March 15, 2019

FINE-GRAINED VIEW INTERPOLATION IN MULTI-VIEW AND LIGHT-FIELD CODING

Thomas Davies

FINE-GRAINED VIEW INTERPOLATION IN MULTI-VIEW AND LIGHT-FIELD CODING

AUTHORS:
Thomas Davies

ABSTRACT

Techniques are described for increasing the number and accuracy of views in multi-view, three-dimensional (3D), and light-field systems, by leveraging low resolution sensors to provide geometry and object location calibration when interpolating new views.

DETAILED DESCRIPTION

Modelling background occlusion and insertion is a key problem in Virtual Reality (VR), 3D, and especially Augmented Reality (AR) video communication. As participants in a call move, the area of the background that is visible to the participants' changes. A multi-camera system can select views that correspond to the relative position of the participants and can display an appropriate view. However, in practical systems the number of cameras is limited, and additional views need to be interpolated. This interpolation must solve boundary problems such as determining the revealed/concealed boundary between the foreground objects and the background. It must also solve distortion problems, such as undoing the projection effects of each of the views to get a realistic interpolated new of a rounded object with significant depth. For realistic experience, both the segmentation and the geometrical warping need to be accurate, but an interpolated view may misplace the location of objects by significant amounts.

These problems are difficult to solve because the camera positions are highly under-sampled spatially and determining the occlusion boundaries is highly uncertain as the light rays that determine their location are tangential to objects with depth. If the objects were flat, one could be sure that light rays at that edge would brush the edge at the same point of whichever camera they entered, but this is not so in the real world. Estimating the edge positions in a virtual view therefore has a lot of uncertainty.

The basic problem of view interpolation is illustrated in Figure 1 and Figure 2, below.

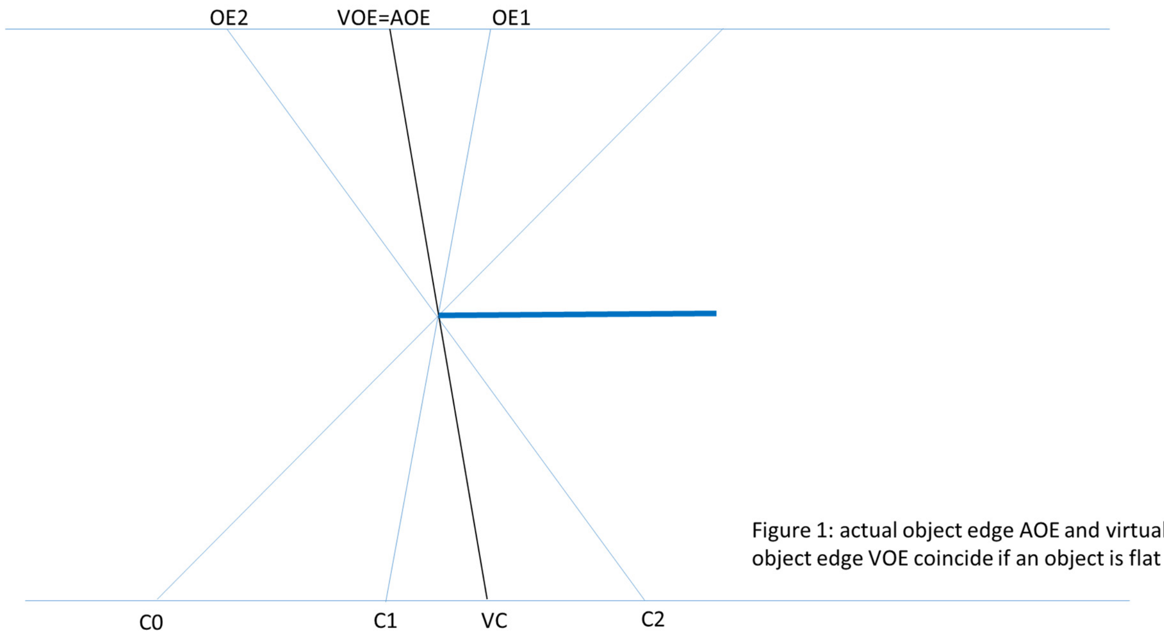

Figure 1

In Figure 1, it can be seen that the position of the edges of a flat object in a view from a virtual camera (VC) can be precisely determined from the edge positions determined by the real cameras C1 and C2 and the real observed edge (OE) positions OE1 and OE2 - in other words, the Virtual Object Edge (VOE) equals the Actual Object Edge (AOE). [Note that these diagrams are simplified by assuming a common view plane, which could be achieved by a global transformation between the C1 and C2 view planes and the virtual camera view planes.]

In Figure 2, it can be seen that if an object has depth the Virtual Object Edge interpolated from the real camera object edges can differ substantially from the Actual Object Edge.

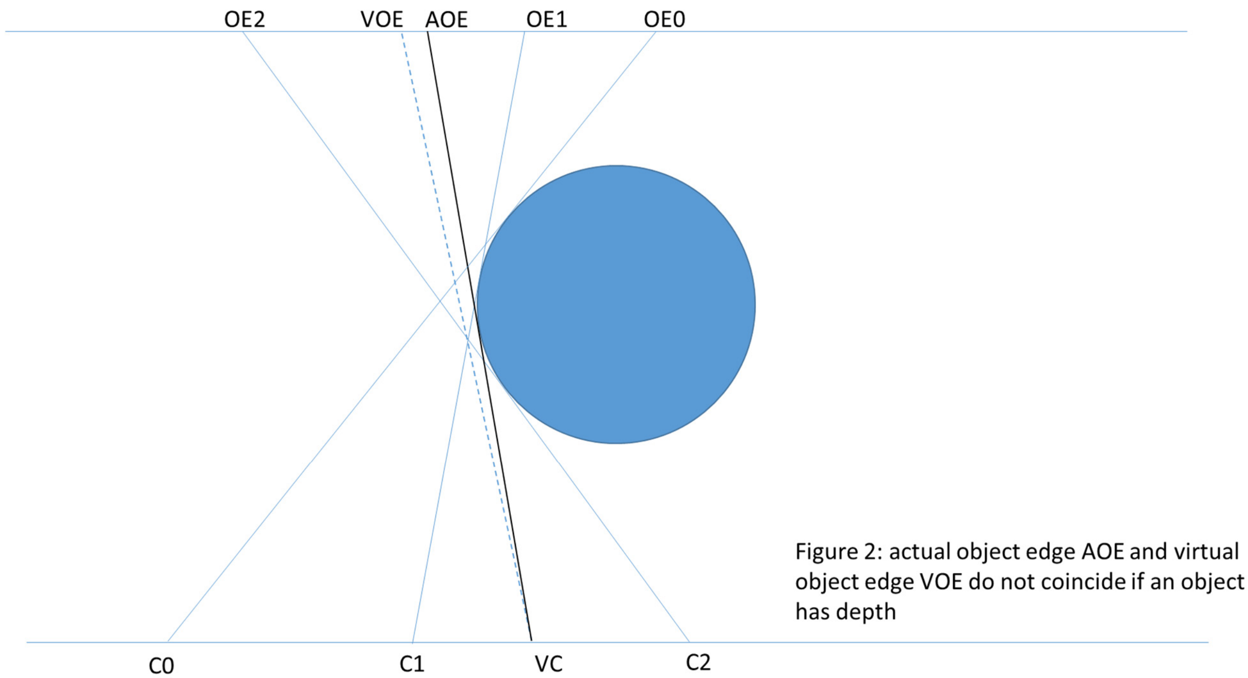

Figure 2

This edge position is not only wrongly determined but, in practice, is uncertain without knowledge of the geometry of the tangential area at the edge of the object. In real-time systems, in particular, it is difficult to extract the object data and depth model quickly and accurately to determine the correct position of the edges. A related problem is that if the object is incorrectly positioned in the virtual view, the local geometry will be incorrectly rendered and objects will look flat and lifeless, like cardboard cut-outs.

Figure 3, below, shows that the AOE and VOE can be made to correspond more closely if the views used for generating the interpolation come from cameras that are closer together and closer to the virtual camera position.

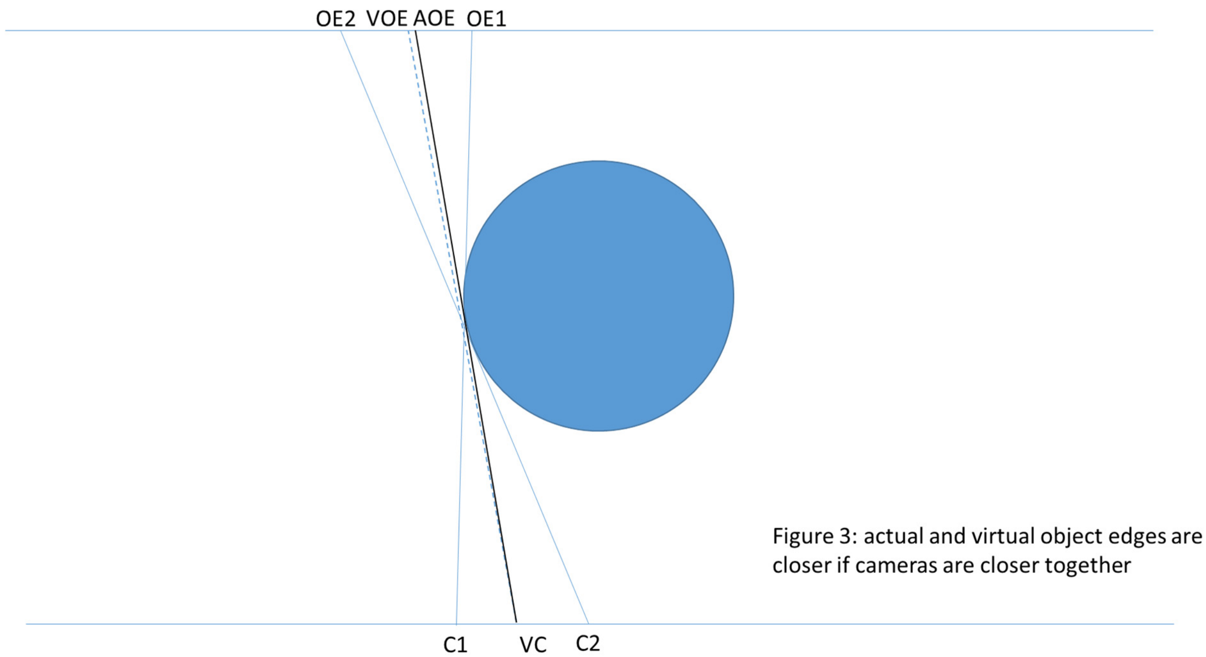

Figure 3

This proposal, therefore, adds additional cameras to provide better estimation of true object position and orientation in the virtual view. The additional cameras may be a series of low resolution cameras or light-sensitive strip sensors between the high resolution camera positions, and/or a moving camera tracking left to right or up and down adjacent to a row or column of high resolution cameras.

In either case, the spatial resolution needed for the additional cameras is not high. All that is needed is to provide reference phases for the location of edges in the virtual view. The virtual view can be registered against the object positions determined in the low-resolution view.

For decoder-side rendering the low-resolution reference view can be sent as a guide by the encoder. Since it is low resolution not much in terms of extra bits is required.

Accordingly, the novel aspect for this proposal is using a number of additional cameras that may be low spatial or temporal resolution to provide extra accuracy in locating edges and determining occlusions without great increases in the number of pixels to be processed.

Figure 4, below, shows how a low resolution camera can be used to derive a distribution of possible edge positions centered close to the true edge position.

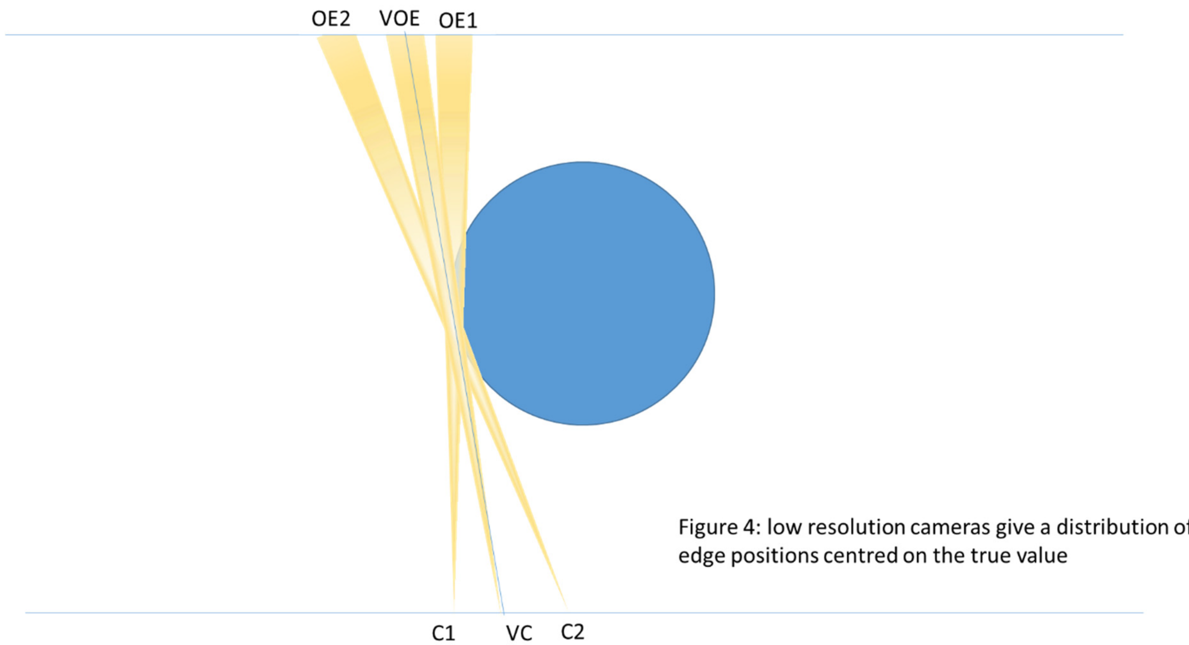

Figure 4

The aperture of each camera can be considered as averaging a light cone onto a set of pixels that provide a low resolution image at each camera position. These low resolution images, when interpolated based on position, give close to the true edge position. Conventional edge detection images typically use blurred images for processing to reduce noise and to extract the edge phase information without interference.

The signal processing architecture is summarized in Figure 5, below. Low resolution images providing high view precision are used to determine object and geometry corrections for a high resolution view interpolator. The low-res images come from an array of low resolution sensors, either stationary or moving. The high resolution images come from more sparsely located high resolution cameras.

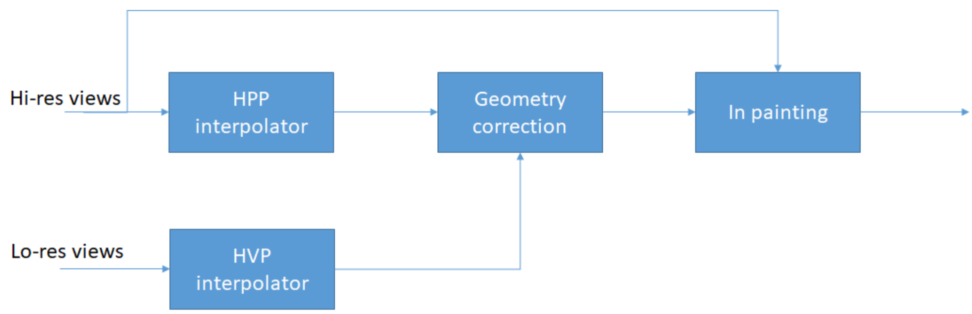

Figure 5: using a High View Precision (HVP) interpolator to correct a High Pixel Precision (HPP) interpolator

Figure 5

Feature mapping techniques can be used to generate correspondences between the two interpolated views and generate local warping parameters. Since the object locations may be different, in-painting of revealed background from the high resolution camera images is likely to be needed.

Figures 6 and 7 show some configurations for camera systems.

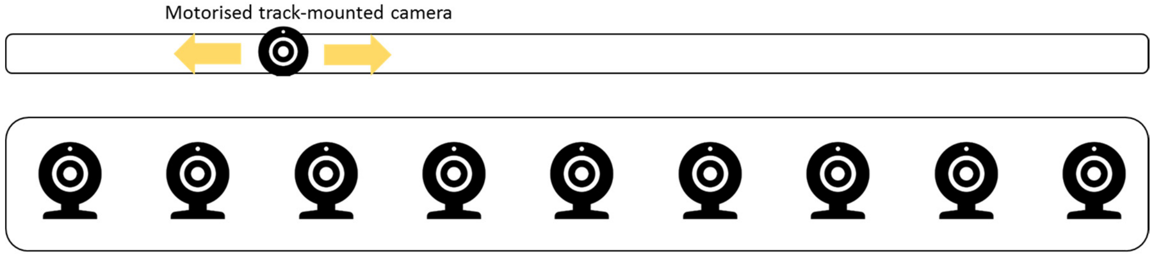

Figure 6

In Figure 6, a motorized track containing an additional camera is placed adjacent to the static cameras. This sweeps at speed along the line of static cameras. The motion itself will cause blurring and reduce the effective resolution of the camera. The motion of the camera may also cause motion within the field of view to be aliased and so the camera must move at sufficient speed to limit this effect: this might be of the order of one transit of the track in a frame period.

Alternatively, a strip of low resolution sensors could be laid along the track behind a series of motorized prismatic lenses. Moving the camera along the track can be emulated by moving the prismatic lenses to direct light from different directions.

Figure 7 shows a static grid arrangement in which the low resolution cameras are densely distributed between high resolution cameras.

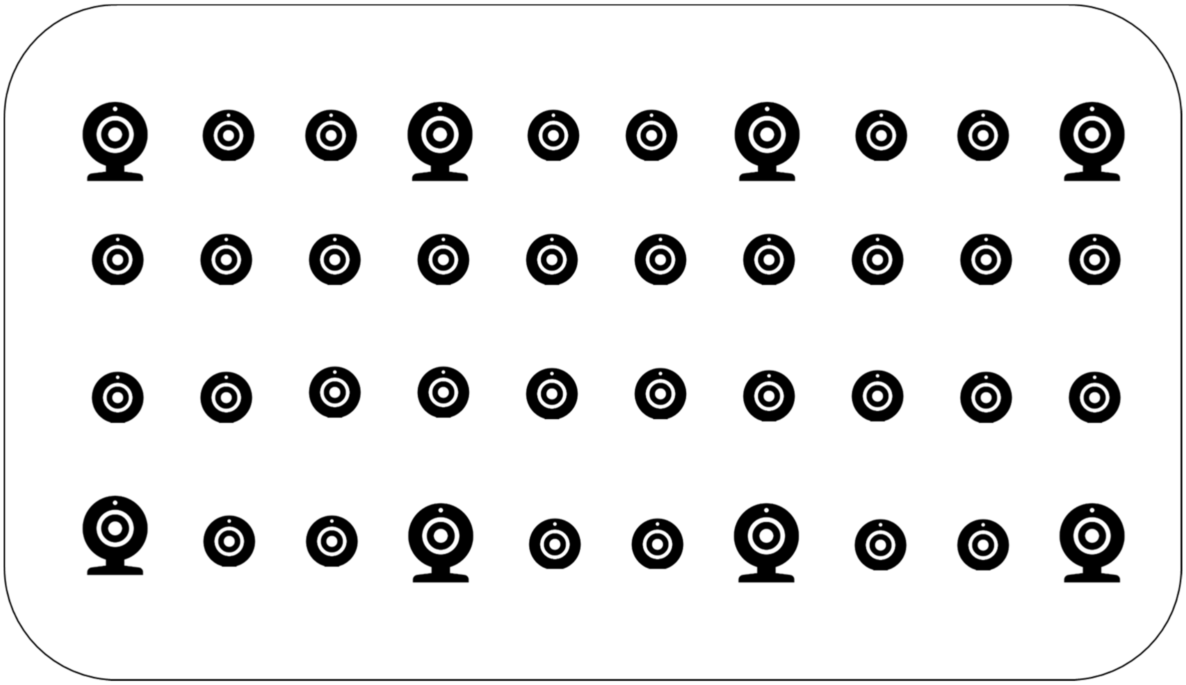

Figure 7

In summary, techniques are described for increasing the number and accuracy of views in multi-view, 3D, and light-field systems, by leveraging low resolution sensors to provide geometry and object location calibration when interpolating new views.