**Objective:**
Our primal objective is to establish a walkthrough procedure running on inexpensive PCs in real-time. Therefore, the real-time computation should be kept at minimum but give users a feeling of walking in virtual space. Users should be able to move around on a floor and look around at will.

**Related Work:**

### 3D reconstruction:
The traditional method to represent a virtual world is a 3D reconstruction of the real world. 3D reconstruction methods provide users a lot of degrees of freedom but it needs a complex capturing system and large computation resources as discussed in [1].

### Light field:
Light field rendering offers an easy and fast way to render virtual world for walkthrough systems, but it needs to sample enormous data of light ray light in the target space [2].

### QuickTimeVR:
QuickTimeVR plays panorama movies just along the axis of time as it was shot but has no degrees of freedom except for view direction [3].

**Methods:**

Our algorithm is divided in 2 distinct processes: offline and online processes.

**Offline:** this process is done beforehand once to prepare the data.

Identifying feature points: feature points must match among the original 4 panorama images.

For each panorama, we map feature points onto a sphere using polar coordinate system and triangulate the sphere using the convex hull algorithm as described in [4]. Polygonal spheres should have the same topology for every panorama image.

**Online:** this process is done in real-time during walkthrough. Instead of computing the RGB value of each pixel on panorama images, we compute only UV coordinates for texture mapping, and render the reference panorama with a new UV map.

The first step is to interpolate the polygonal closed surfaces in proportion to distance from each of 4 original location. This gives a new polygonal mesh at the current location with the same topology as reference ones.

On the current sphere, we cast a ray from its center to each grid point of the sphere mesh to identify the intersected triangle and the barycentric coordinate of the intersection point.

To obtain corresponding UV coordinate value from one of the original panoramas, we use the previously intersected triangle and barycentric coordinate value to cast a ray passing by the sphere’s center. The intersection of the ray and the sphere gives a new UV value which is to be mapped to the sphere at current location.

**Experimental results:**

This method allows interpolating original panorama images and generating a new view at the current location in real-time.

**Reference:**


