How to add image to Location AR campaign

Viewing image in geo-location WebAR can add an extra layer of user curiosity and engagement. This can open an interesting engagement strategy for tourism industry to highlight a local attarction. It is easy to add such experience in Location AR.

Steps to Follow:

Create a new Location AR campaign. Click on the **View Details** and open the campaign. Go to **Web Editor** menu and click on **Edit** button. The default campaign comes up with a location marker (a-link component) pinned to the geo-coordinates along with a text. We need to remove it (so the circular red marker will disappear). Remove this section as highlighted below.

```
return [{
  name: "Marvin XR",
  location: {
    lat: 55.68793823859116, // change here latitude if using static data
    lng: 12.592178945883008, // change here longitude if using static data
  }
}];
}

function renderPlaces(places) {
  let scene = document.querySelector('a-scene');

  places.forEach((place) => {
    let latitude = place.location.lat;
    let longitude = place.location.lng;

    // add place name
    let text = document.createElement('a-link');
    text.setAttribute('gps-entity-place', `latitude: ${latitude}; longitude: ${longitude};`);
    text.setAttribute('title', place.name);
    text.setAttribute('scale', '3 3 3');
    text.addEventListener('loaded', () => {
      window.dispatchEvent(new CustomEvent('gps-entity-place-loaded'))
    });

    scene.appendChild(text);
  });
}
}

</script>
</head>

<body style="margin: 0; overflow: hidden;">
```

Edit   **Save**   **Cancel**

Add this new section which will display an image instead. Copy the following piece of code and insert instead. Please remember few things which will be beneficial for an optimum experience:

- Host your image in hosting site like https://imgur.com or similar hosting sites with public access.
- Scale your image (x:y:z). The further you go away from the geo-coordinates, the smaller it'll become.
- Position (0 0 0) will place the image at the base of 3d coordinates. Change them to shift position.
- Rotation will be helpful if you want to rotate your image in different direction.

```
let image = document.createElement('a-image');

image.setAttribute('gps-entity-place', `latitude: ${latitude}; longitude: ${longitude};`);

image.setAttribute('src', 'https://i.imgur.com/ORJswXY.png');

image.setAttribute('scale', '100 100 100');

image.setAttribute('position', '0 0 0');

image.setAttribute('rotation', '0 0 0');


image.addEventListener('loaded', () => {
    window.dispatchEvent(new CustomEvent('gps-entity-place-loaded'))
});
```

```
    scene.appendChild(image);
```

```
}

function renderPlaces(places) {
let scene = document.querySelector('a-scene');

places.forEach((place) => {
let latitude = place.location.lat;
let longitude = place.location.lng;

// add image to geo-location
let image = document.createElement('a-image');
image.setAttribute('gps-entity-place', `latitude: ${latitude}; longitude: ${longitude};`);
image.setAttribute('src', 'https://i.imgur.com/ORJswXY.png');
image.setAttribute('scale', '100 100 100');
image.setAttribute('position', '0 0 0');
image.setAttribute('rotation', '0 0 0');

image.addEventListener('loaded', () => {
      window.dispatchEvent(new CustomEvent('gps-entity-place-loaded'))
});

scene.appendChild(image);
});
}
}

</script>
</head>

<body style="margin: 0; overflow: hidden;">
<p id="banner" style="font-family: Arial, Helvetica, sans-serif; font-size: 26px; color: white; margin-left: 44%;"><b>Marvin
XR</b>
</p>
<a-scene vr-mode-ui="enabled: false" embedded arjs>
```

If you want to put your branding experience, replace 'Marvin XR' with your desired text. Now click on **Save** button to save the configuration. It's ready for testing. Go and test your new Location AR experience.

Add Additional Image to Location AR

It's possible to add more than one image to your Location AR campaign. You need to add similar **<a-image>** component and append it to the **<a-scene>**. Please remember that if you add exact geo-coordinates for both the images, they will overlap and the second one will be visible only.

It's a good practice to choose a different geo-coordinate for the second image (for example, 50-100 meter apart in different direction) to make sure both of them are visible side by side or in different direction. You can also use positioning parameter to slide their absolute positions. Always select different variable name (image, image 2 etc.) for each image element (to avoid override).

While you put the geo-coordinate values, please remember to limit the digits after decimal to up to six digits for optimal performance. Google map provides 14 digits after decimal which can increase

computation time for the AR campaign to evaluate data everytime, can cause slight delay in performance.

```
places.forEach((place) => {

let image2 = document.createElement('a-image');
image2.setAttribute('gps-entity-place', `latitude: 55.696068; longitude: 12.505245;`);
image2.setAttribute('src', 'https://i.imgur.com/HxYx4ZT.jpeg');
image2.setAttribute('scale', '100 100 100');
image2.setAttribute('position', '0 0 0');
image2.setAttribute('rotation', '0 0 0');


image2.addEventListener('loaded', () => {
    window.dispatchEvent(new CustomEvent('gps-entity-place-loaded'))
});


scene.appendChild(image2);
});
```

```
places.forEach((place) => {
let latitude = place.location.lat;
let longitude = place.location.lng;

// add image to geo-location
let image = document.createElement('a-image');
image.setAttribute('gps-entity-place', `latitude: ${latitude}; longitude: ${longitude};`);
image.setAttribute('src', 'https://i.imgur.com/ORJswXY.png');
image.setAttribute('scale', '100 100 100');
image.setAttribute('position', '0 0 0');
image.setAttribute('rotation', '0 0 0');

image.addEventListener('loaded', () => {
    window.dispatchEvent(new CustomEvent('gps-entity-place-loaded'))
});

scene.appendChild(image);
});


places.forEach((place) => {
let image2 = document.createElement('a-image');
image2.setAttribute('gps-entity-place', `latitude: 55.696068; longitude: 12.505245;`);
image2.setAttribute('src', 'https://i.imgur.com/HxYx4ZT.jpeg');
image2.setAttribute('scale', '100 100 100');
image2.setAttribute('position', '0 0 0');
image2.setAttribute('rotation', '0 0 0');


image2.addEventListener('loaded', () => {
    window.dispatchEvent(new CustomEvent('gps-entity-place-loaded'))
});


scene.appendChild(image2);
});
}
```

Create a Virtual Image Gallery with Various Images

This part can be little tricky but it can greatly enhance your Location AR experience. You can use this for various purposes. For example, an art gallery can showcase open virtual gallery of paintings at an open exhibition. Traesure hunt can be built around with such experience. Live shows can engage audience with various virtual images floating around them like an open gallery to entertain them. Once again, imagination is your limit here.

Like the previous step described above, you can add various <a-image> block to the campaign. But to show the images all around without overlapping, it's a good idea to spread them in an even state. Here's come the trick. Select up to six geo-coordinates, 50-100 meter apart, in different directions and use each of them for an <a-image> element. For optimal experience, select 6 images per campaign. You can experiment with more to see if they fulfill your criteria.

Note: Geo-location AR is based on satellite data. When a user starts the AR campaign, it fetches data from more than one geo-stationary satellites and assemble them to a singular data to use (since each satellite location data slightly varies based on its angular position from the observer). Now this to and fro communication takes around 240 ms which means there will always be a slight delay from a runtime information.

How this is related to Location AR? Since the satellite data is gathered continuously and assembled, each time the average data calculation can slightly differ from the previous one. This can result in slight shaking of the virtual image. If you focus on one image, this will be shown as stable. But as you move along (even if a couple of steps would force GPS to recalibrate your location), the virtual objects can slightly move in or away adjusting to on your new location every time.