

# Riverwalk: Incorporating Historical Photographs in Public Outdoor Augmented Reality Experiences

Marco Cavallo\*  
Dept. of Computer Science  
University Of Illinois At Chicago

Geoffrey Alan Rhodes†  
Dept. of Visual Communication Design  
School of the Art Institute of Chicago

Angus Graeme Forbes‡  
Dept. of Computer Science  
University Of Illinois At Chicago

## ABSTRACT

This paper introduces a user-centered Augmented Reality (AR) approach for publishing 2D media archives as interactive content. We discuss the relevant technical considerations for developing an effective application for public outdoor AR experiences that leverage context-specific elements in a challenging, real-world environment. Specifically, we show how a classical marker-less approach can be combined with mobile sensors and geospatial information in order to apply our knowledge of the surroundings to the experience itself. Our contributions provide the enabling technology for *Chicago 0,0 Riverwalk*, a novel app-based AR experience that superimposes historical imagery onto matching views in downtown Chicago, Illinois along an open, pedestrian waterfront located on the bank of the Chicago River. Historical photographs of sites along the river are superimposed onto buildings, bridges, and other architectural features through image-based AR tracking, providing a striking experience of the city's history as rooted in extant locations along the river.

**Index Terms:** H.5.1 [Information interfaces and presentation (e.g. HCI)]: Multimedia Information Systems—Artificial, augmented and virtual realities

## 1 INTRODUCTION

The *Chicago 0,0 Riverwalk* AR experience provides a novel, interactive way for users to explore historical photographs sourced from museum archives. As users walk along the Chicago River, they are instructed to use their smartphone or tablet to view these photographs alongside their current views of the city. By matching location and view orientation the *Riverwalk* application creates an illusion of “then and now” co-extant. This superimposition of the photographer's and user's view provides a basis for educational engagement and is a key factor in curating the images and the narrative surrounding them, facilitating a meaningful museum experience in a public, outdoor context.

As such, creating the AR experience involves a complex back-and-forth between 3D and 2D experiences of locations. The historical images are 2D, taken from specific locations through specific optics and views. The user, situated at a real-world 3D location, orients their phone's camera in space in order to discover the historical images. The AR experience enables the user to see two integrated views simultaneously: the stream of image data taken from a smartphone's camera and the historic image. The site-specific nature of the project makes it necessary to utilize a virtual 3D environment in which to place 2D augmented content that enhances the experience. This content needs to be placed in such a way so that the designer

\*e-mail: marco.cavallo@mastercava.it, web: <http://mastercava.it/>

†e-mail: garhodes@garhodes.com, web: <http://chicago00.org/>

‡e-mail: aforbes@uic.edu, web: <http://creativecoding.evl.uic.edu/>



Figure 1: *Chicago 0,0 Riverwalk* is a mobile augmented reality application that presents 2D media content from archives of historical events.

of the experience can accurately visualize what the user will see on screen from a particular location.

The first episode of the *Riverwalk* AR experience focuses on a single block between N. LaSalle and Clark Streets; the site of the Eastland Disaster in 1915 [14]. The site was selected because of the importance of this historical event—the sinking of the Eastland cruise ship 100 years ago was the largest single loss of life in Chicago's history—and because of the abundant media available in the archive, which includes newspapers, film reels, extensive photographic documentation. Moreover, the site potentially offers natural wayfinding characteristics that are amenable to developing our project, including a newly built pedestrian walkway along the river with viewing platforms, views from nearby bridges, and historical markers. Additionally, the urban environment offers many features that provide sufficient quality for robustly tracking user location from a variety of views and positions.

The Reid, Murdoch & Company building on the north side of the Chicago River is a historic building that is also featured in the archival photographs of the Eastland Disaster. It is the only major architectural site that is available from all vantage points along the Riverwalk. However, there are also smaller tracking-adequate features along the site (such as signage and buildings in the distance) that can be used for tracking from a more limited number of views. Inversely, the archival photography was captured from a variety of angles: from the riverside, from the bridges, and from boats on the river. In order to match this rich, historical imagery to the user's current orientation, a form of extended tracking is necessary. This tracking ability is used in the *Riverwalk* experience as well as the in-app user guidance in which users are directed to obtain and calibrate tracking by pointing their cameras at the Reid, Murdoch & Company building.

Tracking is extended beyond views of this one building through the definition of a *virtual camera* defined by the sensor data from the user's phone. Though existing platforms include extended

tracking behavior, our goal is to create a more robust extended tracking technique that can be used in an “always on” mode suitable for public outdoor AR situation. In addition, because of its real-world application for a specific site and experience, we want to enable customization of the tracking behavior based on known variables and constraints for a specific site.

Below we present our approach to developing a mobile augmented reality system that incorporates 2D historical photographs, analyzing how domain-specific considerations can be used to improve both the tracking and the user experience. This same technology also enables our custom authoring system for creating this type of AR experience, which can be used to design other similar types of public outdoor projects.

## 2 RELATED WORK

Available public platforms for image-based AR— such as Layar [10], DAQRI [7], Aurasma [3], Vuforia [13], and AR-Toolkit [2]— focus on robust image tracking for 2D features, have been used effectively in print publishing and advertising campaigns that incorporate AR components. Research in natural feature detection for 3D space, and particularly architectural features, to date has largely focused on surveillance and military applications [9]. Museums, city arts councils, and tourist boards are interested in creating public AR exhibitions within the urban landscape, but there does not yet exist a platform optimized for the challenges presented by public outdoor contexts in urban environments.

Some previous works utilize location-based augmented reality with the support of mobile sensors (such as the Andy Warhol Museum’s *Geo Layer* and the Museum of London’s *Street Museum*, both of which highlight geolocated media archives within the urban community). However, for the *Riverwalk* project, the need for accuracy prevents us from relying solely on GPS and mobile sensors for positioning content. The desired illusion is created through matching and alignment— a rough, “floaty” approximation would defeat this goal. In fact, while it has satisfactory accuracy in open spaces, its performance deteriorates significantly in urban environments [11], since shadows from buildings and signal reflections greatly reduce its availability. At the same time, inertial sensors are often prone to drift and local magnetic fields encountered in urban environments may disturb magnetic sensors on mobile devices. Other approaches rely instead only on markerless augmented reality techniques. For instance, *Tidy City* [15] and *City-Wide Gaming Framework* [17] generate urban scavenger hunts by leveraging existing platforms to detect image patterns corresponding to the facades of the desired buildings. However, this general-purpose approach has drawbacks related to the recognition of 3D landscape features from multiple viewpoints, since the recognition of features that are changeable by lighting, weather conditions, and man-made interventions. Additionally, many architectural features include flat surfaces, repetitive patterns, and shiny materials, properties that require more sophisticated detection algorithms.

In order to enable accurate, real-time overlays for a handheld device in urban environments, other works have tried to combine multiple approaches. Art et al. [1], for instance, propose a method for estimating the 3D pose for the camera using untextured 2D+height maps of the environment, leveraging image processing to refine a first estimate of the pose provided by the device’s sensors. Côté et al. [5] note that augmented reality fails to provide the level of accuracy and robustness required for engineering and construction purposes and present a live mobile augmentation method based on panoramic video. In their system the user manually aligns the 3D model of the environment with the panoramic stream, thus avoiding sensor calibration issues. Takacs et al. [12] use an adaptive image retrieval algorithm based on robust local descriptors in order to create a database of highly relevant features that is continuously updated over time in order to reflect changes in the environment



Figure 2: The above image shows a group of people testing the *Riverwalk* AR application with both smartphone and tablet devices along the Chicago River.

and to prune outlier features. Their system relies on geo-tagged data collected from several locations, by several people, at several times of the year and day. The tracking method proposed by Reitmayr and Drummond [11] also combines several well-known approaches, providing an edge-based tracker for accurate localization, gyroscope measurements to mitigate problems arising from fast motion, measurements of gravity and magnetic field to avoid drift, and automatic re-initialization after dynamic occlusions or failures. However, it does not take advantage of geolocated content and does not address the way in which this content is provided to users in real-world contexts.

Our approach enhances tracking characteristics by leveraging real-world locations of content and fiducials in combination with mobile sensors, taking into account site-specific considerations in order to address the challenges of natural feature detection within an urban landscape.

## 3 ENABLING PUBLIC OUTDOOR AR EXPERIENCES

Our work leverages natural feature detection and tracking, but makes use of a hybrid location-based system that utilizes both mobile sensors and the real positions of objects in space. This is motivated not only by the need of extended tracking when a detected pattern is lost, but also from the need for showing augmented content in places where there are few tracking features available, as well as by the possibility of creating new interactive ways to guide the user during the AR experience [4].

Our first goal is to apply an effective use of sensor-based data in order to stabilize the augmented content in the AR overlay illusions. Instead of using simple frame-by-frame live tracking, we interpret tracking data based on typical use scenarios in which the user is more concerned about an image appearing substantial, stable, and visible than they are about exact updating of position every frame.

Additionally, by intelligently calibrating the two methods of sensing position and orientation (i.e., image tracking based and sensor based), our technique minimizes the influence of false positives— the occurrence of brief moments of image recognition that are incorrect in the real world— and tries to use, in a customizable way, our knowledge about the real world. For example, knowing that a recognized building is of a certain size and orientation could allow us to infer that tracking any moments in contradiction to this knowledge are false.

Because we are using an “always on” system in which the sensor-based camera is continually estimating the user’s position and orientation, it is important that we develop a system for turning the site augments on and off based on the users location and orienta-

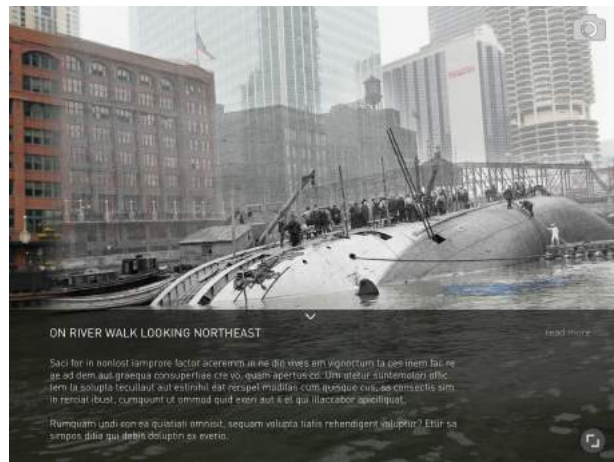


Figure 3: Above are shown two example photos from the historical archive superimposed on a phone’s live camera stream. In order to create an appealing augmented reality effect, the two photos need to appear aligned with specific environmental features and need to be seen by the user from a particular point of view. On the right, the half-sunken ship, the Eastland, can be seen placed accurately in the river—the exact location where it sank 100 years ago.

tion. This can be done through a simple mapping of active AR locations, combined with available information about the specific site and desired use scenarios.

We can define two main types of elements in our approach, *virtual content* and *fiducials*. Virtual content consists of 2D historical photos that will be rendered on top of the live camera stream from the mobile phone, augmenting what the user is able to see. Fiducials (or trackables) consist of specific pattern images whose features can be detected in the real world and are used for estimating the pose of the camera. Though our approach can be potentially be applied to any type of tracking image (e.g. traditional binary markers), here we look only at natural feature tracking as our interest is mostly directed towards outdoor, architectural features.

In most classical markerless AR approaches, once a set of predefined planar features is detected, a 2D or 3D virtual object is then rendered on screen by computing the pose of the camera relative to those features. Our method instead computes the absolute position and rotation of the camera in world coordinates, that is, not only in relation to a single tracked object. In particular, using map projections we define a one-to-one mapping from virtual coordinates to WGS84 real-world coordinates and vice-versa, so that for each  $[x, y, z]$  position in virtual space we have one  $[latitude, longitude, altitude]$  triplet with real-world coordinates. A specific WGS84 location (the position of the user retrieved through their phone’s GPS) is initially associated to the origin of the virtual world. Successive locations are calculated in relation to that original position, keeping a scale of 1 unit per 1 meter.

Both virtual objects and fiducials are therefore characterized by real world position, orientation, and scale. (In our case, scale is characterized only by height and width, since we are working with historical photos). This additional information provides us with runtime information about the spatial position of the AR content that will be presented to the user in order to know where one should expect to find fiducials, and ultimately to define complex relationships between the different virtual objects.

#### 4 THE DUAL-CAMERA APPROACH

Our approach aims to determine the absolute pose of the camera, possibly independent from tracking features in the environment. On top of the live camera stream we render the output of the main virtual camera, which is characterized by a position and a rotation in 3D space and whose field of view matches the one of the mobile

device. This camera moves and rotates in the virtual space as the user walks in the real world, creating a sort of “parallel world” that coexists and overlaps with the normal one, enabling the augmented reality effect.

The absolute pose of the camera is computed by weighting the parameters of two helper cameras (hence, “dual-camera”), defined in the virtual world as follows:

- the *ARCamera*, whose pose is computed from the tracking of predefined features found in the live mobile camera stream;
- the *SensorCamera*, whose position and rotation in space are defined only through the sensors available on the mobile device.

We will refer to the hardware mobile camera as the *MobileCamera* and to the virtual camera that renders the final augmentation as the *MainCamera*, whose position and orientation in space will be computed by weighting the parameters of the two helper cameras.

In the case of the *ARCamera*, we use a common markerless approach to estimate its pose when a trackable is detected through the *MobileCamera*. Although our method is also appropriate for other kinds of pattern-based tracking algorithms, our current implementation is based on the ORB descriptor-extraction algorithm, performing several steps in order to match the feature points of the user-provided pattern with the live camera input (such as ratio tests and a warping step in order to refine the homography estimation). In addition to these common computations, we take into account the position, orientation, and dimension in the real space of the image provided as a fiducial, thus allowing absolute positioning of the *ARCamera*. So, differently from previous methods, we do not simply know the pose in the camera in relation to a single fiducial, but have a global pose of the camera in the real world, enabling interaction with the user or other virtual content. Additionally, the set of fiducials being sought for tracking can be limited by GPS location to those known to be available in that area, thus saving significant computation and allowing different forms of dynamic resource management. The main limitation of the *ARCamera* is that it is enabled only in presence of a tracked pattern.

The *SensorCamera* does not rely on any computer vision algorithm, but uses instead the data provided by the internal sensors of the mobile device. In particular, the compass, accelerometer, gyroscope, and A-GPS information is combined in order to retrieve

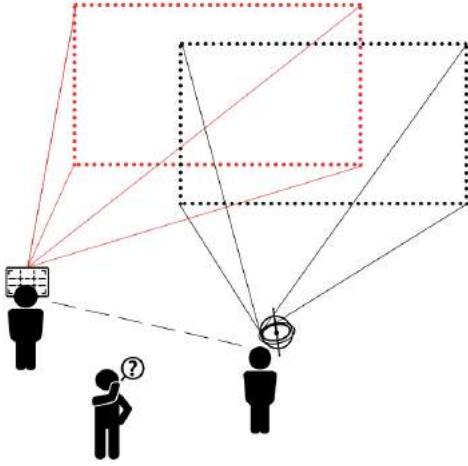


Figure 4: Our goal is to estimate the absolute position and orientation of the camera of the mobile device in space. In order to achieve this, we combine information from two helper (virtual) cameras: the ARCamera (left), based on image tracking, and the SensorCamera (right), which relies on mobile sensors. Optimally the two cameras would have the same pose, but in real-world situations we need to intelligently weight their contribution to the final pose estimation based on the current context.

both position and orientation in absolute coordinates. Most current smartphone devices have their own native sensor fusion algorithms that can compute orientation in the coordinate system of the device, mostly by leveraging the gravity vector, the geomagnetic field, and the rotational acceleration (whose drift is sometimes already natively corrected). Regarding instead the position of the camera, we convert the position retrieved from the GPS to virtual units. Though our current implementation does not comprehend sensor fusion techniques to smooth GPS information in combination with accelerometer and gyroscope, any technique could be applied on top of our method (e.g., visual odometry [8], step detectors [16], or multisensor odometry [6]) to improve horizontal positioning. An intrinsic limitation of the SensorCamera is that it loses one degree of freedom on the vertical axis and needs to be set to approximately the height of the user since GPS vertical accuracy is very low (notice that our system uses height relative to the ground, not relative to sea level). Just as the ARCamera may be disabled when no pattern is detected, the SensorCamera loses horizontal positioning if the GPS becomes unavailable (e.g., indoors) or can have an incorrect heading if magnetic fields influence measurements. Optimally, the two helper cameras would be completely overlapped, having the same position and rotation. Unfortunately, this is not so common in real life experiments and in the next paragraphs we describe our approach to exploit the information available in order to define the MainCamera that renders the augmented scene.

#### 4.1 Final Pose Estimation

As represented in Fig. 4, to combine the information provided by the two helper cameras (ARCamera and SensorCamera) we need to implement a dynamic way to intelligently estimate the final pose of the MainCamera. Our algorithm takes into account specific situations where one of the two cameras is preferred over the other or where both cameras are merged in order to verify their consistency. By remembering that the SensorCamera is always enabled, while the ARCamera activates only in presence of a pattern to be tracked, we can define the following four primary situations: *fiducial found*, *fiducial lost*, *multiple fiducials*, and *no fiducials*.

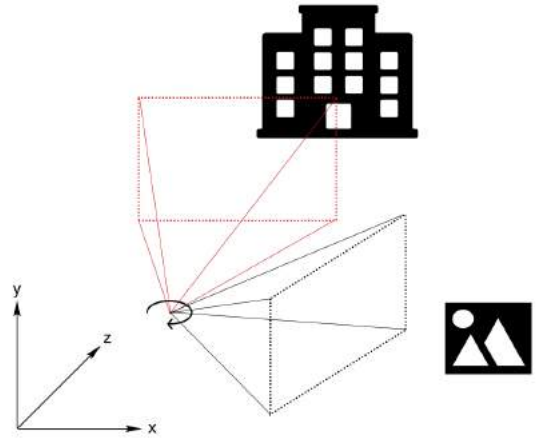


Figure 5: The  $\Delta r$  matrix can be imagined as the difference between the rotation of the two helper cameras. In the example above, the user rotates the device to his right and loses the tracking of a building. When the ARCamera becomes inactive, we can leverage the SensorCamera and the  $\Delta r$  matrix to know how much the user has rotated from the direction in which tracking was lost. This way, considering the absolute position of elements, we can still render virtual content nearby to the user even if no tracking information is available. At the same time, we can evaluate the coherence of what we are rendering by analyzing both how the two helper cameras differ from each other and how they are located and oriented in 3D space.

**Fiducial Found** When the input video stream matches the pattern of one fiducial, the ARCamera activates and assumes a position in space calculated considering the coordinates and size of that trackable, which, without considering the intrinsic accuracy of the tracking algorithm, may represent the greatest source of accuracy. If we assume those properties are correctly set by the creator of the application, the ARCamera pose can be considered in this case more reliable than the one provided by the SensorCamera. After a few frames in which the ARCamera stabilizes its position, we store the difference in orientation between the two cameras as  $\Delta r = r_{ar}^{-1} * r_s$ , where  $r_{ar}$  and  $r_s$  are respectively the rotation in space of the ARCamera and of the SensorCamera. Under the assumptions above, the MainCamera pose is set to be the same of the ARCamera, which in our tests proved to be more reliable than the SensorCamera as the sensors could be affected by magnetic fields and GPS inaccuracies. Tracking errors can be found simply by splitting the  $\Delta r$  matrix over the three axis and considering how much the ARCamera rotation differs from the SensorCamera (i.e. if the former is looking forward and the latter is facing down, probably there has been a false positive in tracking and the augmented content should not be rendered). Our algorithm enables flags for each object in order to set its behaviors in case of known environmental issues in that area.

**Fiducial lost** When a fiducial is lost, the ARCamera is disabled and we need to rely on the SensorCamera for extended tracking. As far as rotation, the  $\Delta r$  matrix is used for maintaining the correct orientation of the main camera even if the tracking is lost or the two helper cameras are not correctly aligned (especially in case of magnetic interferences): the new main camera rotation is simply computed as  $r = r_s * \Delta r$ , guaranteeing a smooth continuation of the movement performed by the user. By filtering out sudden rotations around the global vertical axis, we can prove this extended tracking solution is not affected by compass calibration problems. It is very useful to leverage this situation in cases in which we have few good trackables that are not in the direction where we want our augmented content to appear. For instance, let's consider the case when

the user is in a poorly textured area where the only reasonable trackable is a building, which is 90 degrees left of the object we want to show. In this case we can direct the user to look at the building, correct his or her position with the ARCamera pose computed from the trackable and then make him or her turn right to the virtual object. The user could lose tracking, but by relying on the SensorCamera the algorithm will allow the object to be seen anyway, taking into consideration the delta rotation the user has performed from when he lost the tracking, as we can observe in Fig. 5.

**Multiple Fiducials** Our algorithm supports tracking multiple markers at the same time. The ARCamera pose is computed from the first pattern detected until its tracking degrades (or the trackable goes out of screen), then it is automatically switched to the other available ones using a smoothing function. This is particularly useful for keeping an accurate and smooth camera pose. It is also fundamental for indoor pose estimation, since in that case the GPS signal would not be available.

**No Fiducials Available** When no fiducials are available, the algorithm may behave differently based on different settings. In some cases, we would like augmented content to be precisely positioned only relative to a fiducial and not be affected by potential sensor inaccuracy. According to this situation, the algorithm does not show virtual elements if no trackable is used for pose correction. On the other hand, we could simply make the MainCamera correspond to the SensorCamera when no trackables are available, meaning that the user location will be used for showing nearby virtual elements.

## 5 USER EXPERIENCE OF THE RIVERWALK APPLICATION

By leveraging the real-world, absolute pose estimation provided by our method, we are able to obtain many advantages that enable us to offer a better experience to the user. The presence of multiple tracking sources provides richer data about the environment, that we can then constrain selectively thanks to our own knowledge of the site and the expected behavior of the users. Our user-centered approach to AR leverages all available and site-specific information to create a more intuitive user experience. This is reflected also in the graphical interface with which the user interacts during the AR experience, and which relies on the new possibilities offered by our approach.

For the *Riverwalk* project, we opted for a minimal user interface in order to dedicate more screen space to the AR experience itself and to make the user feel more immersed with the content. With a simple swipe, a slider containing information about nearby virtual content may be activated on the right side of the screen: points of interest are color-coded and grouped based on their location or historical relationship, allowing the user to explore all related content before moving to a different area. The sequencing of the content mirrors the linear movement along the river. When the user selects a point of interest, a pop-up shows where he or she has to aim his or her mobile camera in order to enable tracking, as shown in Fig. 6. The AR experience relies on directing the user to a location and view orientation that matches one defined by a photographer decades ago; the application UI seeks to communicate this positioning, and the number and breadth of locations available, in a clearly visible way. The user can toggle on and off a “minimap” that is located below the slider in order to highlight areas of interest, respecting the color-coding convention mentioned above.

When the user has reached a specific area or has focused his or her camera on a fiducial, different types of interaction may be enabled. For instance, the fiducial itself could be a piece of architecture onto which an overlay with related historical imagery is superimposed. Narrative textual content is added to the experience through annotations, historical descriptions, or pre-recorded audio. Alternatively, directing the user to aim at a fiducial could even have the purpose of tracking his position with a higher accuracy in order

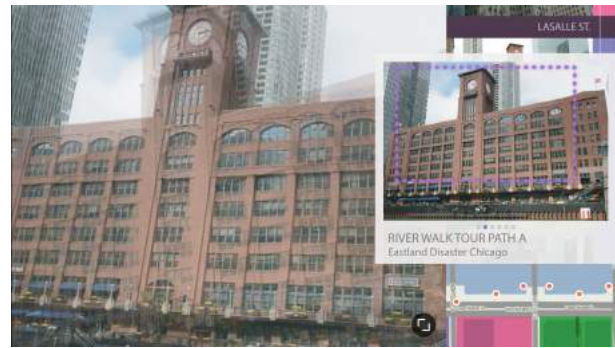


Figure 6: Our approach involves the use of a smart, intuitive interface that guides the user so that he or she will have a better experience discovering historical images. In the picture above, a pop-up indicates where the user should aim his or her device in order to activate augmented content.

to show the user content that is not necessarily in the direction in which he or she is looking. For example, we can make our users look at the facade of the Reid, Murdoch & Company building because of its robust tracking, and then rely on relative rotations in order to display overlays 90 degrees to the left, where the bridge and the skyscrapers in the background would not allow an acceptable tracking. In these situations, an in-app UI directs the user to this view. We rely on narrative audio, visual and textual annotations to guide the user towards the desired content, indicating, for instance, that the user should turn the device until the desired orientation is reached. By leveraging the absolute rotation in space of the device, we can also know when the user suddenly puts down his or her mobile device and as a consequence we can dismiss these indications. If an incoherency from the mobile sensors is detected, instructions are generated that guide the user back toward the fiducial in order to re-establish a robust tracking. In particular cases of sensor unreliability (e.g. in presence of relatively strong geomagnetic fields), some suggestions will be displayed on screen explaining how the user can perform a calibration procedure by moving the device.

When multiple content is closely available from a particular point of view, an orientation threshold is used for determining how to update annotations when a user is moving his attention from the previous overlay to the next one. In particular, there are many cases in which two or more overlays may appear overlapped on top of each other. This is dealt with by considering the angle at which the contents overlap and showing a colored dot that, when pressed, activates a transition between one overlay to an adjacent one, changing their opacity accordingly. A similar and very common case happens when a specific point of view has multiple overlays that need to be displayed in the same position but that completely obscure each other. Here we decided to allow the user to see one of them at a time, displaying the availability of multiple content and creating a transition to the next overlay when the user touches the screen.

Additionally, we consider a feature that allows users to themselves correct their position or the camera orientation when there is poor tracking or misaligned content. Explicitly asking the user to manually improve tracking is an interesting innovation for AR applications geared for the general public. We plan to study this behavior more thoroughly in the future.

We also consider it important to include a static user interface that lets the user visualize the augmented content if poor environmental conditions do not allow any tracking for a realistic AR experience. Weather conditions like fog or rain or simply the absence of light may cause the tracking not to work properly or the overlays to appear inconsistent with the real scene. Additionally, non-AR based methods to display the content created for the experience—

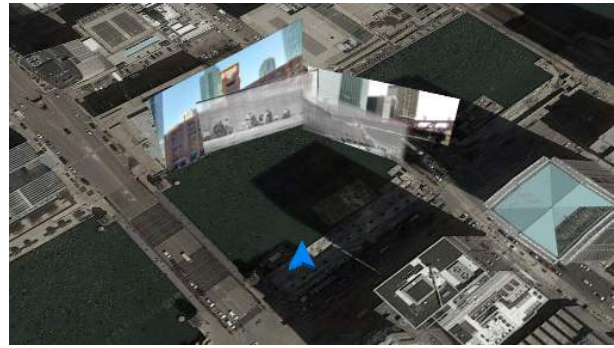


Figure 7: Two example screenshots of the virtual environment that we created in order to design our application. In *User mode* (left) we can move and rotate a virtual camera inside the environment and preview offline how a user would see the overlays from that perspective. At the same time, *Map mode* shows a pointer indicating our current position and orientation, allowing us to position ourselves in the desired positions. From both modes it is possible to move, scale, and rotate the overlays to personalize their appearance from a particular perspective.

a broad range of images, overlay illusions, site photography, historical annotation, audio and textual narrative—allow this content to be viewed off site, in other cities or countries, or at the user’s home.

## 6 AUTHORING TOOLS

In order to make it easy for designers to utilize our methods and to preview the AR content, we created a prototypical virtual to-scale environment of the Eastland Disaster site, exploiting the fact that our method leverages real-world correspondences and absolute camera pose estimation. Our idea involves the development of a system in which overlays could be placed and modified easily by using photographs of the current site as a reference. Placing these images as 3D objects within a virtual world requires many decisions about to their scale, a factor that is used by the mobile application in order to estimate the pose of the AR camera correctly. This virtual environment, containing panoramic imagery for each user location as well as 2D augment overlays, needs to be coordinated with positioned tracking images of specific features in the environment suitable for tracking. Though the location and number of tracking-suitable features are global, their position relative to the augmented content will vary according to the specific views and illusions available for each site.

As shown in Fig. 7, our authoring environment defines two different views, the *User mode*, a 1:1 scaled simulation where we can preview what the user would be able to see from a particular perspective, and the *Map mode*, a 1:100 scaled map representation with a 3D top-view perspective of the previous mode. In both modes, the designer is able to move a virtual camera that represents the mobile camera of a user running our application. We can move and rotate the camera around the virtual scene as a user would when walking on the Chicago Riverwalk and rotating the device to see points of interest. In this way, we can explore the scene from many perspectives and preview how overlays will appear in a user’s mobile device, without necessarily going on site to test our application every time we add new content, thus saving a great deal of time.

## 7 CONCLUSION

There is great interest for museum and archive curators and educators in creating AR experiences of their media archive. To date, the public discovers the extensive and fascinating historical media archives as documentary films or coffee table books. Augmented Reality projects offer an exciting possibility for presenting history within the relevant surroundings. The creation of a platform specialized for these projects could greatly increase the public adoption of public outdoor AR experiences.

## REFERENCES

- [1] C. Arth, C. Pirchheim, J. Ventura, D. Schmalstieg, and V. Lepetit. Instant outdoor localization and SLAM initialization from 2.5D maps. *IEEE Transactions on Visualization and Computer Graphics*, 21(11):1309–1318, 2015.
- [2] ARToolkit. "artoolkit.org". [Online; accessed 06/03/2016].
- [3] Aurasma. "aurasma.com". [Online; accessed 07/08/2016].
- [4] M. Cavallo and A. G. Forbes. DigitalQuest: A mixed reality approach to scavenger hunts. In *Proceedings of the IEEE VR Workshop on Mixed Reality Art (MRA)*, Greenville, South Carolina, March 2016.
- [5] S. Côté, P. Trudel, M. Desbiens, M. Giguère, and R. Snyder. Live mobile panoramic high accuracy augmented reality for engineering and construction. *Proceedings of the Construction Applications of Virtual Reality (CONVR)*, London, England, 2013.
- [6] D. A. Cucci and M. Matteucci. On the development of a generic multi-sensor fusion framework for robust odometry estimation. *Journal of Software Engineering for Robotics*, 5(1):48–62, 2014.
- [7] DAQRI. "daqri.com". [Online; accessed 07/08/2016].
- [8] C. Forster, M. Pizzoli, and D. Scaramuzza. SVO: Fast semi-direct monocular visual odometry. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 15–22, 2014.
- [9] S. Julier, M. Lanzagorta, Y. Baillot, L. Rosenblum, S. Feiner, T. Hollerer, and S. Sestito. Information filtering for mobile augmented reality. In *Proceedings of the IEEE International Symposium on Augmented Reality*, pages 3–11, 2000.
- [10] Layar. "layar.com". [Online; accessed 07/08/2016].
- [11] G. Reitmayr and T. Drummond. Going out: Robust model-based tracking for outdoor augmented reality. In *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 109–118, 2006.
- [12] G. Takacs, V. Chandrasekhar, N. Gelfand, Y. Xiong, W.-C. Chen, T. Bismpiagiannis, R. Grzeszczuk, K. Pulli, and B. Girod. Outdoors augmented reality on mobile phone using loxel-based visual feature organization. In *Proceedings of the ACM International Conference on Multimedia Information Retrieval*, pages 427–434, 2008.
- [13] Vuforia. "vuforia.com". [Online; accessed 07/08/2016].
- [14] T. Wachholz. *The Eastland Disaster*. Arcadia Publishing, 2005.
- [15] R. Wetzel, L. Blum, and L. Oppermann. Tidy City: A location-based game supported by in-situ and web-based authoring tools to enable user-created content. In *Proceedings of the International Conference on the Foundations of Digital Games*, pages 238–241, 2012.
- [16] N. Zhao. Full-featured pedometer design realized with 3-axis digital accelerometer. *Analog Dialogue*, 44(06), 2010.
- [17] F. Zünd, M. Ryffel, S. Magnenat, A. Marra, M. Nitti, M. Kapadia, G. Noris, K. Mitchell, M. Gross, and R. W. Sumner. Augmented creativity: Bridging the real and virtual worlds to enhance creative play. In *Proceedings of SIGGRAPH ASIA 2015 Mobile Graphics and Interactive Applications*, page 21, 2015.