

143

Using laser scanning to produce 3D isovists of real environments

Nicholas Sheep Dalton

The Open University, Department of Computing & Communications
n.dalton@open.ac.uk

Ruth Conroy Dalton

The University of Northumbria at Newcastle
ruth.dalton@northumbria.ac.uk

Iwan Peverett

ip a-viz
iwan@iwanpeverett.co.uk

Sarah Clinch

Lancaster University, Department of Computing
s.clinch@lancaster.ac.uk

Abstract

This paper is essentially a technical guide to laser scanning for real-world, two-dimensional isovist creation. The paper covers what is laser scanning, good scanning practice and then describes how to use the resultant data to recreate an isovist directly from noisy, real-world scan data. We will demonstrate how two different isovists can be created: the traditional 2D isovist, and a weighted isovist generated from the surface of a sign, display or shop frontage. This second isovist is weighted by the viewing angle of someone looking at the sign or display. Future areas of research identified from this paper include: work on 3D isovist representations and methods to efficiently process the point cloud data (produced by the scanner) in order to calculate a volumetric isovist; using the colour data, also captured by the scanner, in order to generate potential, colour-based, isovist representations; future work on the placement of signs and displays for optimal efficacy.

Keywords

3d isovists, laser scanning, spatial cognition, visual representation, signage and displays.

1. Introduction: Moving from abstract isovists to real-world isovists

An isovist is essentially a simple, two-dimensional polygon representing the totality of all-visible, and potentially visible space from a specific location (i.e. the full 360° view, experienced sequentially if a person rotated slowly on the spot).

The term was coined by Tandy, who focused on the natural landscape (Tandy, 1967) and was subsequently introduced into built environment analysis by Benedikt, who, in turn, devised many of the isovist measures in still use today (Benedikt, 1979). The problem is that almost all research on isovists has been based on 'abstract' isovists. By this we mean that although the isovist may be based on real-world locations, they are generated using representations (usually two dimensional drawings) of real world spaces, buildings and locations as a starting point. Most hand-drawn isovists, or more typically, isovist generating software, need a base-model indicating the solid, occluding elements of a building and the converse, the occupyable, navigable space. However, these are abstractions of real world space. In the real world, walls in buildings are rarely perfectly flat or 'true' and corners rarely form perfect right angles. The real world is noisy, messy and imperfect. Our CAD

models (from which we produce isovist analyses) are reifications of the real world: abstracted, generalized and perfected. Just as a mathematical line, as a concept, is the reification of a straight line in the real world, so our idealized isovist representations are perfected abstractions of real-world isovists (for a more detailed discussion on how isovists, and other space syntax representations, are reduced, yet embodied, abstractions of real world phenomena, see Dalton's paper on space syntax and spatial cognition (Dalton, 2005).

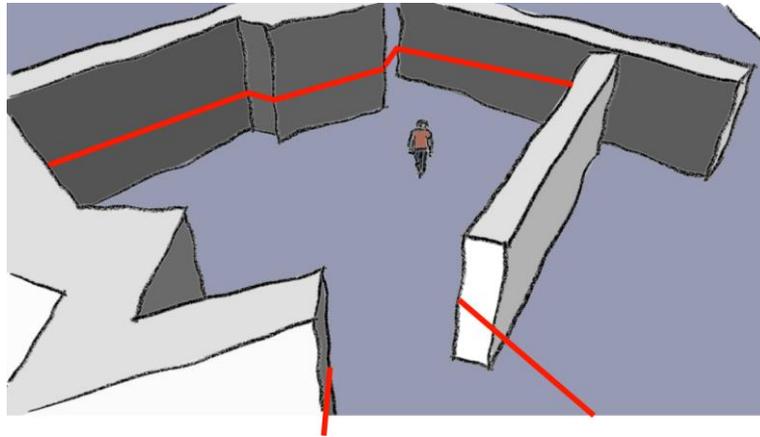


Figure 1: Basic 2D Isovist Indicated by the Red Line

For the most part, the reified abstraction of the isovist is not an issue; just as there is no problem with having a mathematical concept of the 'line' that has properties of zero width, perfect straightness and potential infinite length (plus the theoretic possibility of meeting other infinitely long, parallel lines at infinity). However, what happens if we do want to produce isovists of the real world that are as true as possible to the experience of being in the world and reflect all the noise and idiosyncrasies of real-world architecture? Equally we may also wish to record the experience of being in the real world, in spaces or environments for which there are no convenient, pre-prepared drawings (What do you do if plans are not available? The intermediate step of surveying, measuring and drawing a building can be extremely time-consuming and labor-intensive). Another situation, when true, real-world isovists might be useful, is when drawings or measured plans might be available but the building is so complex (particularly in the third dimension), or when the building's owner has made changes or alterations, so that only by capturing the real world isovist can we be absolutely sure that we have recorded it accurately and correctly.

There is a way of accurately capturing the visual experience of being in the real world and that is simply by using a 3D laser scanner. Laser scanners are not at all new but are typically used for different purposes (predominantly for surveying and documentation) than to produce isovists and/or isovist measures from the real world.

This paper is essentially a technical guide to laser scanning for isovist creation. We will cover what is laser scanning, good scanning practice and then describe how to use the resultant data to re-create an isovist directly from the scan data. We will show how two different isovists can be created: the traditional isovist, and the isovist from a sign/display surface or, by extension, from a shop frontage.

2. What is laser scanning?

Laser scanning, as a term, covers multiple types of survey-data capture devices and techniques: a laser scanner being defined by Böhler and Marbs (2002) as "*Any device that collects 3D co-ordinates of a given region of an object's surface automatically and in a systematic pattern at a high rate achieving the results in near real time.*" These can range from aerial data capture to short- or long-range terrestrial data capture. Although all types of laser scanning are forms of 'light detection and ranging' the acronym LIDAR is normally only used for aerial data capture, whereas laser scanning, as the more general term, is used for terrestrial forms of data capture. Recent developments in the portability, speed and continuously falling costs of laser scanning technology have led to an

expanded use of this technology away from the more traditional surveying domain. Laser scanning is essentially a process of capturing and recording millions of visible points located accurately in 3D space for manipulation and analysis. For the recent Lancaster University Campus Display Study (see the accompanying paper on the research, Dalton et al., 2015) we employed a terrestrial laser scanner, namely the Faro Focus 3D, to capture accurately the 3D space at 19 locations of digital displays (large, flat screen, LED displays deployed around the campus for institutional information-dissemination and communication purposes) in order to analyse the effectiveness of their location.

Laser scanners work on two technologies. The older time-of-flight scanners are arguably more accurate but take longer to scan the area. They are, in effect, laser range finders sending out a beam and measuring the time taken for the return bounce. These scanners have an upper range of about 300m, although some can measure in kilometers with lesser resolution. They measure roughly 50000 points per second. As the technology has been refined, more modern time-of-flight laser scanners can measure a million points per second but have a limited range and are comparable with the other phase-based option. These scanners utilise a constant beam of laser, rather than a pulse, and measure the phase shift of the returning energy. While faster, phase shift scanners are not capable of scanning distances more than 60 to 80m. As distances were limited within the spaces of the Lancaster University Campus Display Study, the phase-based, and easily portable, Focus 3D enabled many more points to be captured in a given time.

The Lancaster University Campus Display Study required a number of spaces to be laser scanned in order for their 3D spatial qualities to be analyzed and combined with configurational data on the location of the screens. Many of these spaces were confined and required just a single scan to be taken. Others were more complicated or were located in larger sites that required multiple scans with the use of targets. The targets allow post-production registration of scans that accurately align the multiple scan locations together using the three dimensional relationship of the targets. The individual scans were registered in Faro Scene software and exported as an ASCII file format where x,y,z, intensity and colour data can be extracted. In this exercise the main focus was on where the individual points of the scan were located in x,y,z coordinates, or three-dimensional space.

3. Example scan data

Format: VERTICAL HOZ X Y Z R G B

0 31 0.00670000 0.00380000 0.98050000 253 253 253

**4. How to
turn the scan
data into an
isovist**

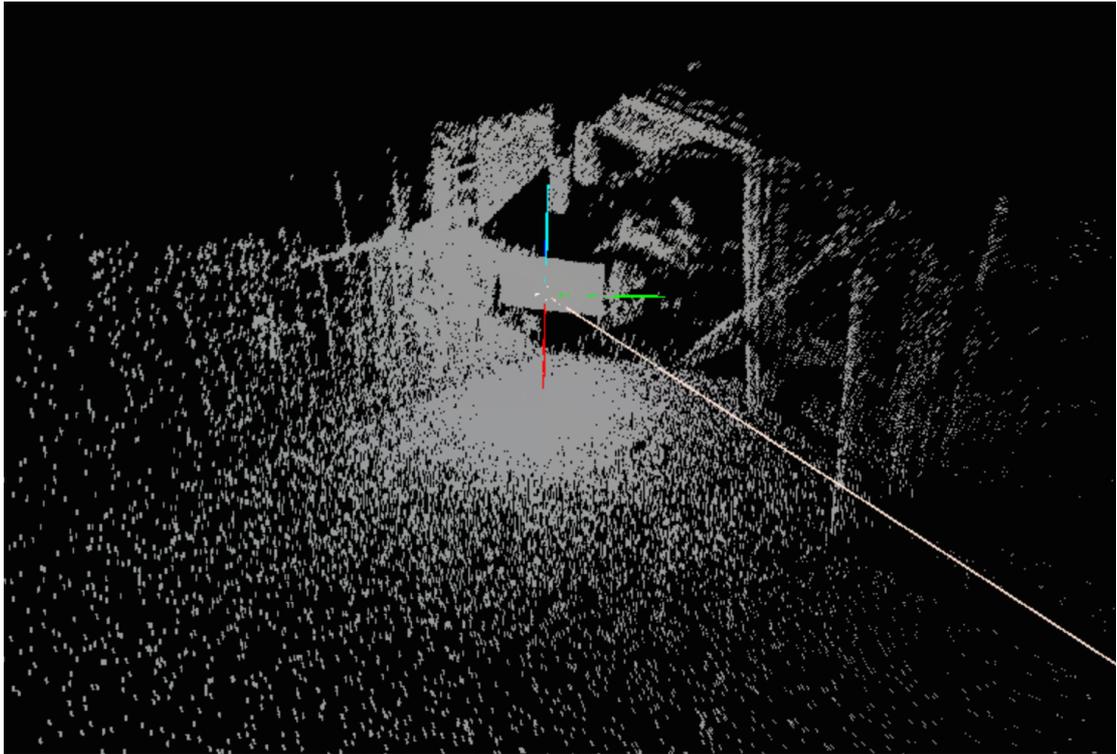


Figure 2: Example Scan Point Cloud Data Set

In Figure 2 we can see a typical data point in the raw point cloud data. In the Lancaster University Campus Display Study, using the Faro Focus 3D scanner, we were generating, on average, 9,912,240 points for each scanning location (and since in the study we scanned 19 locations, this resulted in over 188 million data points. The challenge is how to take this vast, raw data set and use it to produce something that closely resembles the theoretic isovists that have become a standard representation in space syntax research plus the equivalent isovist measures. We will first describe how to produce a basic isovist from such point cloud data. In the following section we will describe how this has to be adapted to produce a 'display isovist' or an isovist generated from the front of a display.

The first task is to establish the centre of the isovist. In our case, we had carefully positioned the scanner, so that the scanner centre was also the location at which we wanted to generate the isovist. Essentially the scanner is in the middle of the point cloud and acts as the origin of the dataset, in other words it is coordinate location 0,0,0. If there is a requirement to offset the isovist location, perhaps adjusting slightly for height, then this offset must first be worked out (see the following section). For this basic isovist generator, let it be assumed that the scanner centre and the isovist centre are correctly aligned and are coincident.

The essential problem is that there are a large number of points, most of which need to be discarded, and they are just 'out there' in x,y,z, space, with no obvious ordering. The method used in this paper was to start from the centre, 0,0,0 and to generate a series of horizontal rays, sweeping around the point cloud, at equally spaced angular increments. For each ray, we performed a dot product test between the ray and each and every point in the cloud (all 9 million points) and if the result of the dot product was less than a prescribed, and user-determined, maximum amount (essentially based on the angular distance between two consecutive rays) then that point is stored, and otherwise it is discarded. Once all points have been tested, each ray, has a cluster of x,y,z, points associated with it, which are those points that are closest to that ray (rather than to any other ray). See Figure 2 below to show the relationship between the isovist centre, the ray, and the cluster of x,y,z, points satisfying the dot product test.

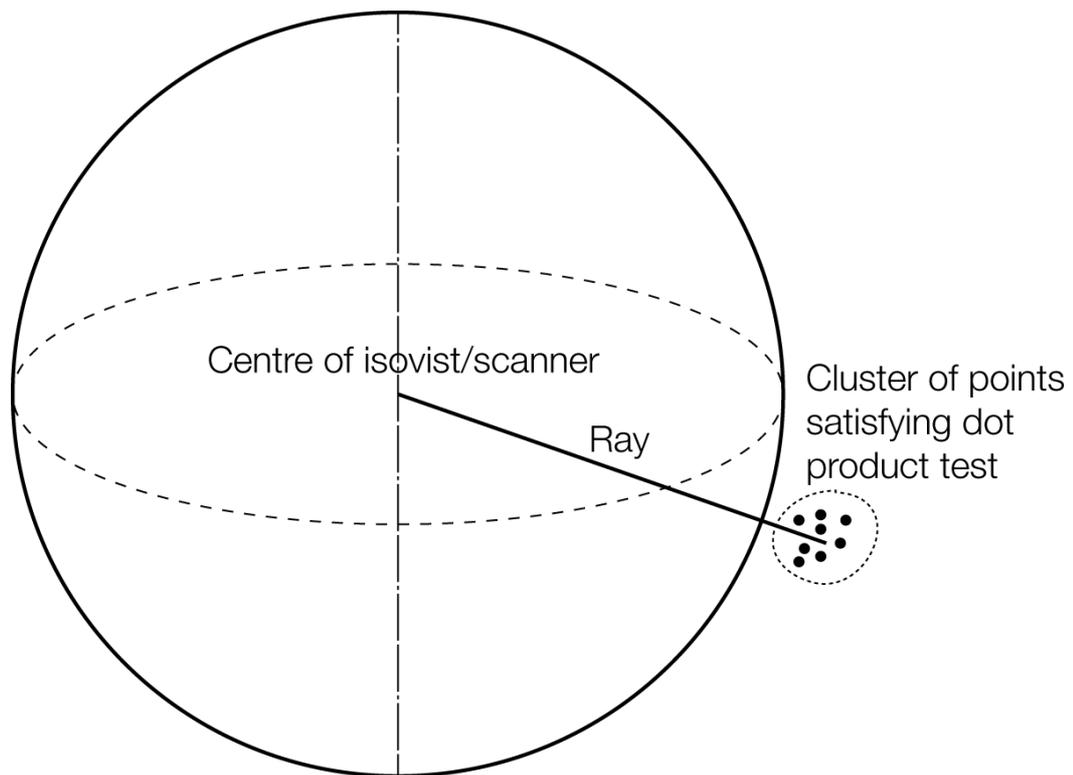


Figure 3: The Relationship Between the Isovist Centre, the Ray, and the Cluster of 'Nearest Points'

At this stage, each ray will have a sub-set of the point cloud associated with it, and this first 'cull' will have produced a mini-point-cloud cluster of between a few hundred to a few thousand points (on Figure 3 the 8 points shown are, in reality, hundreds!). This cluster of points is then reduced to a single point, by simply finding the centroid of the sub-set of the point cloud. At this point the rest of the 9 million point can be discarded and every ray has a single x,y,z point in space associated with it. If 360 rays were generated, then 360 points will form the basis of the isovist polygon. At this point, since the rays are generated incrementally, the list of points produced will automatically be ordered (they are stored in the order in which they are calculated), and so can quite easily be transformed into a closed isovist polygon, by joining them up in sequence.

This may, at first, seem a rather laborious computational method of doing it, since, each ray needs to be tested against all of the 9 million points in the cloud. The other, perhaps more obvious way of doing this is to utilize a z-sort algorithm (which would require an $n \log^n$ time to process), in which the software immediately culls all of the points in the point cloud above and below eye-height (or whatever height is being used), leaving only a slice of points from which to produce the isovist. The essential problem with this, is that at this stage you are still left with an unordered, unsorted set of hundreds of thousands of points and the only way to transform this onto a polygon is to use some sort of adapted or modified hull finding algorithm which is also computationally expensive. In the method presented in this section, although it takes more time to test each point against each ray, once this has been done, many other things become far easier and computationally far quicker. For example, the ordered list of points permits a very quick and easy algorithm to determine the triangular area bounded by two rays and part of the perimeter of the isovist. The area of the isovist is therefore computationally nothing more than summing the area of these small triangles. Furthermore the sum of the distance between the sequential points is clearly the perimeter. Once you have a sequential set of points forming the boundary, the area and perimeter calculations could not be more straightforward. And, of course, once the area and perimeter have been calculated many other isovist measures can simply and easily be derived from these two fundamental measures.

There is, however, another important reason for using the ray and dot product test, rather than a z-sort algorithm and this is because it makes it just as simple to produce isovists for planes other than horizontal planes. If we relied on a z-sort culling, then we would have to use a different method for

non-horizontal isovists. Using this method, we can produce sloping isovists (if the ground plane is sloping), adjust for the scanner not being horizontal or produce sectional isovists (in the z-plane), if we are interested in sectional views, or any other plane that might be of interest. The same method works just as easily, regardless of orientation. In summary, this method is more flexible as well as being computationally efficient.

5. How to turn the scan data into a display isovist

In the Lancaster University Campus Display Study we were not, however, interested in generating typical isovists. The point of this study was to capture the isovist from the front face of a digital display. Such isovists are also known as ‘façade’ isovists, when the isovist from the front face of a building (for example a shop front) is required. Such ‘façade’ or ‘display’ isovists essentially represent the area from which passers by can see the display in question. In this next section we will explain how the algorithm described above was adapted for digital displays.

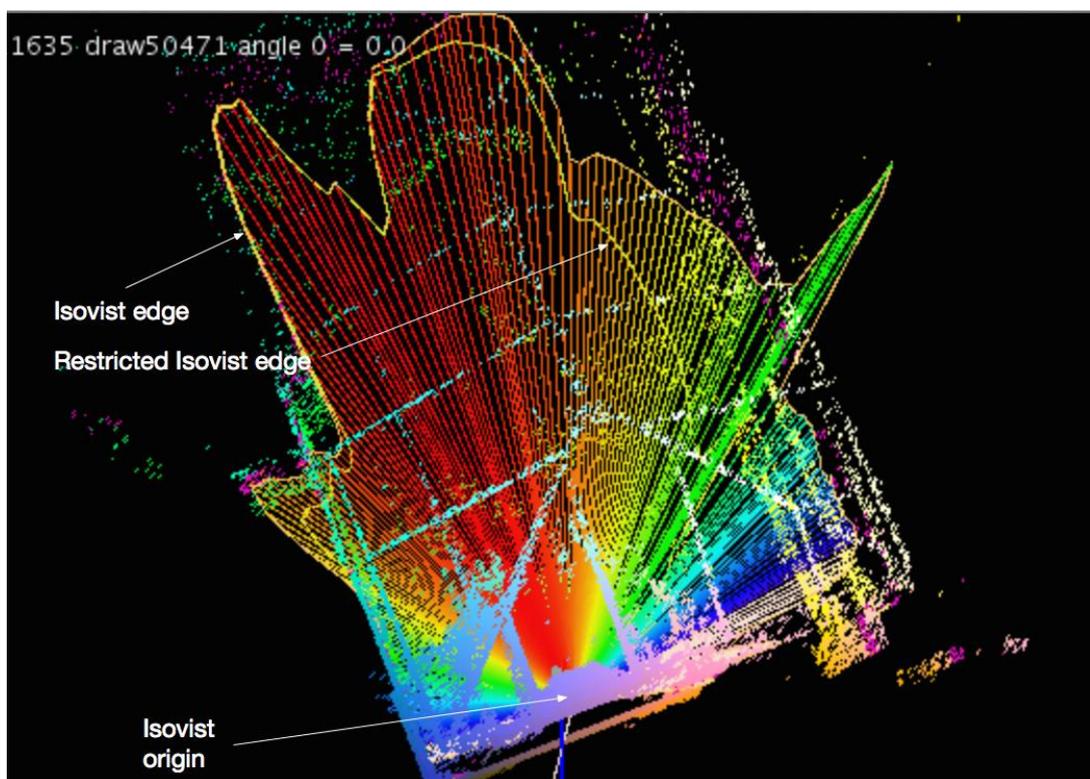


Figure 4: Example Scan Point Cloud Data Set Showing Isovist Edges

In contrast to the simple isovist process described above, the scanner is not actually at the centre of the screen. In order to create the scan, the scanner must be placed just in front of the screen which means that there is an amount of offset, in the case of this study and using the Faro Focus 3D scanner, the closest we could get the scanner to the face of the display was approximately 10cm (this should be measured and recorded onsite when scanning), and hence this is the horizontal offset, δ^h , required to re-set the origin of the isovist, so that it is perfectly aligned with the centre of the screen. In terms of setting up the scanner, since the height of most scanners is adjustable, we were able to adjust the height of the scanner so that it was level with the centre of the face of the digital display. The first stage, therefore, in producing the isovist, is to manually insert the centre of the display into the software, by inputting the offset amount, δ^h . You then need to place, by hand, a second point, which represents the normal vector from the face of the display, see Figure 6.

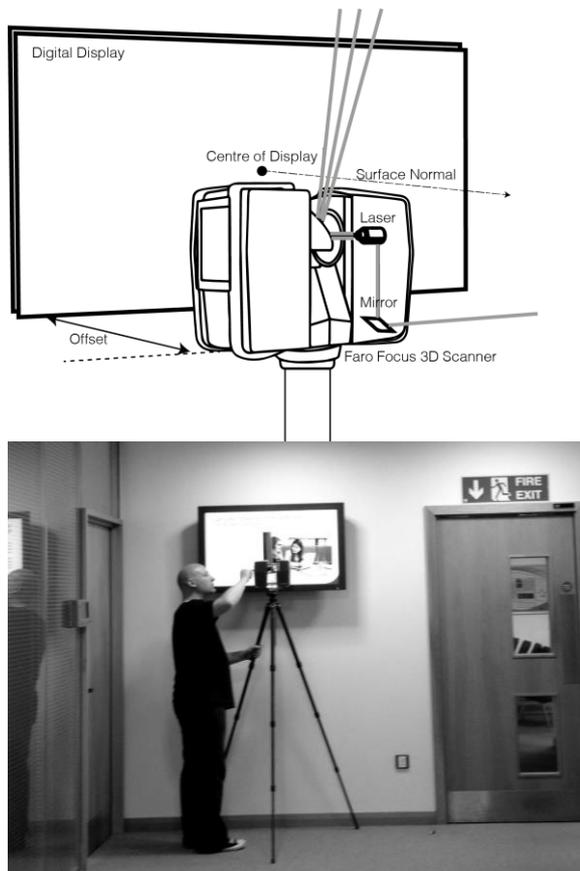


Figure 5: Relative Positions of the Scanner, the Display Centre and the Horizontal Offset (vertical offset not shown for sake of clarity)

Now that a new offset centroid has been set, the rays can be generated as before, but this time from the centre of the display and not the scanning centre. The original scan, when producing the point cloud data will have omitted anything behind the display as it will not have been visible, and so the resultant isovist shape is very 'lop-sided'. The second stage, namely that of testing to see which points in the cloud are closest to the generated ray, and then replaced this cluster with a single point, resulting in an ordered set of points, is undertaken in exactly the same way as for the previous isovist.

There is, however, an additional stage in the process. When it goes on to the final stage of calculating the area and perimeter values, the algorithm weights both of these measures by the Sine of the angle between the ray and the surface normal, adapted from the work on signage readability by Xe et al. (Xe et al., 2007). The reason for this weighting is to take into account user viewing angles; if you are viewing a display from a position directly in front of the display, the information being shown will be more legible, than if you are standing at an oblique angle to the display, whereby the information shown will be visually distorted. This weighting therefore is designed to take into account the visual distortion caused by viewing displays at oblique angles. The resultant of this process is two separate polygons, the full, usual, isovist polygon and, in addition, a second cardioid-weighted isovist polygon. As a consequence of generating two isovist polygons, all isovist measures are duplicated; this algorithm produces both a weighted isovist area and a weighted isovist perimeter measure. For more information on the weighting please refer to the accompanying paper on the Lancaster University Campus Display Study by Dalton et al. (2015). See Figure 3 for the difference between the isovist edge (the un-weighted perimeter) and the restricted isovist edge (weighted by readable angle).

The software, Nebula (Dalton, 2015), to read in the raw scanner data and process it to produce the basic 2D isovist representation and measures as well as the optional weighted display-based isovist was written by Dalton in Java. There are plans to make it freely available/open source in the future.

6. Conclusion

The primary conclusion of this paper is that 3D scanning techniques permit the creation of real-world isovists, in an easy and accessible manner, which has not been hitherto possible. Since 3D scanners are becoming more ubiquitous and their cost is coming down, this opens the possibility of scanners being used in more creative and research-focused ways than simply as the basic surveying, documentation/conservation devices they were designed to be. (Creative examples of using a scanner are still rare, but see 'The Falstone Border Shepherds Show 2013' by ScanLAB Projects, (ScanLAB Projects, 2014). Scanning Falstone Show was commissioned by Kielder Art & Architecture with support from Arts Council England, Paul Hamlyn Foundation and Queen's Hall Arts. Another, earlier example of the creative use of a scanner was a 1999 exhibition of Dalton's students' work at the Architectural Association, entitled 'Corporeal Space', which examined issues of corporeality and identity in virtual environments, and the body as a spatial object, through the medium of a 3D, full-body scanner.)

In terms of future work, what else might be possible to do with a 3D scanner? First, the technique we are showing here reduces a 3D point cloud to a mere 2D polygonal slice through the data set. In other words we are throwing away the largest proportion of data, especially those that relate to the third-dimension. Clearly the production of 2D isovist slices (of any plane, including sectional) is only the first stage of analyzing such a data set. One of the problems, however, is how to turn the 3D point cloud data into any meaningful representation: simply displaying the raw point cloud data is extremely hard to 'read' or drawing inferences/interpretations from. Having worked with the 3D scanning data, the authors of this paper felt that the issue of how to represent 3D isovist data was such a pertinent problem that they have written an accompanying paper (Dalton and Dalton, 2015) on this topic, in which they 'tested' a number of new 3D isovist representations that can be derived from the point cloud data. After testing these new 3D representations, two emerged as having the potential for further development. The first, the Contour Isovist employs the same algorithm designed for this study, but by incrementally moving the centroid in the z-plane, it effectively cuts a series of slices, at different heights, through the isovist field. These are then collapsed onto a single, contoured representation. Another representation that performed well under test conditions was the Tri-planar Isovist, in which the eye-height plan-based isovist is shown in conjunction with two sectional isovists, generated from the same point. Again, the algorithm developed for this study, permits an easy and straightforward generation of sectional isovists, hence permitting the production of Tri-planar Isovists. Further work is therefore needed in order to try these new representations on a number of real-world, spatially complex buildings in order to determine their efficacy. It is possible that through working with 3D point cloud data sets in this manner, further new methods of representation might also be developed. See the paper on 3D isovist representation for a fuller discussion of this issue and further work emerging from it (*ibid*).

A second area would be to process all the point cloud data in order to calculate a volumetric isovist. For example a three-dimensional volumes could be calculated rather than a two-dimensional isovist areas. Similarly a three-dimensional surface area could be computed. Initial results from the Lancaster University Campus Display Study suggest that the measures of 3D isovist volume and isovist surface are providing reliable correlations with behavioral data (Dalton et al., 2015). Currently three-dimensional isovists in real buildings are hard to compute as the necessary three-dimensional information is typically unavailable. One common naïve complaint by those new to the field of space syntax is the lack of analysis in the third dimension. Use of scanning data may permit the field to advance in this area.

A third area of future work is to more fully utilize all aspects of the scanned data. Sharp-eyed readers might have noticed that in the example raw-data entry illustrated at the beginning of this paper, additional information on the colour (expressed as an RGB value) of each point in the cloud is automatically captured at the same time as its location is recorded. At the moment we are discarding this data, but this has the potential to provide extremely useful information about the experience of being in a space. It is well known that colour has an effect on many aspects of behavior and performance from recovery rates in healthcare settings to overall mood and worker productivity (Cooper et al., 1989; Carpman et al., 1993; Kwallek et al., 1988; Yildirim et al., 2007). How might we incorporate isovist colour into our analysis and/or adapt our representations to include colour information?

A fourth area of future work is to continue to use the methods shown in this paper, for the analysis of displays/signage/retail shop fronts etc. i.e. the cardioid-weighted isovist polygon. Initial research in the Lancaster University Campus Display Study (Dalton et al., 2015) have shown that this weighted isovist approach marginally improves correlations, but again, further work needs to be done in this area.

Related to the area of signage and displays are isovists in front of/and through mirrors. If you scan near a mirror the laser beam bounces back. This produces a 'ghost mirror' set of points in the world, but at a lower intensity value than the rest. This effect is normally very easy to spot and typically people use the registration software (bundled with the scanner) to delete any unwanted points such as these non-existent mirror worlds. However, there is no reason why these mirror worlds should not be produced deliberately, if the 'world as seen through the mirror' is the focus of the research. For example, in Psarra's study of the Soane Museum, she examines the effect of the placement and orientation of mirrors on the curated architectural experience (Psarra, 2009). This is a study that could easily be reproduced using 3D scanners placed in front of the mirrors to generate not only 3D isovists, rather than a 2D isovist, but also produce the 'ghost mirror' 3D isovist at the same time. Again, this is an area in which future research could be conducted. In conclusion, the use of 3D scanners has the potential to open up many new avenues for research within the space syntax field, as well as requiring new and innovative means of representation.

References

- Benedikt, M. L. (1979). To take hold of space: isovists and isovist fields. *Environment and Planning B*, 6(1), 47-65.
- Böhler, W. and Marbs, A. (2002). 3D Scanning Instruments. *Proceedings of CIPA WG6 Scanning for Cultural Heritage Recording*, September 1–2, Corfu, Greece
- Carpman, J. R., & Grant, M. A. (1993). Design that cares: Planning health facilities for patients and visitors (2nd ed.). *Chicago: American Hospital Publishing*.
- Cooper, B., Mohide, A., & Gilbert, S. (1989). Testing the use of color in a long-term care setting. *Dimensions in Health Service*, 66, pp. 22, 24-26.
- Dalton, R.C. (1999). *Corporeal Space: corporeality and identity in virtual environments and the body as a spatial object*. The Architectural Association Bar Gallery, May 1999.
- Dalton, R.C. (2005). Space syntax and spatial cognition. *World Architecture*, 185 (41-47), 107-111.
- Dalton, N.S., Marshall, P., Dalton, R.C., Peverett, I., Clinch, S. and Davies, N. (2015). Three Dimensional Isovists for the Study of Public Displays. *Proceedings of 10th International Space Syntax Conference*, London, 2015.
- Dalton, R.C. and Dalton, N. (2015). The Problem of Representation of 3D Isovists. *Proceedings of 10th International Space Syntax Conference*, London, 2015.
- Dalton, N.S.C. (2015). Nebula. Computer software. Vers. 1.0.
- Kwallek, N., Lewis, C. M., & Robbins, A. S. (1988). Effects of Office Interior Color on Workers' Mood and Productivity. *Perceptual and Motor Skills*, 66(1), 123-128.
- Psarra, S. (2009). *Architecture and narrative: the formation of space and cultural meaning*. Routledge.
- ScanLAB Projects (2014). *Scanning Falstone Show 2013*. Gallery One, The Queens Hall, Hexham. 18 January - 1 March 2014.
- Tandy, C.R.V. (1967). The isovist method of landscape survey. In Murray, A. (ed.) *Symposium: Methods of Landscape Analysis*. May 1967, London, England, Landscape Research Group, 9–10.
- Xie, H., Filippidis, L., Gwynne, S., Galea, E. R., Blackshields, D., & Lawrence, P. J. (2007). Signage legibility distances as a function of observation angle. *Journal of fire protection engineering*, 17(1), 41-64.
- Xie, H., Filippidis, L., Galea, E. R., Gwynne, S., Blackshields, D., & Lawrence, P. J. (2007). Experimental study and theoretical analysis of signage legibility distances as a function of observation angle. *In Pedestrian and Evacuation Dynamics 2005* (pp. 131-143). Springer Berlin Heidelberg.
- Yildirim, K., Akalin-Baskaya, A., & Hidayetoglu, M. L. (2007). Effects of indoor color on mood and cognitive performance. *Building and Environment*, 42(9), 3233-3240.