will render the 3d footprint. The tagging coordinator can position the tag to mirror the appropriate placement on the physical object. This establishes a baseline for the onboard gyroscope and creates the relational positioning between the RFID and the footprint. Custom positioning is required because many objects will have surface areas not acceptable for a tag; for instance a tag should not be mounted on the lens of a pair of sunglasses but rather placed on the inside of either temple.
- Once all tags have been encoded and been physically attached to the products and tested, production can commence in the traditional manner as a camera captures the video payload. Simultaneously, a mounted RFID reader array (composed of three offset sensors) captures the placement of tags (because the sensors on the array are offset across three planes [X,Y,Z] and remain at such coordinates relative to the camera lens, known, static positions are established). Figure 9 illustrates three items tagged (shirt, jeans and shoes), coupled with the standard HD camera and the attached array with offset readers.

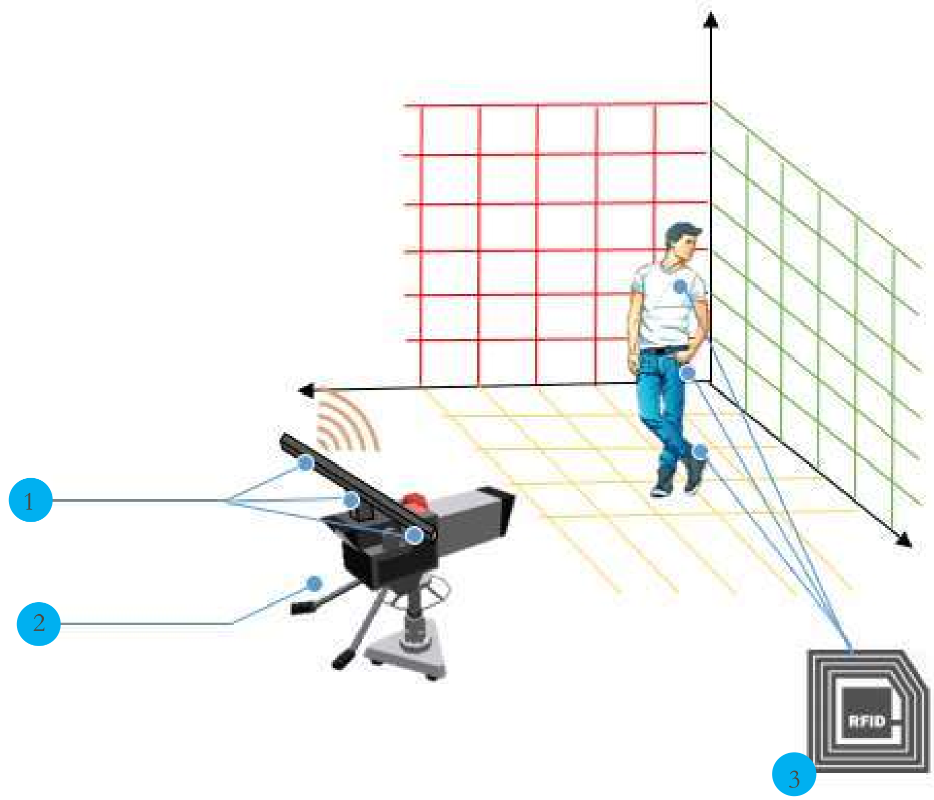


Figure 9: Simulation of a production set. The grid lines are visible merely to provide depth. The person reflects any actor within a scene of a tv show or movie. Note that this character carries through to the proceeding Figures 10 and Figure 11.

Label 1 of the figure to the right represents the offset reader array mounted to a traditional video camera (label 2). Label 3 reflects the three unique instances of an RFID tag, one the shirt, pants, and shoes of the actor. Each has been coded with the specific attributes (dimensions, name, and links) for each product

The output of the configuration will produce the following

- Panel 1 of Figure 10 illustrates the video payload which the camera captures; it represents no departure from the usual payload that the camera would capture whether part of an RF system or not.
- Panel 2 of Figure 10 illustrates the meta payload which the RF system captures. The RF tag position is identified; whether processing of the entire 3D footprint is processed in real-time via the middleware or calculated in post-production using an alternate (non-onboard) processing system needs to be determined.
- Panel 3 of Figure 10 illustrates the output after post-production. The figure is illustrative and does not represent what the viewer would see naturally; the footprints would be invisible until a hovering action is executed. At this stage, the footprints have been associated with the encoded URLs, transforming them into hotspots, allowing them to be clickable and hyperlinked.

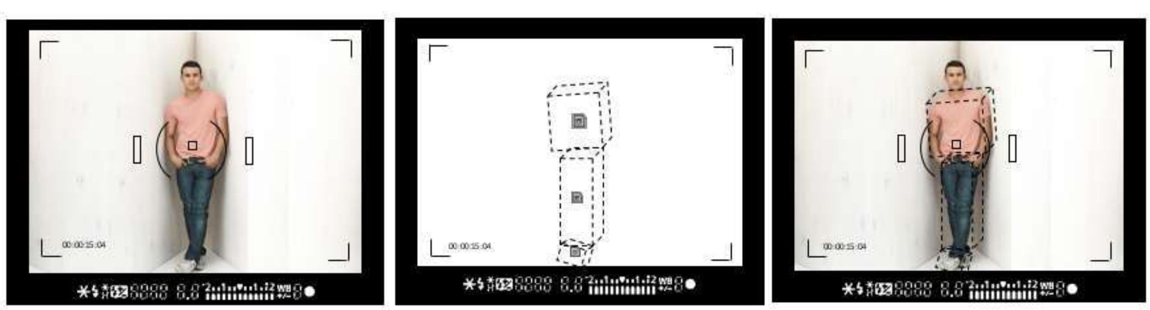


Figure 10: Panel 1 is the normal payload received by the camera. The middle panel (panel 2) represents the payload captured by the sensor array. The leftmost panel represents the merger of the video payload and digital footprint payload. The boxes are hotspots but will not be visible to the viewer unless engaged with a mouse/touch action.

- The post-production file will be a common output type that can be viewed across devices (ie HVEC, MPEG codec).
- Much like closed-captioning, the hotspot data should exist in the meta-layer and can be toggled on.
- The user experience for the end-viewer can be setup to any number of unique configurations depending on the device and firmware of the device. While some configurations may be dictated by device manufacturer and the capabilities of the device (ie touch-enabled), other configurations should be customized by the viewer.
- The panels of Figure 11 reflect three illustrative examples of what the user experience might look like on a tablet.
- Figure 11-Panel 1 displays a discrete configuration in for which a product is recognized as being tagged and lets the user "save for later", allowing continued, minimally interruptive viewing of the content.
- Figure 11- Panel 2 displays a relatively positioned popup window which serves certain attributes of the inputted product data coupled with a dynamic screenshot of linked URL page allowing the viewer to get a "snapshot" of information
- Figure 11-Panel 3 will re-render the layout into a split-screen letterbox, with the media content on the left, and a full web-enabled browser window on the right. This allows the user to engage with the advertiser's website as needed. Options can be toggled to pause the media content when using this configuration so as to allow the user to immediately click-through the web-page if desired.

Figure 11. Three possible screen configurations on the end-user's (viewer's) device during playback. The leftmost (panel 1) is a passive setup allowing a user to load product information later. The middle panel (panel 2) demonstrates a popup-on-hover window leveraging an inset browser. The final panel (panel 3) uses a split-screen configuration; it is only engaged on click or hover, and returns to full-screen after the user exits the product pane.

Revenue Model

Revenue can be generated from a three-fold model:

- The primary source of revenue is through a familiar ad model used across advertising platforms in which an advertiser pays on a Cost-Per-Click (CPC) basis. The model for BeKnown may be Cost-per-Click or Cost-per-Hover since information can be conveyed using a mouse hover. Product engagement is tracked upon click and sent to internal BeKnown servers; a redirect from the server passes the user onto the original destination. The importance of this measurement cannot be understated, as ROI from current product placement efforts is difficult to track, serving as a missed opportunity for advertisers to understand how programming contributes to product awareness and purchase decisions. A click model provides an evidence-based approach to understanding the influence of placements and their value across a variety of programming.
- The secondary revenue source is derived from the sale of advanced analytics information to advertisers, consulting firms, retail industry experts et al. When provided to advertisers, the information may include such data as frequency-of-click-per-timestamp, time-on-screen, and other calculated metrics that can provide robust insight for retailers. Take for example, an on-screen character driving the latest model from BMW. BeKnown

will receive the metadata of all tags for all frames during post-processing (layer sync). From this data, BeKnown can calculate the average duration that a viewer is exposed to an item before it becomes compelling. If the majority of users click on a product at the 2:01:013 timestamp, BeKnown can associate this with the frame to determine if a specific interaction took place between the character and the product (ie. the character uses the self-parking feature of the vehicle). This information becomes invaluable to companies as they attempt to determine what attributes pique interest or induce curiosity. Moreover this information can be anonymized and rolled up at the industry level to be sold to retail consultants as research tools. For instance, aggregation of in-programming products of vehicles may reveal black cars that are being driven prompt higher click-through rates than silver cars that are stationary.

- The AI and ML revolution will drive the third revenue stream. As noted previously, computer vision often requires a library of "absolute truth" images by which the algorithm can compare the images in question to determine if the visible attributes can be reconciled against those known images. Some of those libraries of known images are publicly accessible; one of the most leveraged for academia is the open-source project maintained by Princeton University and Stanford University dubbed "imagenet" which boasts over 14 million images. The primary limitation the author would argue is that these images are generic, classified at very high-level terms. A search query for "cars" using Imagenet produces a number of sub-categories, however none are brands (Jeep/LandRover is an exception because of the iconic status). Popular makes like Honda or Ford do not have a library, rather styles such as "convertible" or "minvan" are the lowest level in the hierarchy, as shown in Figure 12. Given this constraint, BeKnown can provide supplements to these libraries since the

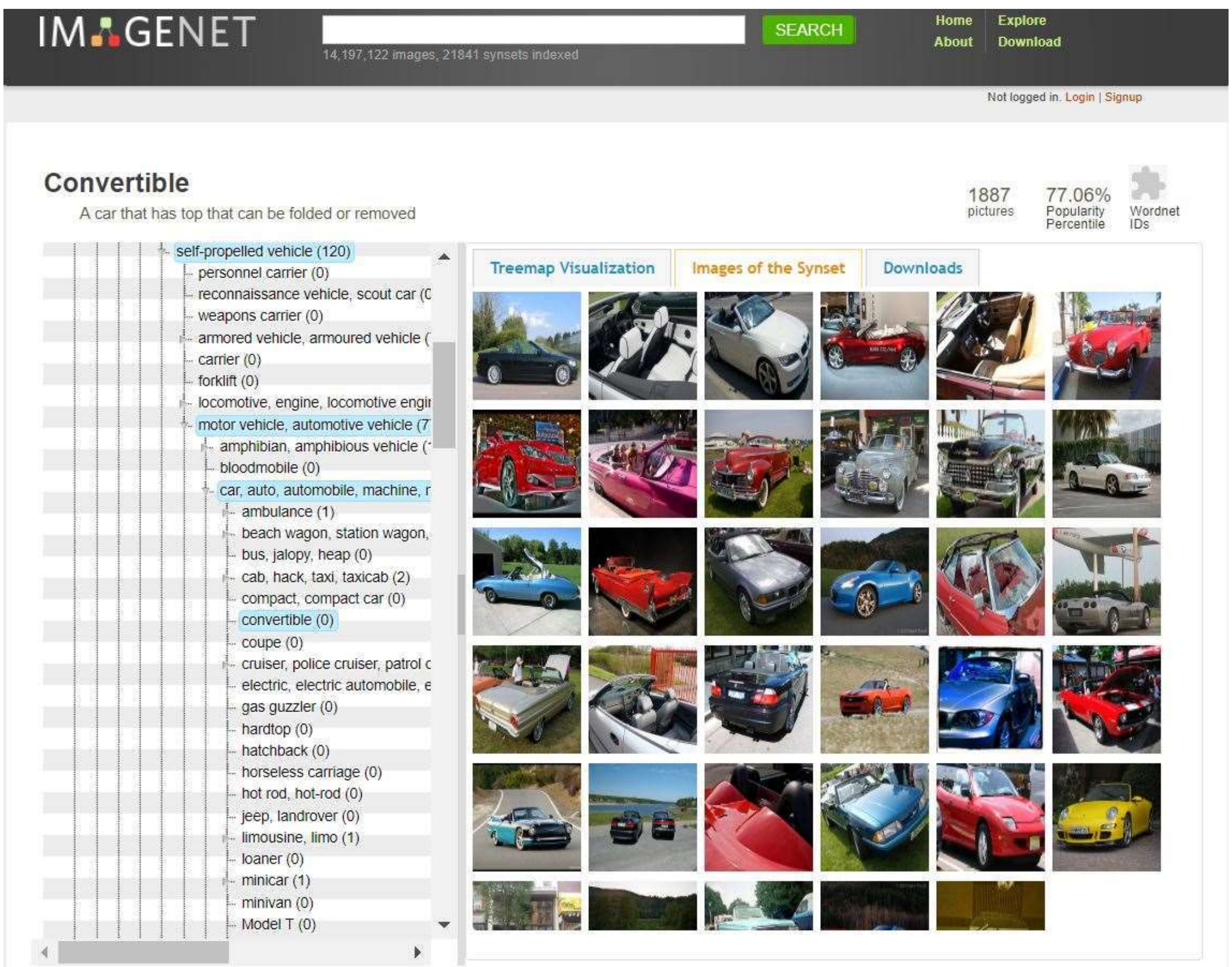


Figure 12: Screen grab of ImageNet, showing the output of a query for "cars", filtered in the left pane using the "convertibles" directory/library. The main pane showcases a few of the images in the directory that are used as reference images to determine if the target image is a convertible. No brand or model/makes are available unless considered iconic (jeep/landrover, Model-T)

system necessitates the hard-coding of each tagged item with brand and model. This information persists as each frame is captured. From this archive, a rich set of images can be derived which can provide a multitude of angles for that object (originating from either the camera's repositioning in the scene [orbiting, etc] or the item's positioning as it moves freely. As such, each video frame may represent a slightly nuanced snapshot of the product, ultimately creating hundreds or thousands of verified unique views. This robust library can be sold to institutions, scientists, or engineers who require precision in evolving their computer vision applications.

Appropriate Use-Cases

Not all video media is appropriate for embedded branding. This is not necessarily a result of any technical constraints that might come from implementing a RTLS/RFID system, but rather stems from the nature of the content. Outlined below are the use-cases, and justification thereof, which are most conducive for product placement.

- Music Videos
    - Pop music often makes references to clothing brands/styles, accessories, cars, alcohol brands, etc in the lyrics of the song
    - If products are not expressly referenced in the lyrics, the stylized nature of many videos inherently lend themselves to products being on screen because they convey a particular lifestyle
    - The short length of music videos (4-5 minutes) increase the opportunity for repeat exposure
    - The influence of the artist can accelerate interest in the product (see section **Influence of Music Videos**)
- Sitcoms
    - Sitcoms provide settings to which the viewer can relate because of the commonplace setting in which many of these are scripted; the environments, whether a living room, kitchen, or other room offer dozens of tagging opportunities. All objects in the background or foreground can be tagged including
        - Furniture
        - Artwork
        - Appliances
        - Décor accessories (table lamps, area rugs, etc)
        - Clothing and accessories worn by all on-screen characters

Tier 2 opportunities (opportunities after feasibility has been validated)

- Movies
    - Requires greater tagging effort because of number of products and duration of film
    - More likely to include environmental variables such as wider panning shots, explosions, or other factors that might skew locating techniques.
    - Use of CGI might remove physical presence of a product (ie cars used in extreme action scenes)

Inappropriate Use cases include

- Sporting events: In-programming placement of products is not conducive since players don uniforms and props are not used
- Period Pieces: the lack of modern products coupled with many custom-produced props make content that is not set in present-day poor candidates

Evidence-Based Support

Influence of Music Videos

A use case that has played a prominent role in evaluating the effectiveness of embedded branding has been Lady Gaga's video of the 2009 hit "Bad Romance." According to an article in Quirks Marketing Research Review entitled "*Eye Tracking Product Placement and Lady Gaga: What Bad Romance can teach us about Embedded Branding*" the video has been a favorite to examine because it has no less than seven placements in its five minute duration. One such study was designed and executed by Key Lime Interactive and SensoMotoric Instruments using an eye-tracking tool to determine "if there are significant trends in size of branded products or length of exposure that reliably result in product recall, and to use preliminary data to design a study that could attempt to determine ideal parameters for embedding products without disturbing the viewers' experience."[3] The results of the experiment determined that 56% of views were able to recall at least one of the products advertised (those in the <34yr old-age bracket were able to recollect significantly more, likely because the video served this target demographic). In addition, the output of the visual heat-mapping demonstrated the amount of time a viewer dwelled on a product increased the ability to recall the product. Finally, it was noted that the frequency of appearance of a particular product had a slight statistical significance to product recall. Together, the amount of time a viewer dwelled on a product, coupled with the number of appearances, explained 92% of product recall.

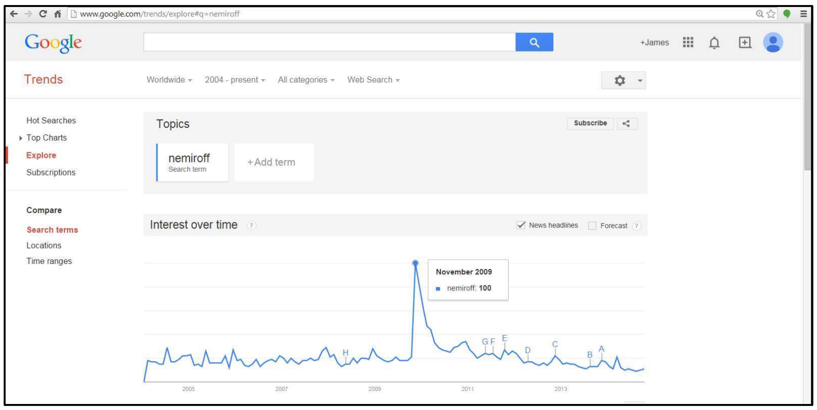


Figure 13: Output of Google Trends Tool using the search tern "Nemiroff." The spike coincides with the release of the "Bad Romance" video on November28th , 2009.

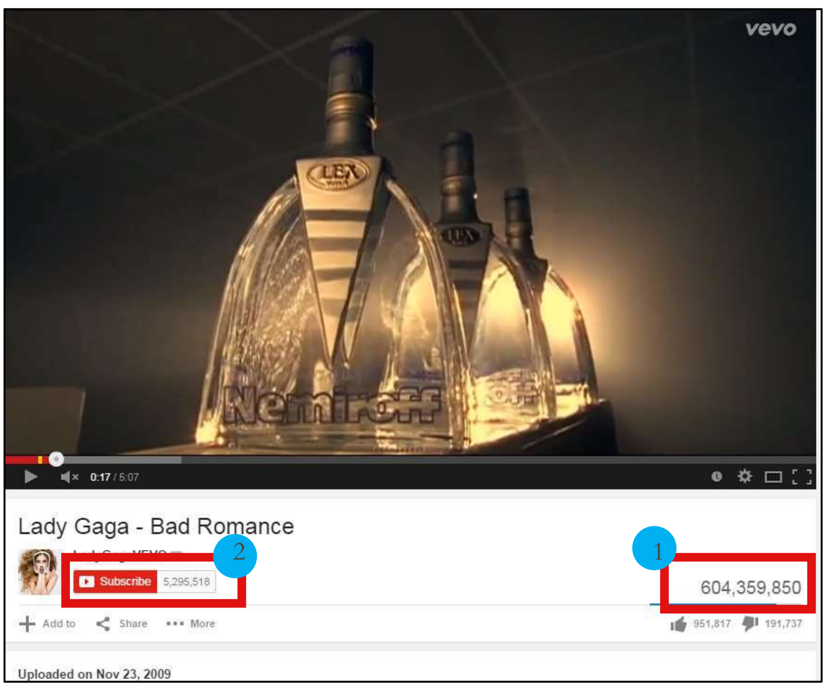


Figure 14: A screengrab from YouTube of Nemiroff Vodka from Lady Gaga's "Bad Romance" music video. Label 1 highlights the number of views on YouTube. Label 2 highlights the date the video was posted to YouTube. This data coincides with the spike seen in Figure 13.

[3] Knodler, Jennifer. Eye Tracking, product placement and Lady Gaga: What Bad Romance can teach us about embedded branding. Quirk Marketing Research Review. http://www.smivision.com/fileadmin/user_upload/downloads/case_studies/article_quirks_smi_csladygag.pdf

The outputs of the experiment are persuasive, however, the study by Key Lime Interactive did not measure users' interest or propensity to take action; it only gauged memory. To determine whether, in fact, the placements were significantly compelling to cause users to the take action, the author of this paper (Chambers) leveraged data aggregated by Google's Trend tool in a rudimentary experiment. The tool plots global search queries over time. By entering a product name, in this case "Nemiroff", a type of vodka represented in the video, one can measure the volume of queries performed by internet users; the output can be leveraged to reveal any dates or date-ranges in which significant changes occurred. The output of the tool, when using the search term "Nemiroff", is seen in Figure 13; a substantial spike is visible for the November 2009 time period which coincides with the release of the video across mediums (including YouTube [Nov 28th 2009]). Of course this spike might be attributed to a global unified marketing campaign executed by the vodka brand and its agencies across multiple channels including print, TV, and banners. A cursory search did not however reveal that such efforts coincided with this date, nor can it be proven that the promotion of the brand was isolated to solely the Lady Gaga video.

Beyond the presumption of correlation between view and engagement (in the form of search, in this case), the sheer volume of views as reflected in Figure 14 (Label 1) serves to validate that music videos are an ideal media platform for brands to get exposure. The short duration, coupled with compelling audio and video draw unique visitors and likely bolster high non-unique visitors (return visitors), contributing to over 600 million views.

A second notable example that provides evidence regarding viewers' recall of product placement and the influence of such to elicit a behavior is that which occurred during the 2014 State of the Union address. This case is particularly unique because it involves paid and organic placements. Viewers of the SOFTU identified Michelle Obama's attire as mirroring that of Julianna Margulies' character on the TV drama "The Good Wife" (Figure15). The paid placement (worn by Margulies) was impactful enough to be recalled by viewers, who used Twitter to broadcast the relation. It is not publicly known what the direct impact was on sales for the manufacturer, Michael Kors, following the original air-date of "The Good Wife". However, when augmented by that of Michelle Obama, the Michael Kors online store immediately sold out of inventory of the dress.[4] While one may argue that the influence of a figure with the prominence of a First Lady would prompt noteworthy increases to any designer's sales, it can be rebutted that viewers must first be educated about the products seen on-screen. Twitter and online articles post-facto are not the most efficient or effective channels by which viewers should be educated; in-programming product placement serves as more appropriate real-time mechanism for identification.

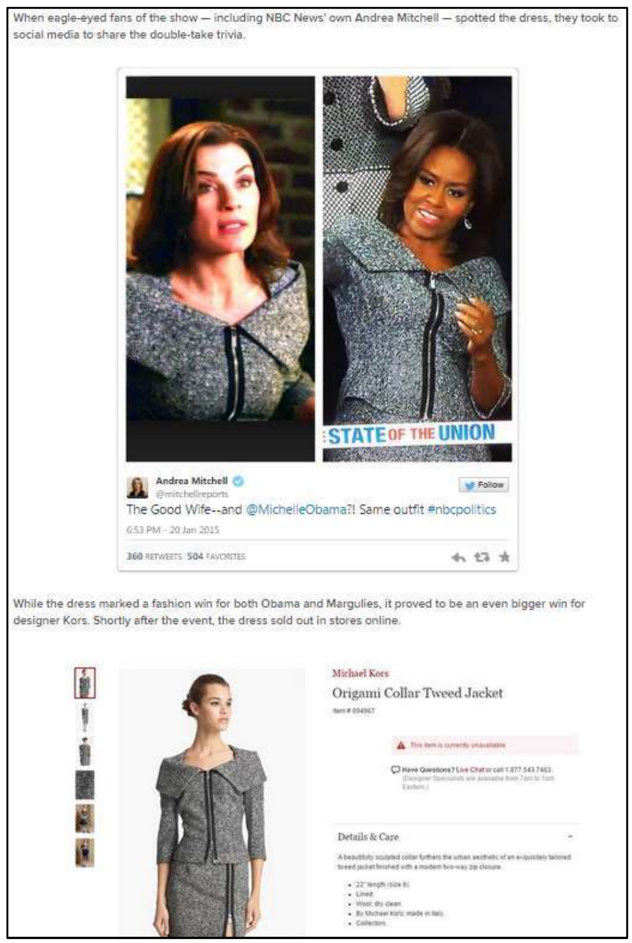


Figure 15: An excerpt from an online article (Today.com) which showcases the placement of a product in the TV drama "The Good Wife" and the same product worn by Michelle Obama during the 2014 State of the Union address. Also pictured is a screen-grab from the dress' manufacturer (Michael Kors) online store, in which the dree sold out shortly after it was donned by Michelle Obama.

[4] Hines, Ree. "A 'Good' look for Michelle Obama: First Lady Channels 'Good Wife' at SOTU. Today.com http://www.today.com/style/first-lady-michelle-obama-channels-good-wife-state-union-address-2D80438888?google_editors_picks=true

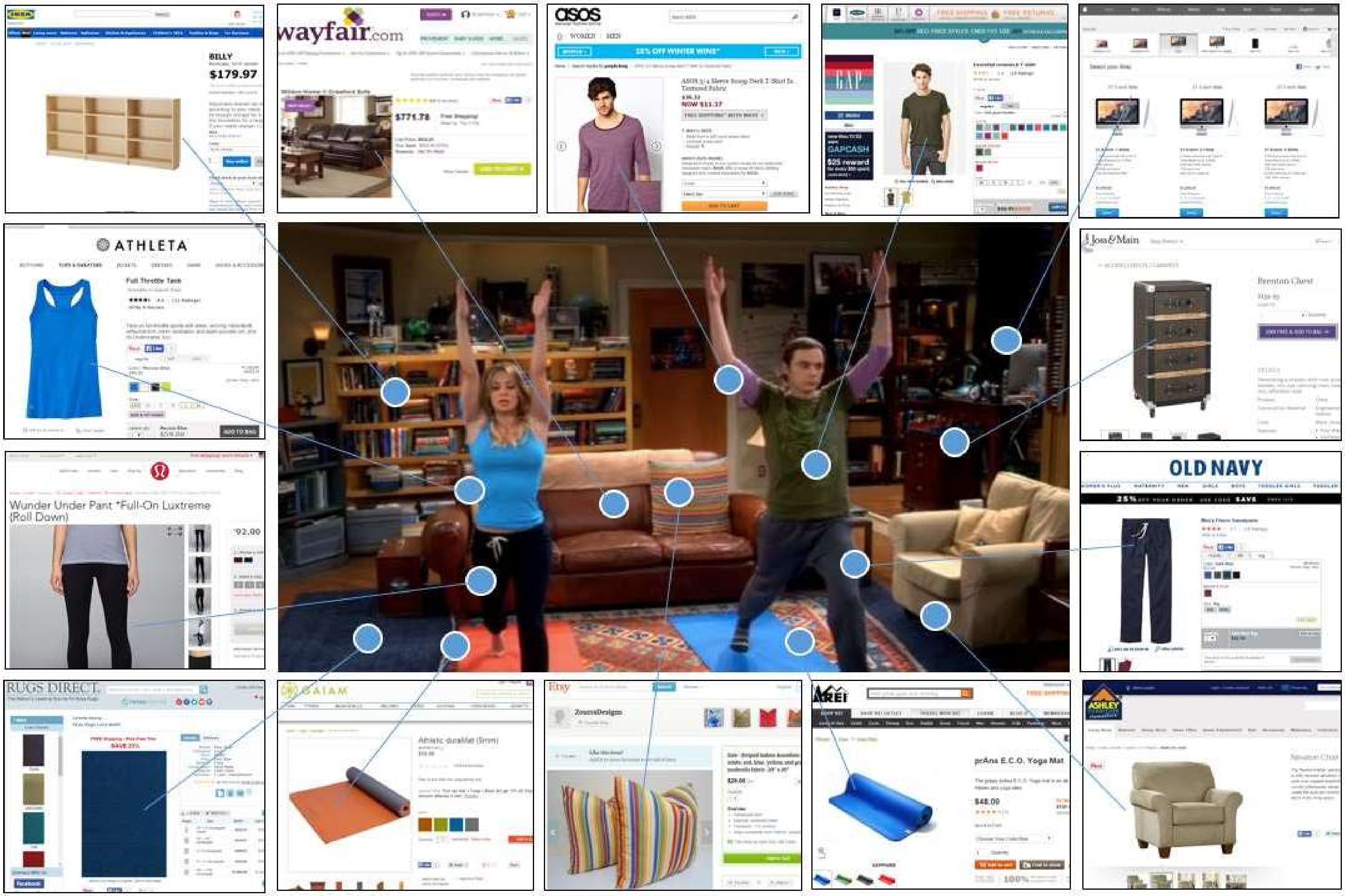
Challenges

Some marketers or researchers might suggest, as Jennifer Knodler does in her article regarding Lady Gaga, that "once metrics [regarding contextual placement] are established, it's not far-fetched to think that product placement space in pop culture could be bought and sold at standard rate in the same way you would opt for a fourth-page ad vs fill-page ad in a print publication." However, the author of this paper (Chambers) would posit that this is a slippery slope, and while the data is valuable and can be sold to inform retailers about certain attributes, it should not enable the advertisers to dictate creative direction for movie or television by altering the manner in which producers or directors script or direct their works.

Conclusion

This paper has attempted to convey a novel technological solution to create a new advertising platform for visual media TV, movies) that allows users to engage with products on-screen. Existing digital advertising revenue models (CPC, CPM) allow this platform to generate revenue immediately upon implementation. This document is not intended to wholly address all technical requirements; rather it is meant to serve as springboard to commence conversations with those industry experts who have in-depth knowledge regarding radio frequency identification and real-time location systems. From these conversations the feasibility of the proposal can be vetted, and in the case that viability exists, to determine the appropriate ranging methods and other high-level engineering requirements discussed in paper, allowing further refinement of this proposal, and eventual prototyping.

Appendix 1: Illustrative Product-Placement Opportunities



A mockup using a frame from the sitcom "The Big Bang Theory" reflecting a significant volume of on-screen tagged items. Note that the blue dots do not represent the hotspots for the associated item, they are illustrative markers to identify tagged items. The hotspot would be visible to a viewer on hover, and encompass the footprint as calculated by the system's logic engine (sourced from the values inputted by the tagging coordinator).

Appendix 2: Technical Functionality (production-ready)

System Components

A cursory exploration of the technical specifications was conveyed in the body of the paper. This section provides a more granular look at the requirements of the RTLS and RFID ecosystem that will allow enable the operability described throughout.

- A series of RFID tags
  - The tags must be as small as possible (preferably with printed antennas), to allow discrete attachment to products with small footprints such as wristwatches, eyewear, smartphones
  - The type of tags needs to be considered, largely as a function of size-constraints and range. If case warrants active tags, then a cell battery would be required
  - Capable of transmitting data 100meters
  - Life of battery does not need to exceed one-year
  - Production cost of tags at scale should be under $20
  - Will include micro-electromechanical (MEM) gyroscope to measure orientation
- End-User Software
  - Allows users to program information onto RFIDs; includes following attributes
    - Product Description (brand, type, color)
    - Dimensions (height, width, depth): to be coded for the native-shape of object, that is, the state in which the object will be most utilized on screen. For instance a the dimensions of a wristwatch should not be entered to reflect the length of the bands when lying flat, rather it should be measured as if the watch is being worn, with the bands wrapping around the wrist and latched.
    - Position of the RFID within the dimensional space defined. Many objects will not allow a centrally positioned tag, thus the software must allow the user to position the RFID accordingly (See Figure 7).
- Location Engine
  - Use ranging and positioning estimation techniques to determine the spatial placement of the RFID
    - Best ranging technique for case is to-be determined; can include "Time of Arrival", "Angle of Arrival", "Time Difference of Arrival", "Time of Flight", "Round Trip Time", or "Received Signal Strength Indicator" (as discussed earlier).
    - Best position estimation technique for case is to-be-determined; can include "Triangulation" or "Trilateration"
- Middleware (Processing engine)
  - To determine the entirety of the objects visual footprint, the middleware will consider the following variables:
    - Location of RFID in space
    - Dimensions of object encoded on RFID
    - Position of RFID on object as programmed on RFID
    - Feedback (angular velocity) from gyroscope embedded on tag
    - Focal Length of the camera: this variable is necessary since a camera can change the size of scene without physically moving the camera (and in turn moving the mounted sensor array) via lens setting. As such, a wired connection between the reader array and the camera is

necessary in which a serial data connection is utilized to capture the changes in the zoom. In addition, this connection will pass the timestamp to ensure frames can be synced (both metadata are considered ancillary non-video payload).

        - It should be noted, that if accurate bounding of the object cannot be achieved through RF-localization and orientation, then artificial intelligence/computer vision can augment the general positioning. The dual methods are important because the localization effort accomplishes tasks where AI/CV underperforms, specifically, it removes the need for the compute power to scan all frames within the standard HD framerate to search for any number of objects. Here the location system (positioning + buffered bounding box [defined as an increase to the original inputted dimensional values in order to accommodate inaccuracies in localization]) can carve the frames in to manageable sections because of the approximate position of the tag. With a significantly smaller area, coupled with the inputted object metadata (type="apparel, suit") or even images scraped from the inputted web address for the object, the AI algorithms have a manageable area to search for a single, known object, rather than searching against all images in its tag-library.

- RFID Reader Array
    - Consists of three RFID anchors
    - Simulates three discreet fixed points in a traditional RTLS system because distance and location between each is both known, and unchanging. However, this use case allows the grouped anchors and camera to move as a single-unit as the filming-direction warrants.
    - Readers are offset from one another to provide unique X, Y, and Z coordinates
    - Readers provide tag positioning with accuracy within 10cm
    - Array receives all data encoded on individual tags
    - Processing unit can be built into array's housing

Appendix 3: Minimally Viable Product (feasibility)

The objective of the Minimally Viable Product (MVP) is to leverage off-the-shelf RTLS hardware in order to develop a *software* prototype. Software is emphasized because despite the production-ready solution having a combination of hardware and software, in order to get to a demonstration state for feasibility-testing software development is being prioritized. Some requirements for the future-state will be conveyed here but should be noted as considerations rather than a component of the MVP's output.

The selected hardware for use in the MVP will act as a proxy for the to-be designed hardware. Components of the proxy hardware may be considered for inclusion in the final hardware selection. It is expected that the developed software will be able to accept inputs from alternative hardware solutions with few core-code changes.

The off-the-shelf hardware solution chosen for the MVP is the TREK1000: Two-Way-Ranging (TWR) RTLS IC Evaluation Kit developed by DecaWave. This solution is a Two-Way-Ranging (TWR) RTLS (Real Time Location Systems) leveraging DecaWave's Ultra-Wideband (UWB) IC. It was selected because it natively boasts superior location and tracking accuracy and equally as important, includes access to the source code which can be used as a starting point for those developing their own products. The DW1000 chip which underpins the system features:

- X-Y location accuracy typically <20cm
- IEEE802.15.4-2011 UWB compliant
- Up to 6.8Mbits/s data rates
- Low power consumption
- Support high tag densities (short packet durations)
- Support long ranges (coherent receiver technologies)
- Reliable communications (immunity to multipath fading)
- Enables cost effective solutions (single-chip, 6mm x 6mm)

The DecaWave system consists of 4 boards that can be configured for a number of distinct use-cases. The configuration and specification of the boards are shown in figure 16 and figure 17. As noted, elements of this hardware (the DecaWave UWB IC) would be considered for the proprietary hardware development. The footprint of the evaluation boards are significantly larger than needed because of the components embedded in the board for configuration such as the readout display, various jumpers and DIP switches.

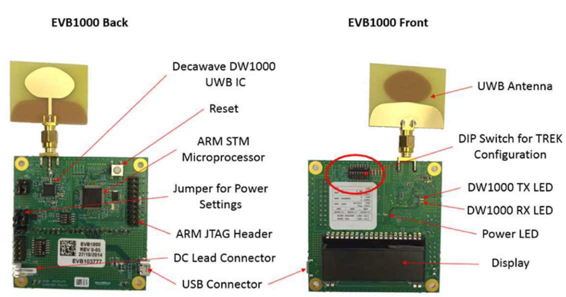


Figure 16: Front & back views of the evaluation board with components labeled

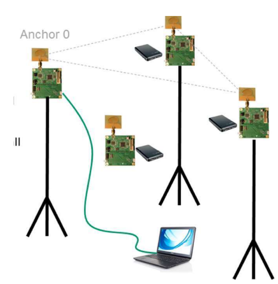


Figure 17: A "traditional" configuration with anchors placed in a manner to facilitate localization of a tag on a horizontal plane (floor plan)

The configuration settings of the various switches do not warrant being codified here, as they are available in the DecaWave Getting Started Manuals (attached as addendum to the MVP). The execution of additional steps as outlined in the manual (installation of drivers and software, connection of power supplies to the tags, etc) will allow the system to immediately begin tracking whichever board was configured as the tag (via Dip switches). This tracking can be refined and plotted via the DecaWave software per figure 18. The output depicts four dots mapped in 2D space; three of the dots represent static anchors while the fourth dot is the tag. As that tag moves within the perimeter created by the anchors, the corresponding "dot" will move in near-real-time on the software output. An additional feature of the bundled software is the ability to add a "floorplan"; this is an image or schema that can be uploaded and displayed as a layer underneath the dot-tracking. This would allow a user to reconcile where a tags placement against a mapped/physcial location assuming the anchors are placed such that the mapped area is consistent with the coverage area of the anchors.

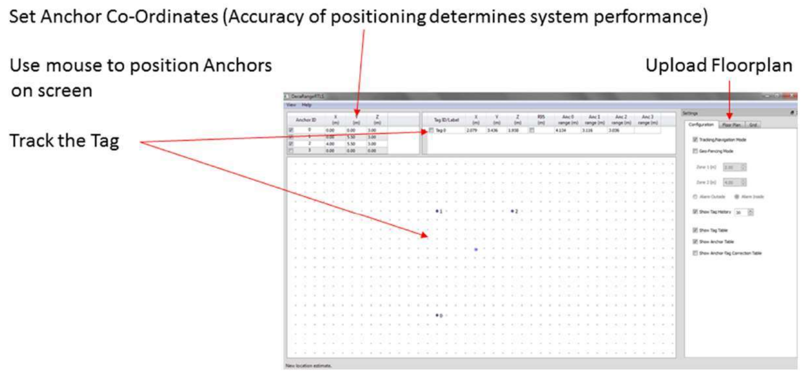


Figure 18: The bundled software provides a multi-pane layout; the top pane prompts the user for the anchor positions in relation to one another. The main pane is the tracking grid that will visually show positions of the anchors and tag in 2D space. The right pane provides a number of advanced features including an option to upload a "floorplan" or other imagery that would be displayed as the backmost layer in the main pane so that the tracking indicators (dots) would appear on top of the floor plan, the latter of which represents the physical environment where the tracking is taking place

Examples of the software in practice will be used throughout this document to demonstrate how it can serve as the basis for more advanced development that will define the MVP-requirements.

Other Hardware components

The system that is being developed consists of additional elements beyond the DecaWave Trek1000. This section describes each component and the configuration.

Component	Image	Description	Qty	Cost	Total Cost
Canon XH G1 camcorder		Semi-professional camcorder; model selected because of inclusion of time-code output. This feature is common on professional grade cameras and produces a timecode/timestamp that can be outputted to other devices to allow post-production syncing of multiple cameras, sound devices, or other digital media	1	$1163.58 (used)	$1163.58 (used)

SmallRig Professional Camera Cage		Generic camera cage. Cages are used to mount various hardware together such as microphones, and lighting units. In this use-case rods are mounted to the cage, these rods provide ample distance between the anchors	1	$79.99	$79.99
Kamisafe VT-1500 65"/166cm Adjustable Camera Video Tripod		Generic tripod used to mount the cage, camera, rods, and anchors	1	$75.99	$75.99
3.7V 350mAh lithium-ion rechargeable battery		Power source for the tag. A smaller non-rechargeable, semi-flexible battery is preferred for to-be hardware	1	$6.95	6.95
Dell Latitude E6410 Core i5		Model was selected because of the inclusion of the 1394 firewire port to allow transfer of video from camcorder to computer	1	$119.00 (used)	$119.00
AmazonBasics Brand USB 2.0 A-Male to Micro B Cable (3 Pack), 3 feet, Black		Generic USB to micro cables; these provide power to the anchors	3 per pack	$7.99	$7.99
AmazonBasics 4-Port USB 2.0 Ultra-Mini Hub		USB hub for consolidation of generic USB cables into single input into computer	1	$6.99	$6.99

Eightwood Brand 75ohn BNC plug male to 3.5MM male coaxial power audio cable		Cabling used in conjunction with camcorders timecode-out port. This produces the master timecode that is accepted by the computer and will be written to the hotspot payload	1	$11.59	$11.59
GearIt 3 FT 4 Pin to 4 Pin Firewire DV iLink Male to Male IEEE 1394 Cable		Generic firewire cable used to transfer video payload from camcorder to computer	1	$4.54	$4.54
Camvate brand 15mm Rod clamp with ¼" threaded hole		Mounting bracket used to connect rods to cage	3	$8.00	$24.00
SmallRig brand 40cm/16in long, 15mm wide 2pack aluminum alloy rod with M12 female thread		Rods used to provide ample distance between anchors. Mounted to the cage using rod clamps	3	$22.00	$66.00
Jiafeng brand Universal easy-clip grip Handlebar bike Mount Holder for smartphones		Generic spring-loading clamps that attach to rods and grasp anchors	3	$9.99	$29.97

Setup

Figure 19 showcases the constructed rig sitting atop a tripod. Each anchor extends out from the cage on a rod. A tag is positioned off-screen, in view of the camera and readable by the anchor tags. It should be noted that this setup is equivalent to the manner by which the DecaWave system was intended, it however is rotated from the position shown in inset of Figure 19. Here the anchors face outward parallel to a horizontal plane to capture signals ahead of it, as the camera concurrently captures action in a traditional tripod position. The intended use case is mounting the anchors at points near a ceiling of a given area to create a perimeter; the tracking tag moves below the mounting threshold of the anchors; the position is then triangulated.

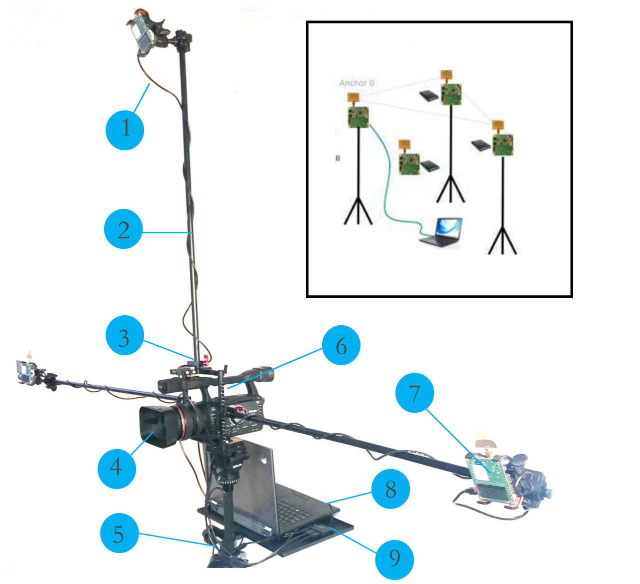


Figure 19: Assembled testing rig; the inset demonstrates the intended configuration. The assembled rig maintains the configuration of the originally intended use, but is rotated to track against a vertical plane rather than horizontal

The construction of the rig

1. USB 2.0 A-Male to Micro B Cable
2. 40cm/16in longaluminum alloy rod
3. 15mm Rod clamp
4. Canon XH G1 camcorder
5. Tripod
6. Camera Cage
7. Decawave Anchor Board attached using grip clip
8. Dell Latitude Laptop
9. 4-Port USB 2.0 Ultra-Mini Hub

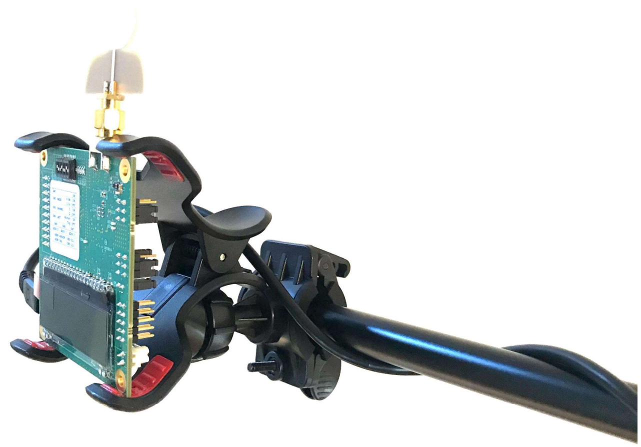


Figure 20: Close up of the anchor board, secured to the rod using generic grip clip. USB cables are used to power the board. The cables plug into the computer via a USB hub (not shown)

Figure 20: Close up of the camera inside the cage, clamps are used to attach the rods, and ultimately the anchor boards to the rig, creating a single unit that moves in unison

Preface to Software Capabilities: Out-of-the-Box Solution

The capabilities of the DecaWave's bundled software warrants describing because it provides core functionality that needs replicated and enhanced upon. As noted above, the common configuration for the DecaWave system is one in which the anchors are mounted at an area above the tag's field of movement, creating a coverage area. In the use case conveyed here, the entire plane is rotated such that the anchors are mounted on a tripod facing outward parallel a horizontal plane, and the tag would move in various paths in front of the tripod.

The bundled software requires the configuration of several inputs, specifically the distance between each anchor. Once inputted the anchors will appear in the main pane; when powered on the tag should also be displayed. As the tag moves in the physical world, the dot depicting the tag will move correspondingly in the main pane (scale can be adjusted using settings in the right pane).

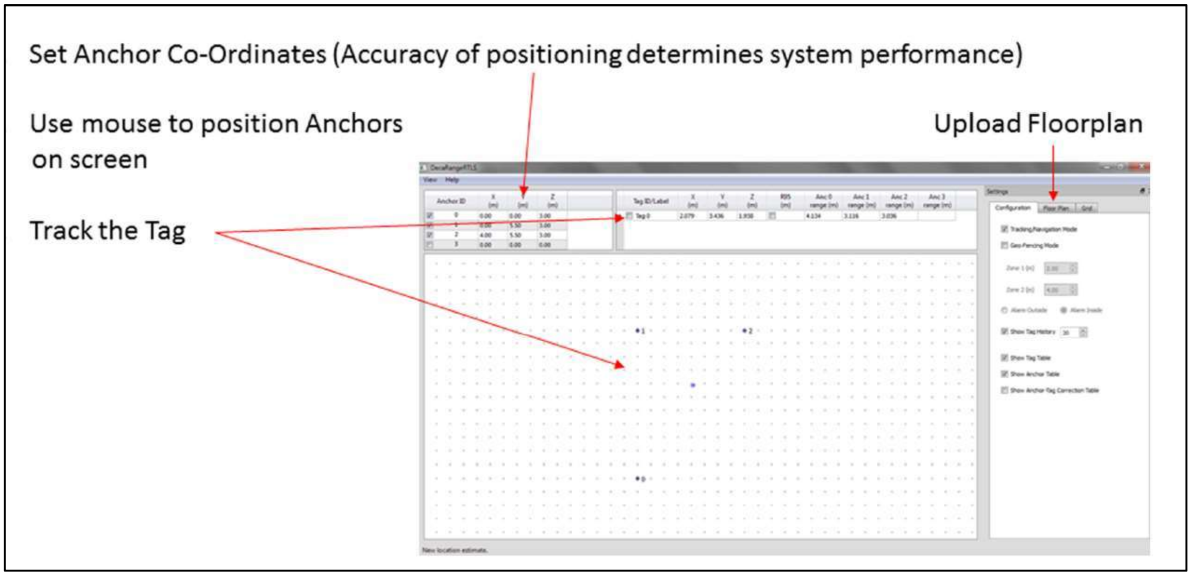


Figure 21: Screengrab of the bundled software with callouts of certain features, including the anchor placement-configuration, the localization plot and advanced settings to upload a floorplan

The software includes a feature that allows users to upload a floorplan. The intent of this functionality is to underlay a representation of the tracking area with attributes such as walls, doorways, furniture or other such objects that inform the user of a tags location within a mapped physical location.

In the following demonstration the floorplan feature will be utilized in a novel manner by uploading a static image representative of the camera's range of view (for the MVP the underlying image should be the livestream of the camera's videofeed). As noted, this can be achieved because the scene is being rotated, maintaining the relative position of the tags-to-anchors on parallel planes, but rather than being orchestrated with the anchors at the ceiling-level and the tag at a particular height from the floor, the anchors and tag are position in a wall-to-wall scenario (two planes vertically parallel).

Figure 22 illustrates the software with a static image as the underlay; three dots (sans circle) form a triangle reflecting the scaled position of the anchors on a plane that sits 15-20feet from the scene. The dot marked with the green circle represents the current location of the tag. It is important to note that despite the tag traversing the interior triangle formed by the anchors, the tag remains accurately tracked. This suggests that the trilateration method is not limited to the distance of the anchors mounted to the camera rig.

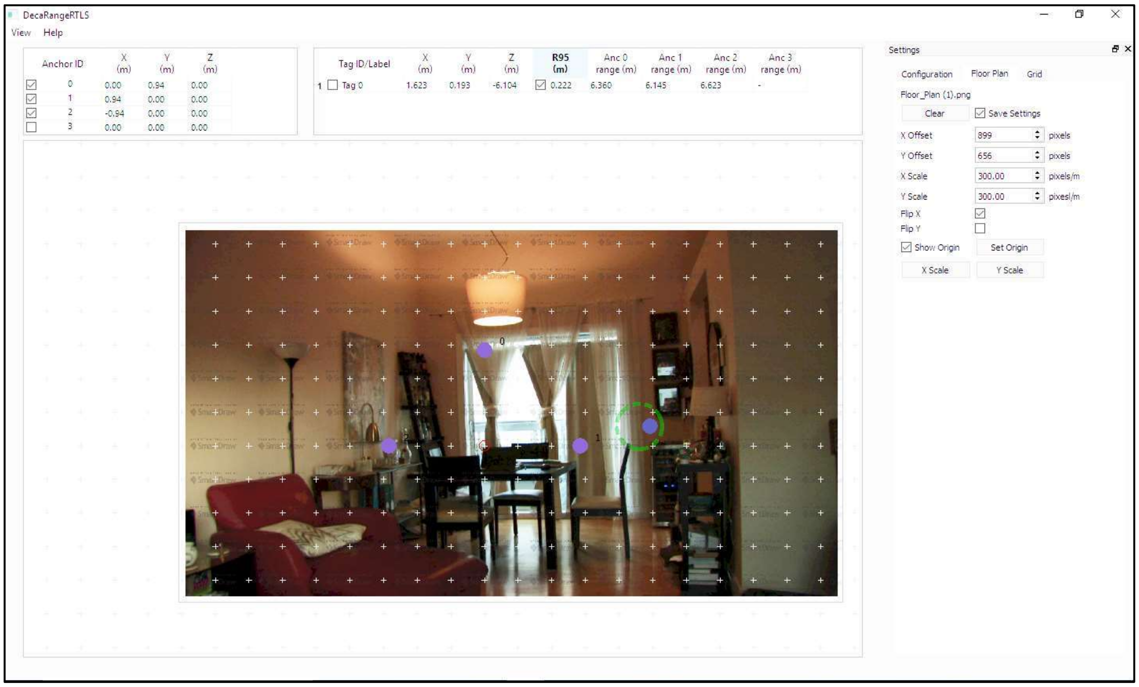


Figure 22: Screengrab of the bundled software with callouts of certain features, including the anchor placement-configuration, the localization plot and advanced settings to upload a floorplan

Figure 23 showcases three positions of the tag taken several minutes apart. The background image was intentionally captured without the tag in the scene, so that it could be used for all three sequences, however the virtual mapping of the three positions of the tag (with the green circle) does in fact align with its physical position, despite not being visible. It should also be noted that the software does support a Z axis via the inclusion of a third anchor. That capability is not demonstrated here but should be considered for version2.0 of the MVP.

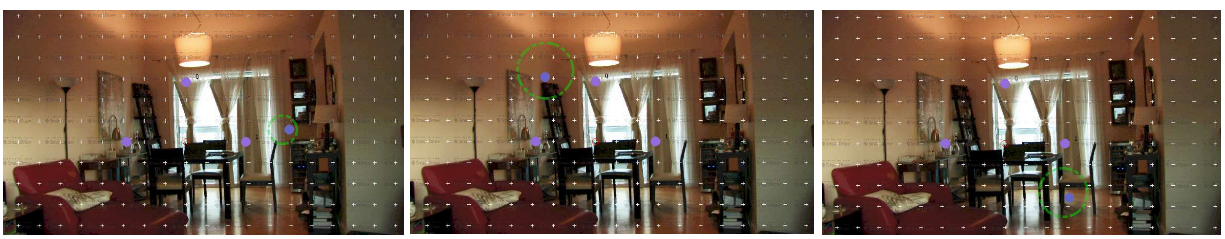


Figure 23: Screengrab of the bundled software with callouts of certain features, including the anchor placement-configuration, the localization plot and advanced settings to upload a floorplan

The functionality of the bundled software is to demonstrate the DecaWave system's positioning capability, as conveyed in the figures above; no further interaction regarding the tag is available for the user in the application. The goal of the MVP is to expand the capabilities to provide inputs for metadata regarding the tagged object, and display that data when an end-user engages with the virtual footprint of the object, as explained in the coming section.

Custom Software Design

This section captures the intended functionality of the to-be-developed software; it should be operational with the DecaWare hardware previously described. (Access to the hardware [anchors, tags, camera, etc] will be made available). The objective is to convey the minimal requirements to demonstrate feasibility, however future requirements are additionally codified in distinct insets. These future requirements are presented primarily to provide guidance so that considerations can be made regarding the current logic to ensure it will not impede or constrain the future needs. Additionally, this information is presented to reveal how these new features will create a more robust user experience and enhanced capability for the user.

Requirements

- The application, dubbed BeKnown.io, should be an executable program available for Windows and/or iOS.
- Upon loading, the application will allow users to create a new scene or load existing ones. A "scene" within BeKnown aligns to the traditional definition of a "scene" within the production realm, that is, a division of action consisting of a location, actors and props, that, when combined with other scenes, creates a logical sequence of storytelling.
- Figure 24 demonstrates this binary opportunity for the user to select an existing scene or create a new scene

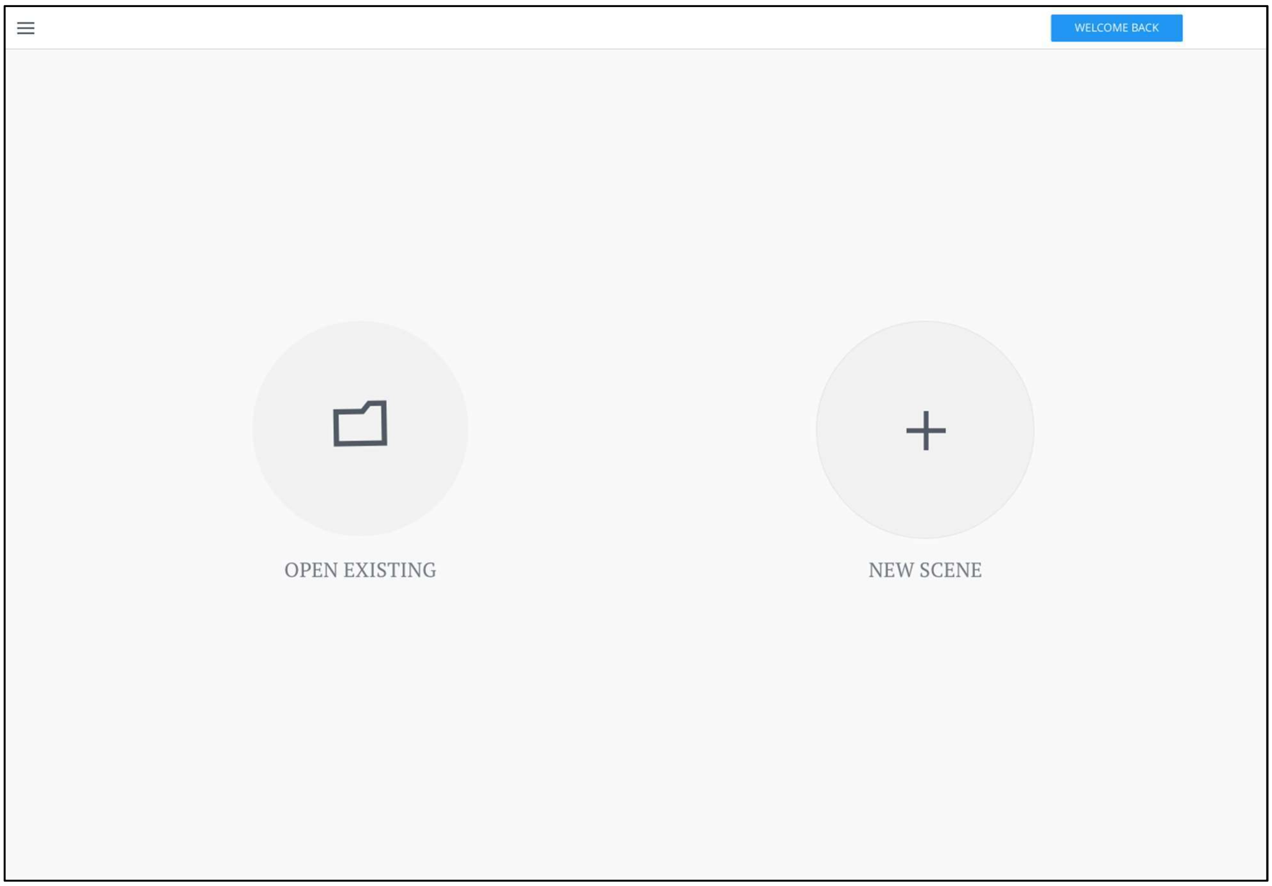


Figure 24

- Upon selecting "New Scene" the user is prompted with a dialog box per Figure 25; this dialog box allows the inputting of:
  - "Production" as a dropdown; the values include "Create a New Production" and all previously-created productions as shown in Figure 26. Productions are defined as a Movie, TV Show, Music Video or other complete story
  - "Act/Episode" as a dropdown (dependent on values in the "Production" dropdown; the values include "Create a New Act or Episode" and all existing acts/Episodes associated with the selected production. Acts/Episodes are defined as sub-units within a production.
  - "Scene name" as a text input. This is the more granular level within an Act/Episode and represents the division of content for which the tags will be associated
- "Tags" as a text input. This allows the user to enter keywords that make the scene easy to find during a search. It should be noted that the term "tags" in this instance is not akin to the term used to define the positioning tag

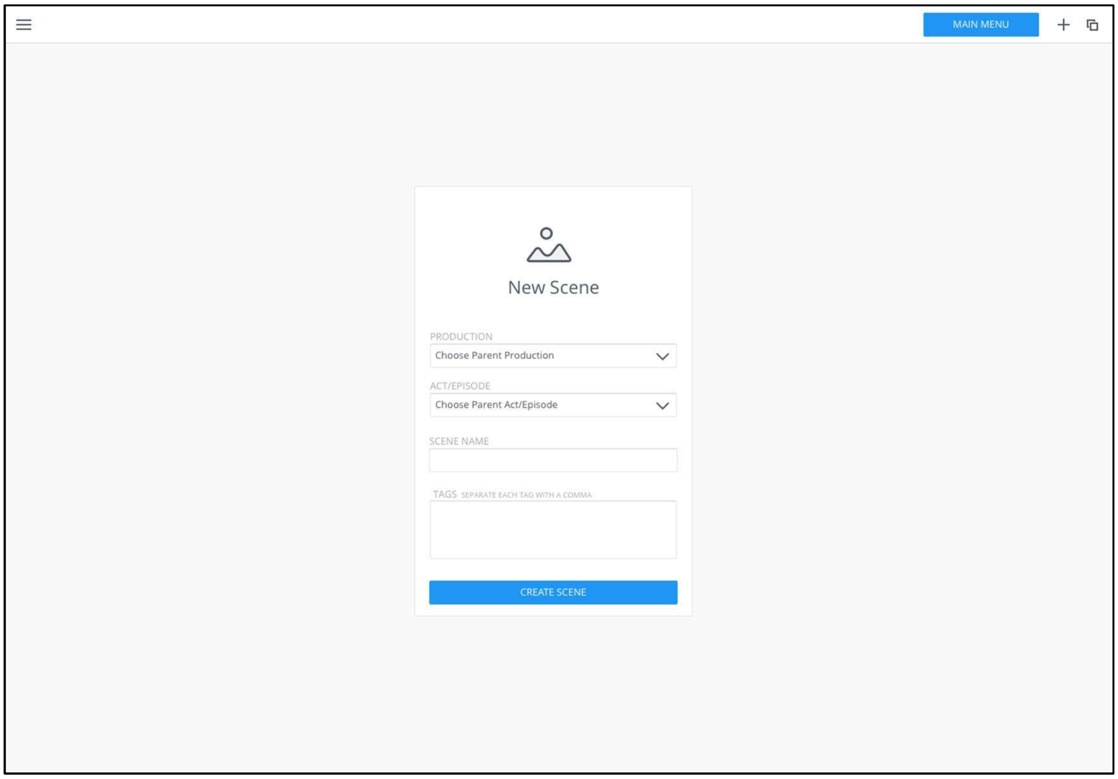


Figure 25

- Figure 26 showcases the dropdown menu, with "Create New Production" persisting at the top of the list

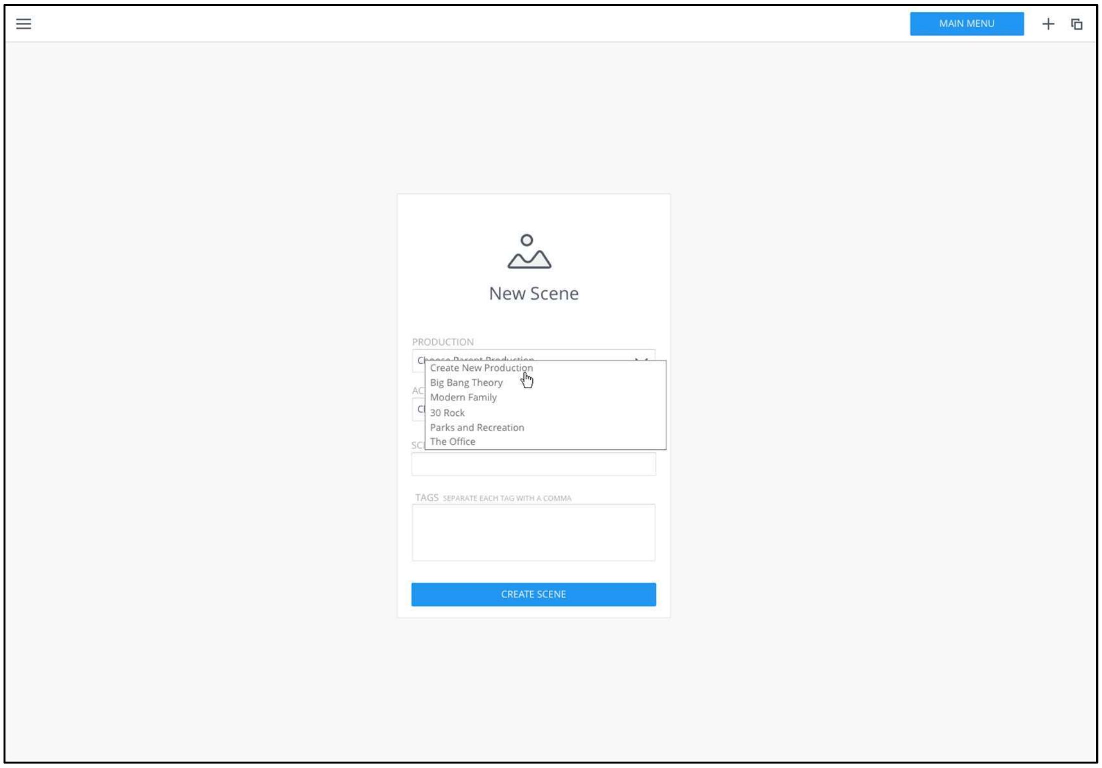


Figure 26

- When "Create New Production is selected", additional metadata fields are revealed, expanding the dialogue box. These fields include:
  - Director as a text input
  - Show Title as a text input
  - Season as a text input
- Similarly, when "New Episode/Act" is selected, additional metadata fields are revealed, expanding the dialogue box. These fields include:
  - Act/Episode title as a text input
- "Scene Name" and "Tags" persist as part of the original dialogue box

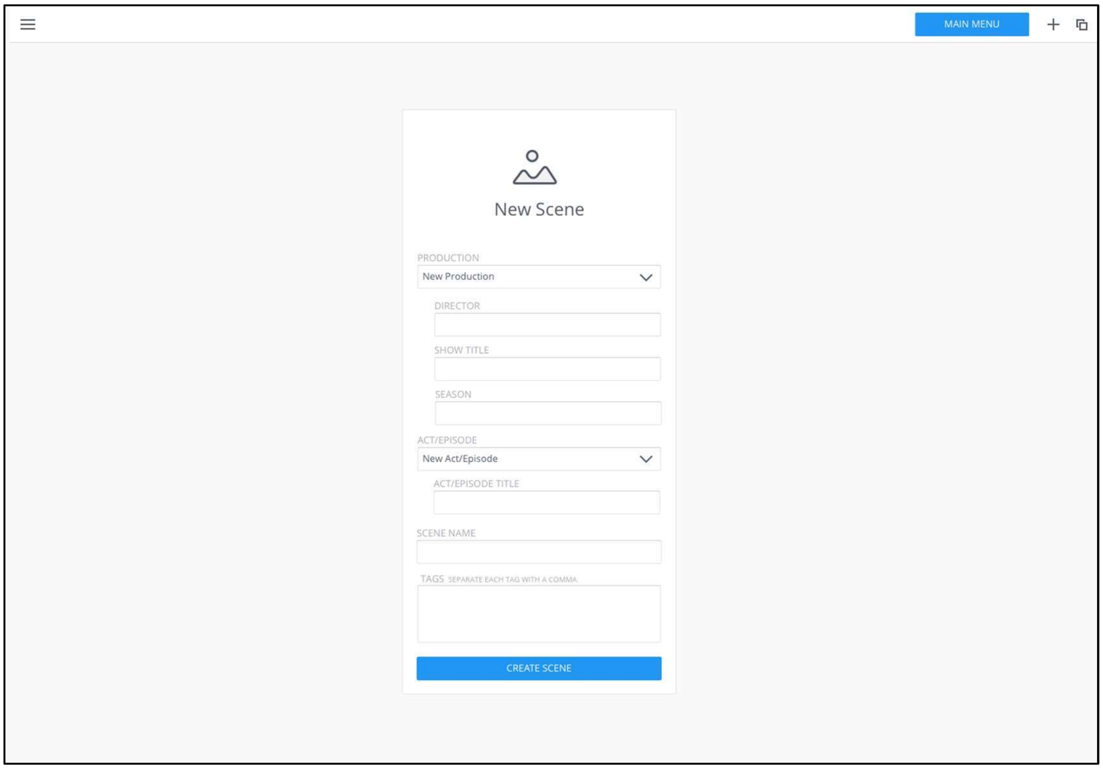


Figure 27

- Populating the inputs as illustrated in figure 28 creates metadata that will ultimately be exposed in the scene-details (figure 29, top banner) and in the scene explorer (figure 34)

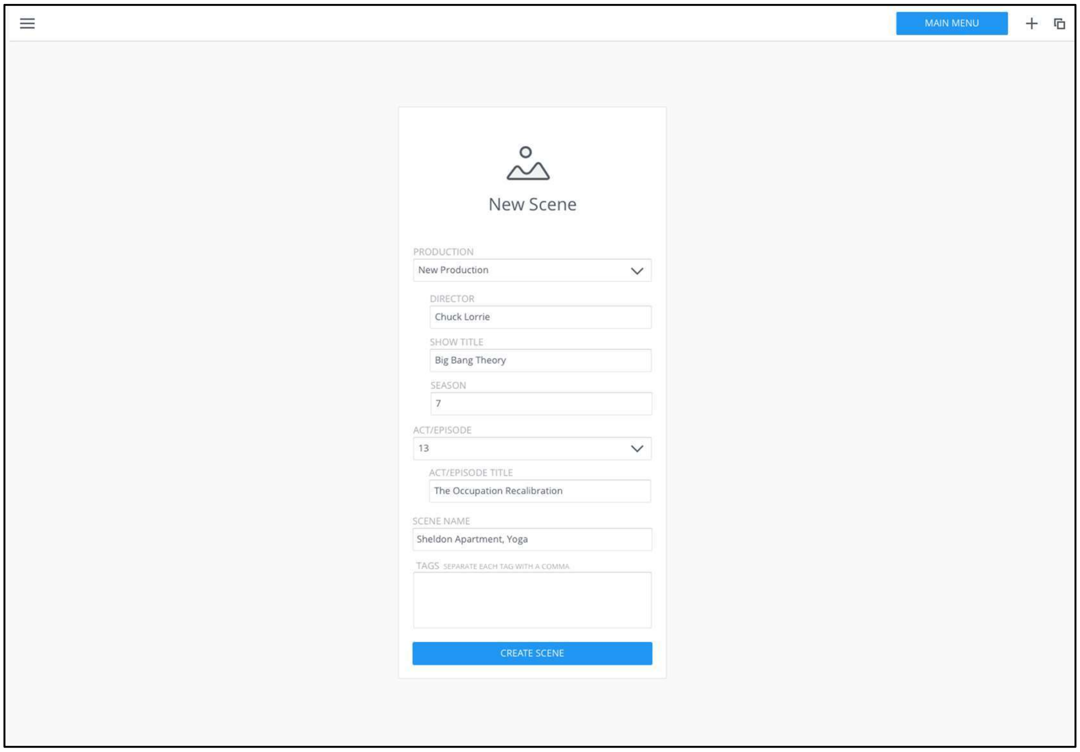


Figure 28

- Upon clicking the "Create Scene" button in the previous screen, the Scene Detail loads (figure 29); this is a multi-pane window consisting of:
  - Left pane that houses programmed tags
  - Top pane housing metadata
  - Main Pane: housing the live video stream and virtual elements which can be turned on and off using toggles in the bottom pane. In the MVP, only a live stream will be available to demonstrate object tracking (it is assumed processing/rendering of hotspots will occur in real-time computed by the attached computer). For the MVP, non-live playback/recording is not a requirement, but should be considered as a future-state capability
  - Bottom pane housing toggles
- Upon a new scene-creation
  - the left pane will be empty because no tags have yet been programmed
  - the "Show empty tags" toggle will be enabled by default revealing circular iconography in the main pane, corresponding to each triangulated tag

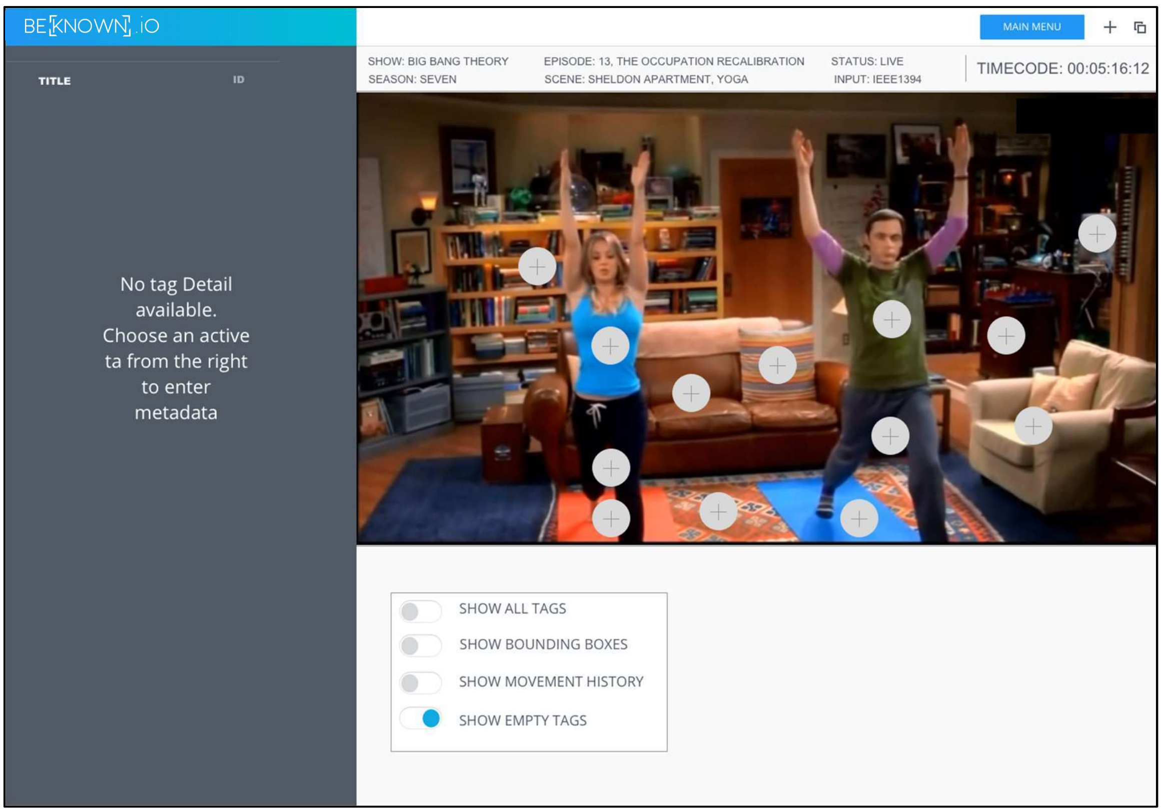


Figure 29

Future Capability

The majority of entertainment content produced is non-live content. Rather it is recorded, edited in post-processing, then distributed to networks or online. In order to accommodate, the future state software will be capable of writing the metadata to the frame's data layer (much like how closed-captioning data is embedded as packets). To facilitate ease of merging the video payload with the localization payload, precise timecodes can be passed from the camera to the computer/BeKnown Software, using the BNC audio cable which transmits data from the camera's genlock port to an audio-in on the computer [there are existing software solutions that perform this as seen here: https://www.videotoolshed.com/handcrafted-timecode-tools/ltc-midi-readerconverter)

- Clicking any one of the available tags will prompt a popup for the user to enter the meta data associated with that particular tag. Figure 30 showcases a dialog box within the context of the app, while figure 31 provides a more detailed view of the inputs

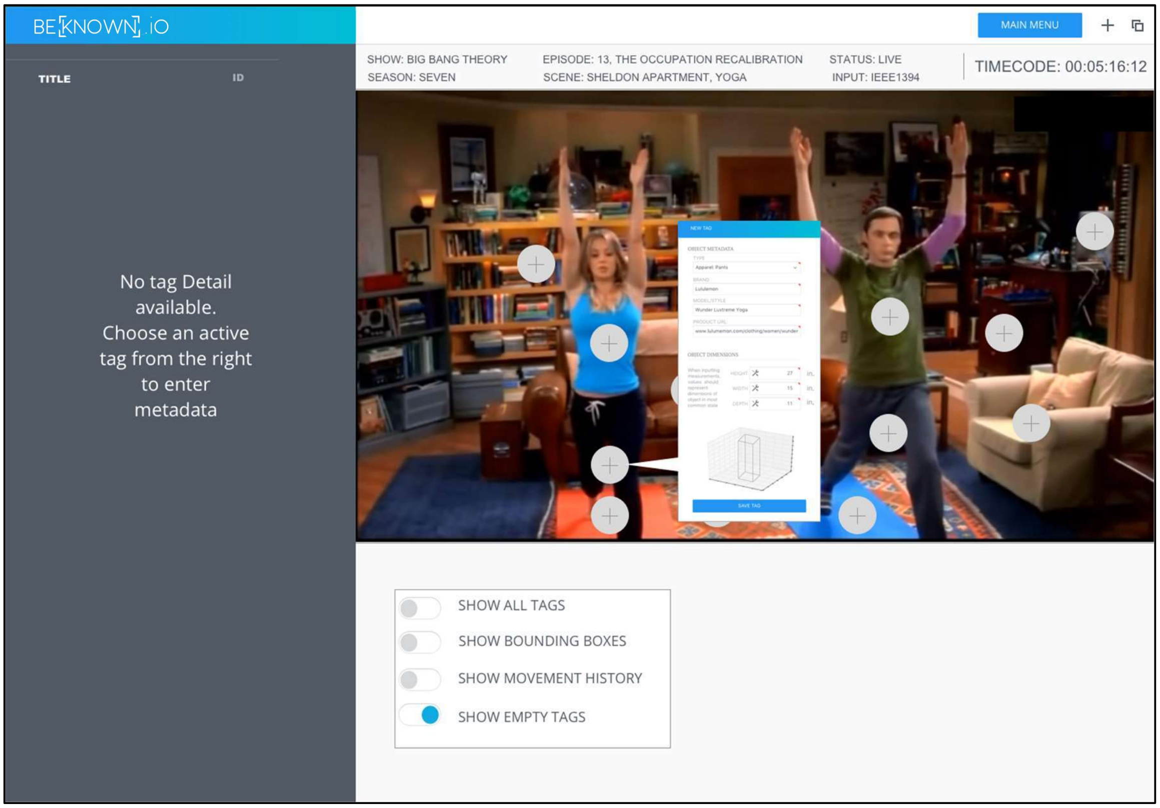


Figure 30

- The "New Tag" dialogue box includes the following required inputs:
    - "Type" as a dropdown: used for classifying the tagged objects into broad categories; initial values might include "Apparel: Top", "Apparel: Bottoms", "Athletic Equipment", "Furniture", "Jewelry & Accessories", "Electronics", "Appliances", "Home Décor/Accents".
    - "Brand" as text input
    - "Model/Style" as text input
    - "Product URL" as text input
    - Object Dimensions are used to construct bounding boxes around the object. For the MVP it will be assumed the tag is placed in the center of the object and thus the bounding box would be calculated from this center point.
    - The measurements are the actual real-world dimensions of the item. The grid demonstrates how the bounding box would appear in 3D space

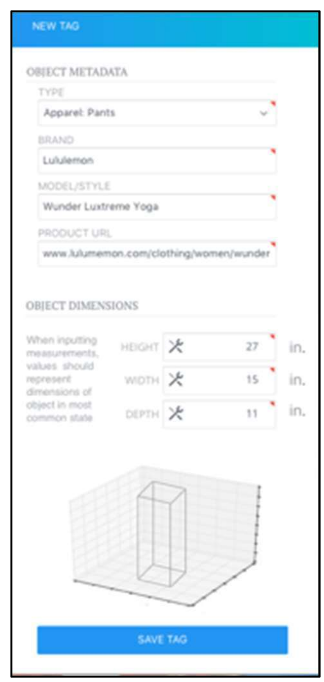


Figure 31

> Future Capability
>
> In future state the user will be prompted to position the tag within the 3D space in relation to the bounding box (for instance the user could indicate that the physical tag was placed at the top of an object, such as the collar of a shirt; the relationship between the bounding box and object will ensure the box is correctly aligned as the object rotates or changes orientation).

- Saving the tag will create a record in the left pane seen in Figure 32. Note no data is written to physical tag (ie transmitted to the tag and stored in memory), rather the software is creating a relationship between the unique tag ID and the metadata as records within a table in the backend of the software. This data is always available for use in other scenes. For example, if an other episode was filmed in the same location with the same props, the software will read these props/tags as known objects and not need re-tagged.
- The left pane also houses icons which allow the record to be edited (at which point the pre-filled dialogue box would be displayed again in the main window), or deleted.
- When the "Bounding Box" toggle is in the "on" position, the bounding box will show on-screen leveraging the dimensions inputted.
- The position and size of the bounding box are a function of the inputted dimensions, coupled with position in 3D space.

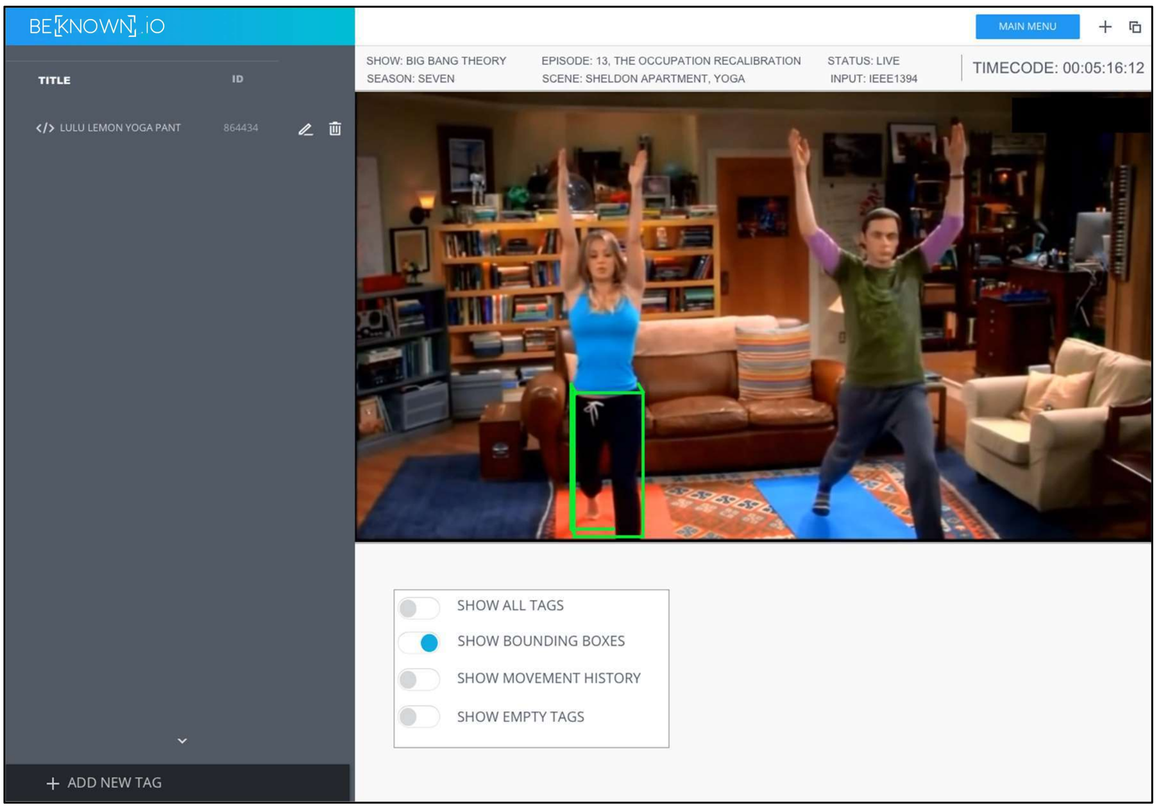


Figure 32

Future Capability

For v1.0 of the MVP, the X,Y axis captured by the location system is sufficient; this suggests the bounding box will be a static size, and assumes the objects move along a single plane rather than moving closer and further from the camera. In v2.0 a fourth anchor should be added to the hardware system. This will allow tracking along the Z axis; this measured distance from camera-to-tag would be used to determine the aspect ratio when rendering on-screen.

In software versions greater than 2.0 digital signals from the camera's zoom will also be an input to determine the appropriate size of the bounding box. This is necessary because the camera may not physically move, but the zoom creates this illusion. The signal created by controlling the zoom will need captured and processed.

Additional in the future state, the physical tag will include a gyroscope for orientation. This information will be transmitted to in the same packets used for localization. As the object rotates, pivots, etc, the data sent by the gyroscope should be processed such that the bounding boxes turn and rotate accordingly.

- Engaging the “Show Empty Tags” toggle will show tags in the main pane that have not yet had data inputted as demonstrated in Figure 33. (Note that the empty tag icon inside the bounding box is related to the yoga mat, not the pants which do have data associated with them)
- This configuration (“show bounding boxes” and “show empty tags” in the “on” position) allows the user to see both the accuracy of the related bounding boxes for inputted tags and those tags yet to be populated

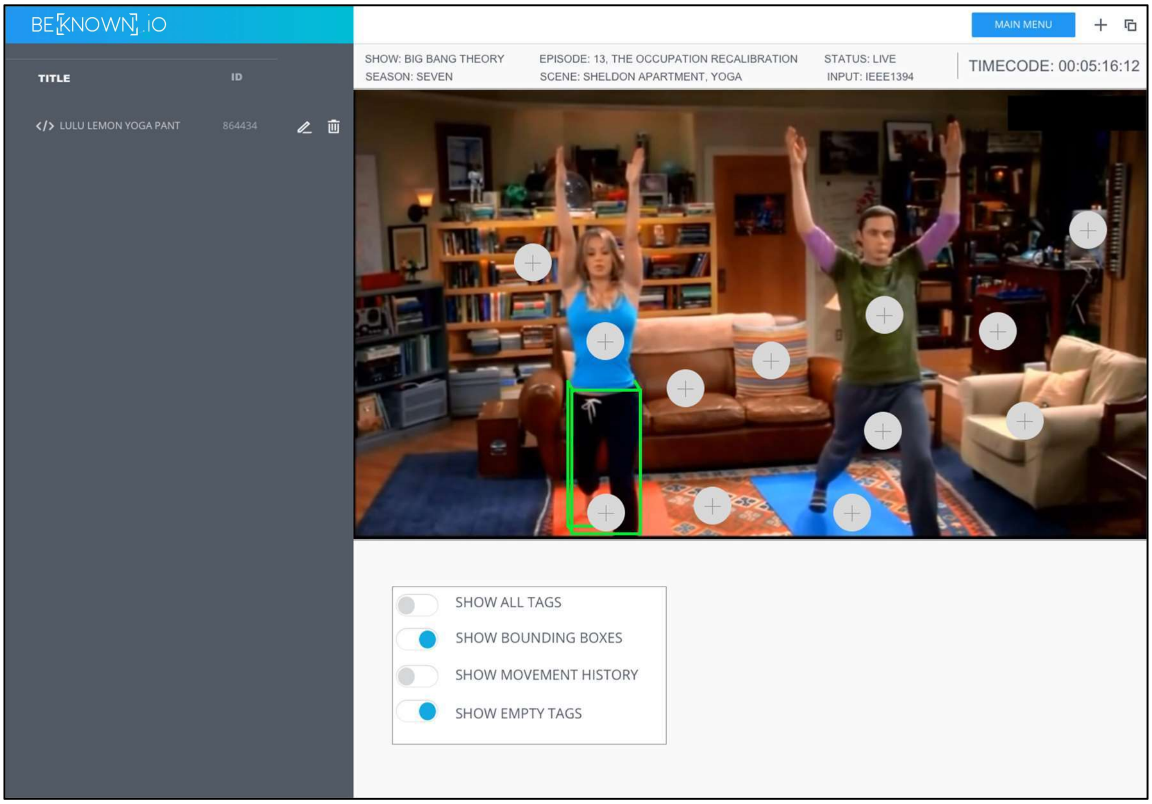


Figure 33

- The following set of screens represents the alternate path on-load of the application, where the user selected "Open Existing Scene"
- The Scene Explorer (Figure 34) reflects the asset hierarchy in the left pane. These are folders which are dynamically created when the user selects "New Production" and or "New Episode/Act" during the "New Scene" dialogue. Productions are the top level, followed by Episode or Act; the lowest level are scenes, which appear in the main pane of this window. Some metadata is available including Scene Name, count of tags (only those inputted, not those yet to be programmed), modified date, and iconography for additional actions
- The iconography includes
    - Pencil to edit the scene
    - Duplicate (this functionality is not needed and can be ignored)
    - Eye to go directly to full-screen mode (no edit functions; this mode simulates a consumer view)
    - "X" to delete the scene

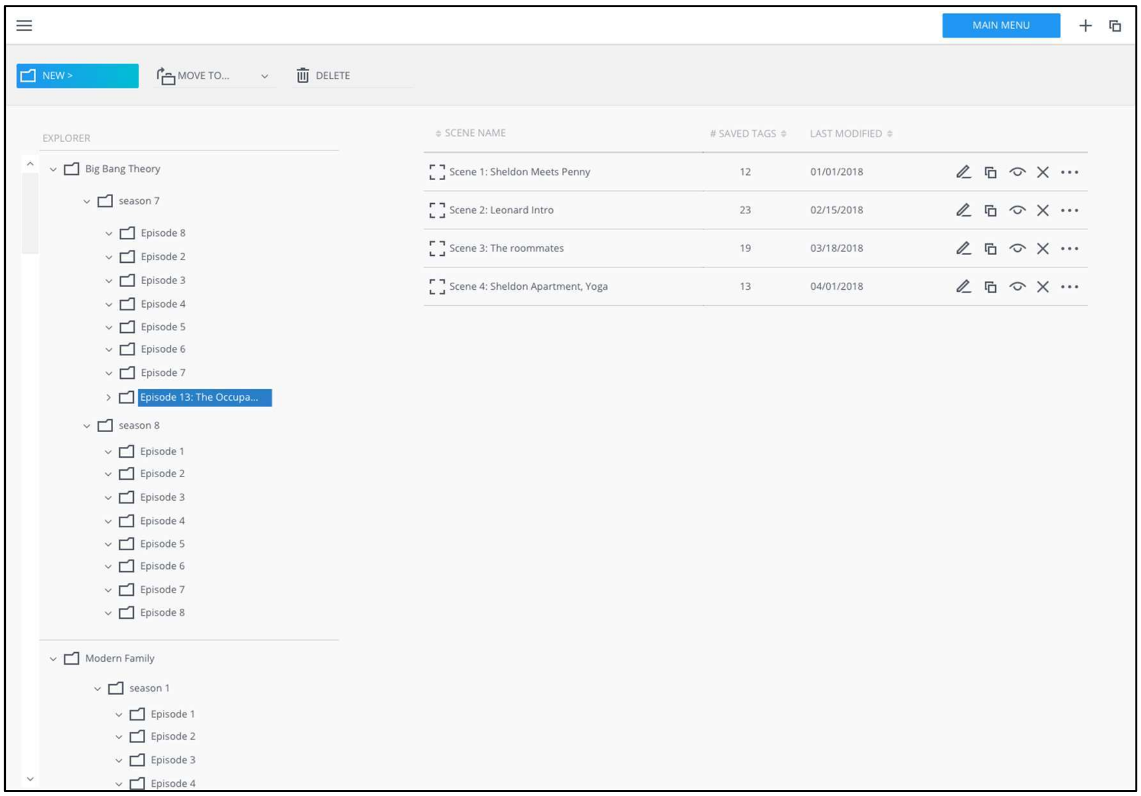


Figure 34

- Once a scene is selected (via double-clicking or selecting the pencil icon in the Scene Explorer), the "Scene Detail" page loads with all inputted tags reflected in the left pane. The middle pane showcases the scene with all toggles turned off, simulating the end-user experience as shown in figure 35.

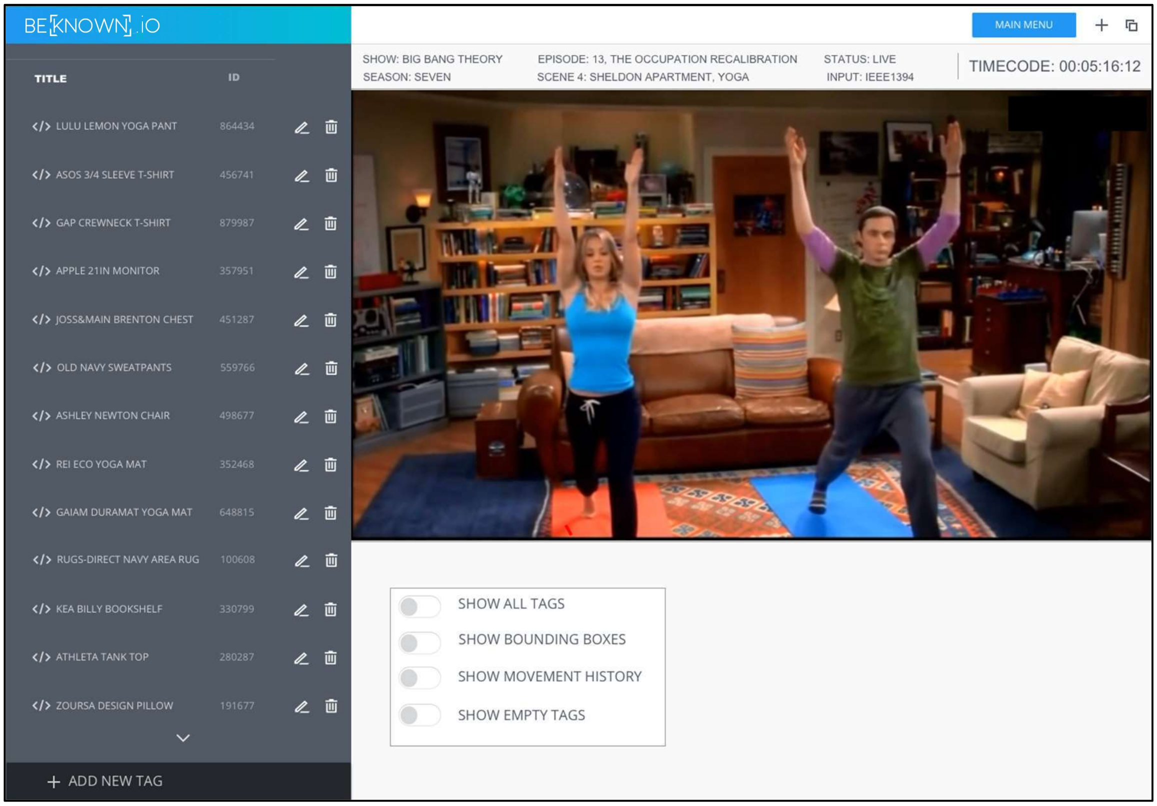


Figure 35

- When the "Show Bounding Box" is toggled on and "Show all Tags" is off then the bounding box will for a tag will only be shown when a particular tag is selected from the left pane.
- Figure 36 demonstrates "Athleta Tank Top" has been selected as denoted by the subtle layering (enlarged and shadowing effect); in-turn producing the bounding box in the main pane.

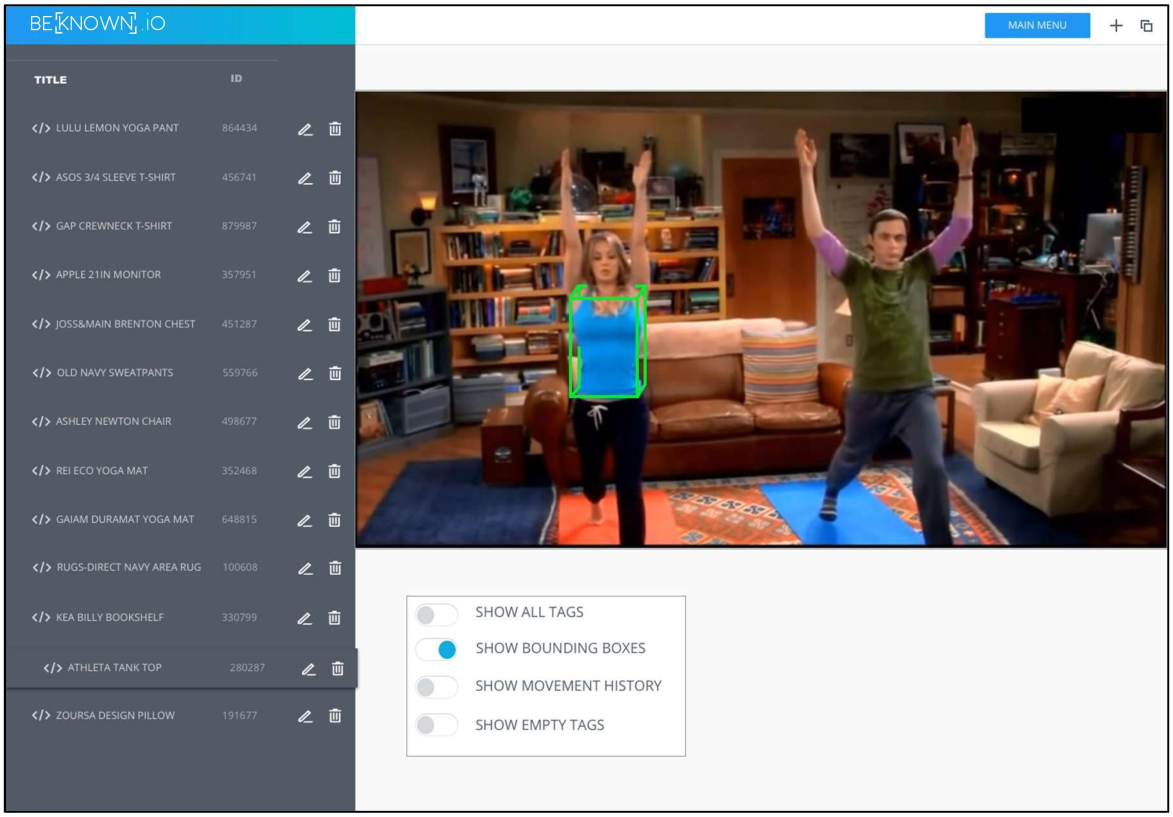


Figure 36

- A dialog box will render when the cursor hovers on over the footprint of the bounding box per figure 37.
- This is the user-centric dialogue box, rather than the editing dialogue box; it includes
  - A rendering of the webpage for the product (as a function of the original URL).
  - Brand (as inputted originally by the user)
  - Model (as inputted originally by the user)
  - Website (as inputted originally by the user).

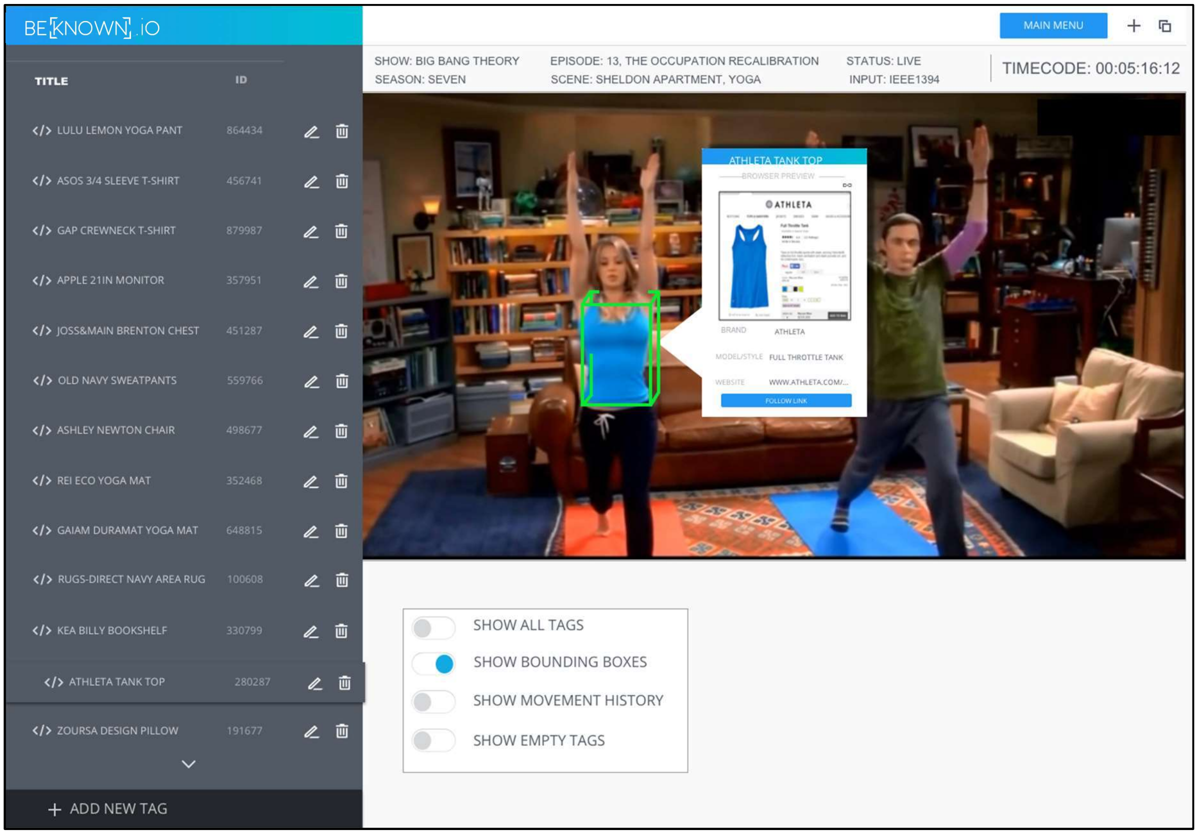


Figure 37

This concludes the section dedicated to the MVP v1.0 software development requirements. Proceeding sections will convey the future-state of the hardware.

Hardware Design

This section offers a cursory and visionary view of the to-be hardware. The actual development and feasibility require validation by those who specialize with hardware engineering.

The development kit provided by Decaware consists of four boards, 7cm x7cm (not including antenna). The design includes features required for testing and configuring the kit. The production version of the BeKnown hardware would be minituraized to remove the LCD readout and DIP switches used for toggling the board from “anchor” to “tag”.

Figure 38 shows back and front views of the board, with two components specifically highlighted (additional components are labeled in figure 16).

1. Decaware DW1000 chip
    a. IEEE802.15.4-2011 UWB compliant, Wireless Transceiver
    b. Allows the location of objects in Real Time Location Systems (RTLS) to a precision of 10cm indoors, even while moving at up to 5m/s
    c. Allows high data rate communications, up to 6.8Mb/s, in Wireless Sensor Networks
    d. Excellent communications range of up to 290m thanks to coherent receiver techniques
    e. Short packet durations support high tag densities – up to 11,000 in a 20m radius
    f. Highly immune to multipath fading
    g. Low power consumption allows operation from batteries for long periods
2. ARM STM32F105 microprocessor runs the module’s RTLS firmware and supervisory function. It is uses the Cortex-M3 core, with a maximum CPU speed of 72 MHz. The chip is intended for applications where connectivity and real-time performances are required

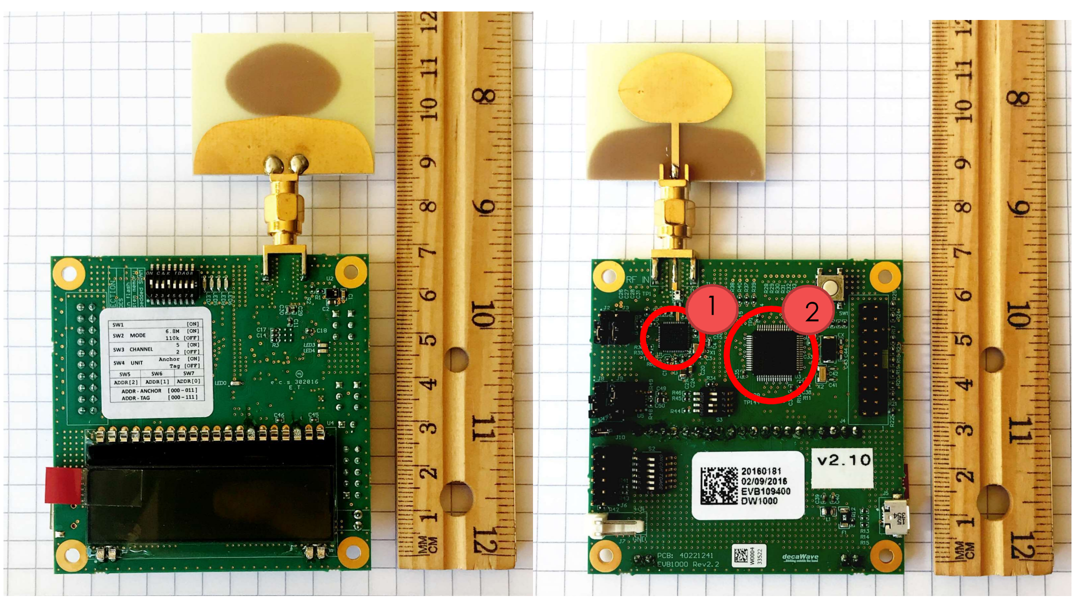


Figure 37

The dev boards require a power source; as currently configured in Figure 19, the three anchors are powered via USB cables to the computer, while the tag uses an external lithium battery. The footprint of that battery however makes the portability of the combined unit less-than-ideal. An alternative would be an integrated battery design in which a cell battery, such as the coin cell in figure 38, is designed into the circuit.

Figure 38: Coin cell battery

Additionally, the antenna bundled with the dev kit is inappropriate for the use cases described for the BeKnown system. Instead it is proposed that flexible, printed antennas be utilized. These antennas could be a variety of sizes and shapes to best adapt to the object being tagged. Examples of printed antenna designs are showcased in figure 39.

The final hardware design should integrate the battery, RF chip, microprocessor, and antenna into a form-factor no bigger than a key fob; this would allow it to be easily affixed to inside of clothing. For smaller items, passive tags, lacking a battery, might be required. These could interact with nearby active tags in a mesh network, however the scope of that feature is to be determined.

Figure 39: Examples of flexible, printed antennas used for RF applications