Where's My Pixel? Multi-view Reconstruction of Smart LED Displays

Carl Lewis Angie Chandler Joe Finney
School of Computing and Communications, InfoLab21
Lancaster University
Lancaster, UK
Email: {carl.lewis, angie, joe}@comp.lancs.ac.uk

***Abstract*—The ubiquity and proliferation of digital imaging devices and computational power enable the use of computer vision in a variety of ubiquitous applications that previously would have been impractical. This paper presents a new such application domain called emergent displays (a type of actuator network), and goes on to describe its use of computer vision as a means of simultaneous localisation of large numbers of nodes. The effectiveness of a state of the art computer vision tool is analysed against this application with quantitative and qualitative results, and then put into context against further more general ubiquitous applications.**

***Keywords*-Smart pixel; Vision-based localization; Visual communication; Machine vision; Pervasive computing.**

I. INTRODUCTION

Public display technologies are now commonplace. Applications ranging from commercial advertising to digital signage have driven their deployment on a massive scale, with over 709,000 devices installed in North America in 2008 alone. Most deployments are based around off-the-shelf, inexpensive LCD or plasma technology measuring less than one metre in diameter. Larger displays are also popular in high profile locations, with flat-panel LED displays that measure tens of metres across. More recently, a new classification of public display technology has been proposed—the Emergent Display, a visual actuator network [1].

Unlike traditional computer display technologies that are formed on rectangular two-dimensional surfaces, emergent displays envision every pixel in a display being an intelligent, self-organizing, independent computational device that can be placed anywhere in three dimensions, allowing them to organically form displays to suit any environment. The ultimate vision of an emergent display could be considered a 'spray-on' display surface, where miniature pixels can be dynamically painted onto any surface, and self-organize to form a coherent display. Emergent displays are normally characterized by:

- A large number (typically thousands or even millions) of small, inexpensive, intelligent pixels that are dynamically deployed in an ad hoc fashion into an environment. Deployments can be either two or three dimensional, but are typified by pixels wrapped around the surfaces of large physical objects, such as public buildings.
- A low infrastructure computer network (either wired or wireless) that allows communication between the pixels.
- Irregular and unpredictable display geometry and densities. The very nature of these displays means that the overall shape and density of the display is also defined by the ad hoc deployment process, and can vary even within a single display.
- A localization technique that can locate and identify the pixels in 3D space after deployment.
- A rendering engine that can translate graphical content into network commands that control the pixels in real time.

Although still an area of active research, there have been several serious research attempts to develop prototype emergent displays. Whilst a thorough review is beyond the scope of this paper, examples include the Urban Pixels [2], The Particle Display System [3], LumiNet [4] and the Firefly project [1]. These projects have all adopted different hardware designs and approaches to the architecture and networking of pixels, but share the common challenges of emergent displays listed above. All of these prototypes utilize *light-emitting diodes* (LEDs) as a display component, due to their low cost and high modulation bandwidth. This paper focuses on the use of LEDs as a means to fulfil the localization requirement for emergent displays.

A significant volume of research has been undertaken in recent years (primarily in the wireless sensor network field), that investigates techniques for the localization of small devices, including approaches based on GPS (Global Positioning System), RF (Radio Frequency)/ultrasound signal strength, time of flight and/or angle of arrival [5]. However, such approaches are not well suited to the domain of emergent displays. GPS or RF localization solutions would require unacceptable levels of complexity and cost on each pixel, as they imply the need for expensive radio receivers and still yield relatively poor levels of accuracy—published results indicate tens of centimetres in the common case.

A far more suitable approach to satisfying all the requirements listed above is to use *multiple camera viewpoints with 3D reconstruction techniques* to generate the location information, particularly given the proliferation of CCD cameras and the increase in use of cameras for emergent displays, sensor nets and similar applications. Multiple camera viewpoints are already utilised for localisation in projects such as PhotoTourism [6], SenseCam [7], and the well-known Kinect platform [8] and in the past, multi-viewpoint techniques involving LEDs have been used to localize cameras.

For instance, in [9], LEDs were added to a building and some were located using a laser surveying system to form a control set. A high-speed camera took photographs of the LEDs from multiple viewpoints, while the LEDs transmitted a unique identifier to allow correspondences to be found. The camera properties were then worked out using the control LEDs as a calibration pattern. Once the camera properties were known, LEDs or other features could be triangulated. Similarly, in [10], known-position LEDs transmit location information which is picked up by CCD cameras mounted on mobile nodes. The camera's position, and hence also the node's, is then determined. Emergent displays on the other hand, must for reasons of practicality locate the pixels in the display *without any assumed reference points or calibration patterns*.

This paper presents a new technique for constructing 3D models of the relative locations of LED light sources within an emergent display. This method is notable because it does not require the use of reference points or calibration patterns, unlike similar techniques. This technique combines a feature detection algorithm, which uses *visible-light communication* (VLC) based on *on-off keying* (OOK) of LED light sources, with an existing state-of-the-art multi-view reconstruction application (Bundler). This paper presents a new methodology for comparing computed location models to a control model and applies this methodology to provide an experimental evaluation of the new localization technique using the Firefly system as a research vehicle. Results are presented in terms of the number of viewpoints required to produce a location model, the proportion of attempts which result in a valid model, the number of LED lights successfully identified, and the accuracy of the generated model. A number of heuristics are then suggested with which to optimize the localization process. We conclude by suggesting areas for future work, as well as considering how this new form of localization may be applied more broadly within the field of ubiquitous computing. Firefly forms the basis for experimental evaluation in this paper, so we first provide a short conceptual overview of the system to aid the reader's understanding of its concepts.

II. Firefly: An emergent display prototype

Firefly is an emergent display prototype that enables tens of thousands of pixels to be dynamically deployed in displays measuring tens of metres across, and is targeted at providing support for displays embedded into building architecture, as exemplified in the 3000-pixel field trial deployment illustrated in Figure 1.

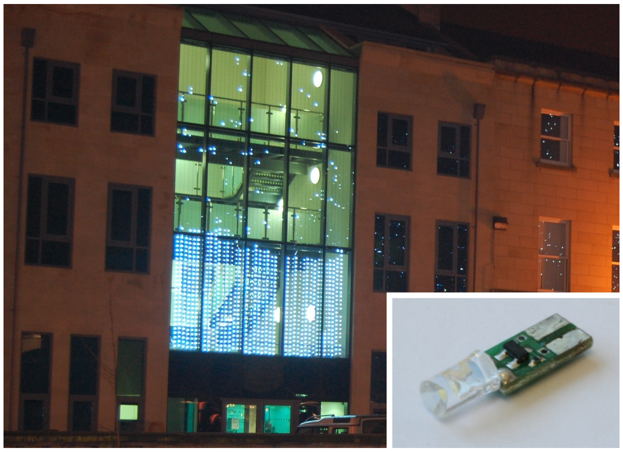


Figure 1. Citylab display and Firefly pixel

Central to the concept of Firefly is the Firefly pixel, also depicted in Figure 1. Firefly pixels consist of a microprocessor, a single-colour or RGB LED, and a small number of inexpensive discrete electronic components (transistors, capacitors, etc.). These pixels measure 6 mm × 20 mm, and can be constructed for less than $1 in component costs.

Firefly pixels are wired together using a simple two-wire bus. Up to 240 grey-scale or 80 full-colour pixels can be connected on a single, two-wire Firefly string measuring up to 50 m in length. One wire serves as a ground, while the other carries both power and data to the pixels, at a rate of approximately 80 kbps. This is sufficient to allow real-time control of the pixels at 30 frames per second. Firefly pixels also self-configure a display-wide unique 24-bit identifier.

Once manufactured, Firefly pixels can be placed (typically hung or wrapped) in any position or topology, much like a string of common Christmas lights, but on a larger scale. Note that, unlike a conventional screen display, a pixel's location bears no relation to its address. Therefore, a mapping from address to location must be determined. In Firefly this is achieved through applied computer vision techniques in two stages: 2D Imaging and 3D Reconstruction.

A. 2D Imaging

Firefly pixels are localized using a collection of 2D views. Each of these views is generated from a series of 26 images taken from a single viewpoint (typically using an SLR camera):

- One frame with all pixels off; this is used as a reference frame against which other frames are compared.
- One frame with all pixels on; this is used to pre-locate all pixels to speed processing of the remaining frames.
- Twenty-four frames to represent the identifier of the Firefly pixel, encoded using OOK in Big-Endian order.

During this process both pixels and camera are synchronously controlled, enabling every pixel in view of the camera to simultaneously encode one bit of their identifier in each image.

Once a complete set of images has been gathered for a given viewpoint, the identifier and 2D image location of every Firefly pixel visible from that viewpoint can be established. This is achieved by comparing every individual image to the reference frame before using a simple threshold filter to determine whether a given Firefly pixel was on or off in each image. The identifier of every Firefly pixel can then be simply recovered. In order to improve resilience, forward error correction codes (such as a Hamming code) can also be used at this stage. The final pixel location is taken to be the average of the centre points of that pixel over each frame in which the pixel is illuminated. An example of how two such viewpoints might be generated from a simple display is given in Figure 2.

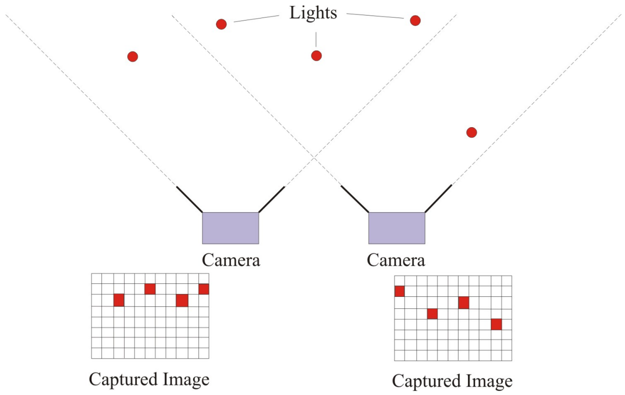


Figure 2. Example of two viewpoints of a simple display

A Firefly pixel's identifier, when combined with the 2D location coordinates, enables the generation of a *view*—a list which matches identifiers with 2D locations from a given viewpoint. An arbitrary number of views can be generated for a Firefly display. These are then composed in a second stage, where a full 3D model can be reconstructed.

B. 3D Reconstruction

In order to reconstruct a 3D scene, we utilize standard photogrammetry techniques. Photogrammetry is a special case of visible-light localisation, in which geometric information, in particular a 3D model, is extracted from multiple 2D images of a scene. A thorough description of the mathematics behind photogrammetry (epipolar geometry), is given by Hartley and Zisserman [11].

In a typical photogrammetric application (e.g., Photosynth [12]), photographs from several viewpoints are passed to a feature detection algorithm, such as SIFT [13]. This algorithm finds the locations of distinctive features in an image and matches them to corresponding features in other images. For Firefly, however, this is unnecessary, as feature correspondences are identified through the creation of views (maps from pixel IDs to 2D locations), as described in the previous section. These features may then be entered into a *bundle-adjustment* algorithm, which estimates a model of the 3D positions of the features.

To implement this design, we chose to adopt the widely used photogrammetric application Bundler [6]. Bundler takes a list of features in each view, a list of feature correspondences between views, the focal length (in pixels) of the camera, and produces a 3D model of all the Firefly pixels, which we call a scene. In contrast to previous techniques for estimating LED positions, Bundler does not require that the camera positions are known beforehand.

C. Discussion

This section has described the conceptual operation of Firefly—an emergent display prototype—and its reliance upon existing computer vision techniques for localization. From this, a number of questions become apparent:

- How many views are required in typical Firefly deployments to produce an accurate and complete 3D model?
- How accurate a model can be produced, as compared to a control case?
- What heuristics can be applied in the field to improve the accuracy and completeness of a final Firefly scene?

In order to address these questions, the following section undertakes a quantitative experimental analysis of the Firefly prototype.

III. Evaluation

A. Experimental Procedure

The results presented in this paper investigate the effectiveness of the procedure used to localise Firefly pixels within a display, both in terms of the initial 2D views collected with the cameras and the generation of the full 3D scene using Bundler. The goal of this experiment was to see how many views are required to generate a good 3D model, how much variation there is in the models produced by different combinations of views, and whether there are any simple heuristics for identifying poor views prior to their use in scene generation. These results are then put into perspective against other applications which may require this type of positioning system, such as sensor networks or robotics.

Configurations

The data presented in this section were generated through a series of experiments on two distinct displays: a 2D Firefly display with LEDs in strict grid formation and a 3D display formed from a wrapped cylinder. During our experiments each of these displays was positioned with sufficient space around it to enable a wide variety of views, including both views close to the display and views at a distance, and views were obtained at a wide variety of orientations, heights and angles—as allowed by the space available.

Fifteen distinct views were taken of the 2D display, with a further forty-five views of the 3D wrapped surface. 3D Bundler scenes were then generated using randomly selected combinations of these views, with between two and 44 views selected for each of the models generated. This process was repeated fifty times, with the accuracy and completeness averaged over each set of fifty scenes to give a final representative result for each quantity of views.

Control Case

The analyses of the Bundler generated scenes in this section include estimates of absolute error from the modelled point positions to the actual point positions. The actual point positions were determined by direct measurements of the displays combined with certain assumptions about the display configurations; these are described in more detail in the following sections. As even an ideal Bundler model may differ from the control model, it must first be transformed into the coordinate system of the measured model using a 'best-fit' similarity transform. This similarity transform recognises that the model may have an arbitrary origin, scale, and orientation, as these properties are impossible to determine without an absolute reference point. These transformations were computed using a probabilistic RANSAC-based algorithm, in which a small subsets of points are assumed to be inliers and used to compute transformations which closely match the measured model.

Metrics

From the control case, it was naturally possible to determine the overall *accuracy* of each scene generated as the mean error of each Bundler-modelled pixel location to the matching control measurement. This value was computed for each scene generated, and scenes were also marked as successful scenes and failed scenes. A scene was counted as failed if Bundler did not converge to a solution or if the accuracy was not within 10 cm.

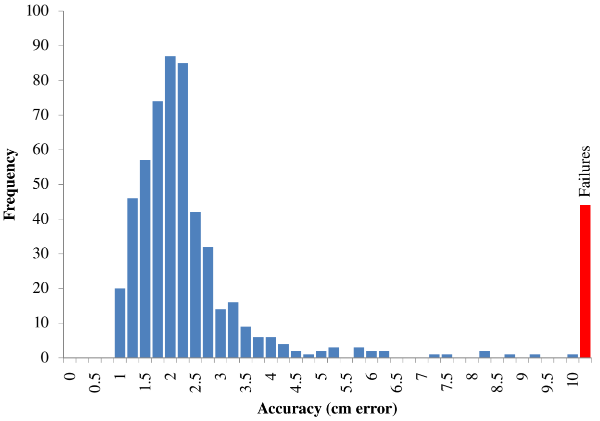


Figure 3. 2D grid histogram of accuracies

A histogram of the accuracies for the first display is shown in Figure 3, which demonstrates that scenes with mean error of greater than 10 cm are clear outliers. All experiments described in this paper exhibited similar distributions, so are not included here for conciseness. Finally, whilst we recognise that a typical accuracy of 2 cm is considerably worse than demonstrated in other documented photogrammetry applications [14], we attribute this to the resolution of Firefly pixels within the views, rather than the Bundler process itself. It is nonetheless sufficiently accurate for the application domain.

In addition to the accuracy, the *completeness* of each scene is measured as the proportion of pixels which are successfully modelled at all by Bundler, as Bundler omits pixels entirely if the data required to triangulate them are insufficient or contradictory.

B. *Flat Surface*

The first experimental display is a 2D grid, chosen for the ease with which an experimental control scene can be constructed, as well as the ability to visually spot errors in the Bundler model. This display consists of a fixed, wooden back plane on which pixels are placed in a regular, 32 × 12 grid at 5 cm intervals, providing an ideal known configuration for initial tests. The experimental control scene of this display assumes that the pixels are in a perfect, co-planar, evenly-spaced grid.

Fourteen views of this display were taken from a variety of distances and angles relative to the display. Randomly chosen subsets containing 2–12 views were used to generate scenes, for 550 total scenes. The success rate (proportion of scene-generation attempts which were successful), mean accuracy, and mean completeness of the scenes, grouped by the number of views used to generate them, are plotted in Figure 4.

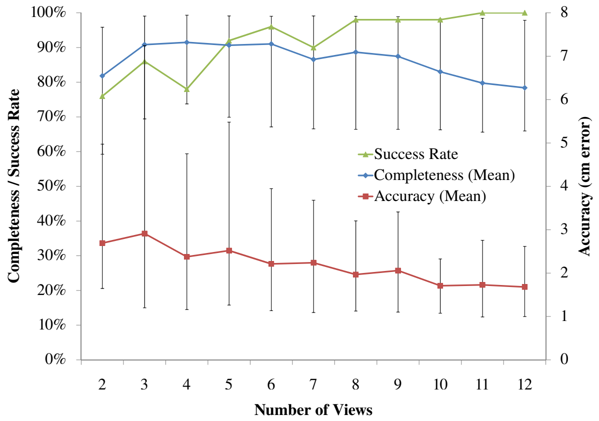


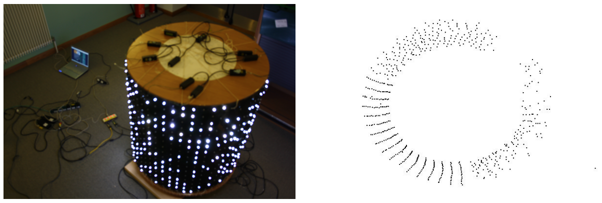
Figure 4. 2D grid display analysis (error bars represent the 5th and 95th percentiles)

The completeness is fairly steady with respect to the number of views. This is to be expected, as most views of this display contain all the pixels. However, the completeness does appear to decline slightly at the tail of the graph, though not conclusively; this is examined in more detail in the next experiment. Accuracy, on the other hand, improves slowly with the number of views, as does the success rate, until it levels off after 6–8 views.

C. Wrapped Surface

The second experiment uses a Firefly display constructed from a 3 m × 1 m fabric in a woven 5 mm grid, in which Firefly pixels are placed. This fabric display is useful as it is fairly easy to generate a control model by counting squares in the weave, yet it can also be wrapped around a surface to produce a 3D display. In total nearly 800 pixels were used, with 320 measured as control points. Initially, this fabric was placed flat and analysed in order to confirm our previous results for the 2D grid. The results achieved were consistent with those of the 2D grid and therefore are not reproduced here.

The fabric was wrapped around a 1 m diameter cylinder to form a 3D display. A cylinder was chosen as it accurately represents the distribution and obfuscation characteristics of emergent displays, yet is quite easy to determine 3D control positions for the pixels from 2D positions in the weave. In order to do so, it was assumed that the cylinder was stretched perfectly tautly around the cylinder. The cylinder and one resulting Bundler model are shown in Figure 5(a) and Figure 5(b), respectively.



(a) Cylinder (b) Scene produced (top view)

Figure 5. Cylinder experiment

As with the 2D grid display, views were taken from a variety of distances and angles. However, as it was impossible to see all pixels from any one viewpoint, many more views were taken (forty-six in total). These were selected in random combinations (fifty combinations each for 2 to 44 views), to form a total of 2150 scenes. These were analysed in the same way as the for 2D grid display. Figure 6 shows the results.

Compared to the 2D grid display, more views were necessary to achieve a good completeness, as not all pixels were visible in each view. What is surprising is that the completeness and accuracy appear to *decline* slowly but steadily after a peak between 12 and 16 views. It is not clear why they should decline in this way, though it may be hypothesized that this is due to a system within the Bundler process becoming significantly overdetermined at this point. The next section discusses heuristics by which good viewpoints of a display may be chosen.

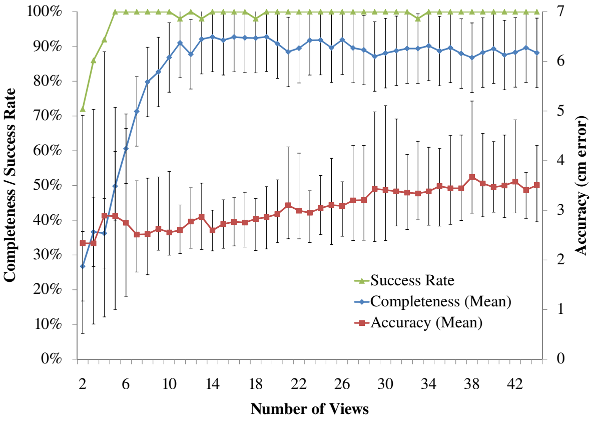


Figure 6. Cylinder display analysis (error bars represent the 5th and 95th percentiles)

D. Heuristics

The previous sections examined how success rate, accuracy, and completeness of a scene change with the number of views used to construct it. The results were generated by random combinations selected from a wide range of viewpoints. This section examines whether any simple heuristics exist for choosing 'good' viewpoints in the field which perform better than selecting at random.

Table I
FAILURE RATES OF VIEW CREATION FOR 2D GRID

View	Pixels Detected (out of 444)	Duplicates	False positives	False negatives
0	439	0	0	5
1	431	0	1	13
2	428	0	0	16
3	376	0	15	68
4	424	0	0	20
5	438	0	1	6
6	438	0	0	6
7	440	0	1	4
8	435	0	3	9
9	441	0	2	3
10	425	0	8	19
11	370	0	35	74
12	325	0	29	119
13	416	0	6	28

Initially, we consider determination of 'good' views simply by observation of the characteristics of the individual views themselves. For example, taking the 14 views of the 2D grid, shown in Table I, there are several types of failure which may be identified from examination of the view alone. These include: duplicates (two features with the same ID), false positives (features with IDs which do not correspond to real pixels), and false negatives (pixels with no corresponding features in the view). Duplicates are non-existent in this experiment, while false positives are rare and correlate fairly strongly with false negatives. For these reason, completeness was considered as a possible measure of the ‘goodness’ of a view.

Scenes constructed only from nearly complete views showed encouragingly high completeness themselves. However, they also showed greater inaccuracy and lower success rate than scenes which were constructed from an equal number of views, some of which were less complete. It seems likely that these inaccuracies are a result of a lack in viewpoint variation; views of the 2D grid which are most complete are likely to be taken a short distance from the display at an angle nearly perpendicular to it. Therefore, excluding less complete views results in less information for triangulation.

Testing this idea more thoroughly, 3 out of 14 of the 2D grid views were classified as oblique. Scenes were generated from 6 views randomly selected from the 14 2D grid views, and analysed based on the number of oblique views each contained. It was found that accuracy tended to improve linearly with the number of oblique views used (up to 3), while completeness tended to decline linearly instead. These results are shown in Figure 7.

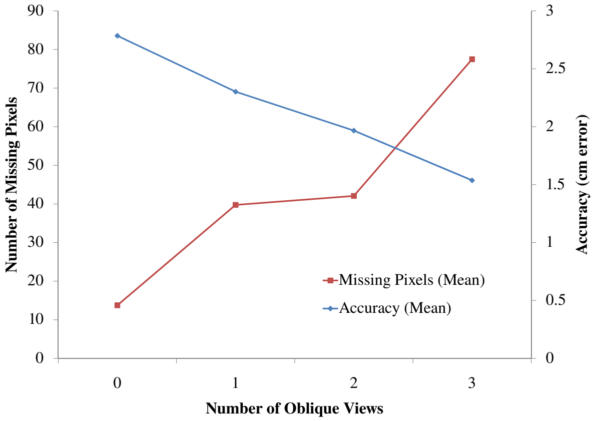


Figure 7. Effects of oblique views on accuracy and completeness

Overall, this suggests that a variety of viewpoints should be selected when localizing a 2D display, in order to achieve a good balance between completeness and accuracy. More perpendicular or oblique shots may be used depending on whether completeness or accuracy is more important for the application, respectively. The results from Section III-B suggest that 6–8 views will be sufficient in most cases to produce a successful scene.

Extending these heuristics to the 3D wrapped cylinder, it is of note that every view is in essence both face on and oblique to some of the pixels. This may contribute to the high success rate achieved by the cylinder relative to the 2D grid, although it is also suspected that 3D scenes will naturally perform better due to greater variation being available to aid camera reconstruction.

Furthermore, we also consider the effects of views containing reflections on the generated scene. An experimental analysis in which scenes were generated from views that intentionally contained significant reflections exhibited a consistent and notable drop in completeness without a significant effect on accuracy (detailed results are omitted here for conciseness, but are available on request). This allows us to conclude that Bundler performs well in detecting reflections as outliers and removing them from the final scene (thus reducing completeness but maintaining accuracy). This also means that the view with a reflection does not contribute towards the localisation of the reflected pixels, and therefore sufficient alternative data on each of these pixels must be available in the remaining views to maintain completeness when the scene is generated.

In the previous section, the minimum number of views required to effectively (in most cases) generate a given scene was discussed, suggesting approximately 6–8 views would typically be effective for a 2D scene and 12–14 for a 3D wrapped surface. However, whilst these values provide good guidelines to the suggested number of views, without some understanding of the characteristics of views that would produce a good scene, any number of views could generate an unusable scene. Based on experiments described in this subsection, the following (largely intuitive) heuristics can be reached:

H1 Include more than one view with good completeness.

H2 A small number (up to 50%) of oblique views will increase accuracy.

H3 Views should be as diverse as possible.

H4 Avoid reflections if possible, but if views containing reflections are added, ensure that each reflected pixel is contained in additional views.

IV. Conclusion and Future Work

This paper has discussed the requirements for emergent displays (a new application domain requiring multi-view reconstruction techniques) and documented a preliminary experimental evaluation of the performance of Bundler (a state of the art tool in 3D reconstruction) in supporting that domain. More specifically, two experimental displays each containing several hundred pixels were modelled using a Bundler-based technique. The success rate, completeness, and accuracy of over 2500 models were analysed with respect to the number of viewpoints used to generate them, and heuristics for choosing good viewpoints were developed and presented.

Our results indicate that, on the whole, Bundler does operate sufficiently well to support this domain. Typically a minimum of fourteen views are required to accurately generate a 3D model of an emergent display, with 90% completeness. However, although Bundler also exhibited resilience to dealing with reflections, it achieves this through aggressively treating reflections as outliers, resulting in 3D models that prefer accuracy over completeness. Whilst this is highly beneficial for the photo-tourism application domain Bundler was originally designed for, emergent displays (alongside other sensor and actuator nets) have different requirements. Here, pinpoint accuracy of pixels is of relatively little importance, whereas maximizing the number of usable pixels in the display is of prime concern.

In terms of alternative applications of this technique, relative to many commonly used methods of localization, such as RF signal strength and angle of arrival, GPS, or ultrasound, this multi-view computer vision technique performs well. RF methods give accuracies of a few metres, which is simply insufficient for many ubiquitous applications, including emergent displays. Differential GPS and ultrasound methods can achieve accuracies comparable with the computer vision technique described here, but the per-node cost technologies is prohibitive. Therefore, this visible-light localization technique would seem an ideal alternative for many ubiquitous systems, due to the low per-node cost (in terms of physical size, memory footprint, and processing, as well as component cost), its high accuracy, and the ubiquity of existing infrastructure in the form of web cams.

The current goal of the technique described was to localize LEDs in a static display. In the future, we intend to look at whether a similar technique could be applied to the localization of other devices using LED markers. This would provide a low-cost mechanism for tracking objects in, for instance, robotics or ambient workplace applications. In order to reduce the infrastructure requirements of this technique further, error correction and multiple-access techniques may be investigated so that lower-resolution cameras (in particular, webcams), may be used.

Other future works will focus on the refinement of the multi-camera processing technique itself. In practical terms, this will include a closed-loop algorithm to determine which pixels are poorly located at run time, improving the likelihood of pixels subsequently being well located by the multi-view algorithm. In addition to this, there will be large-scale field trials to provide further insight into our heuristics. However, given the primarily empirical nature of this work thus far, we also intend to relate the heuristics back to computer vision theory and investigate the causes of the peak and decline effect observed in the 3D cylinder experiment.

References

[1] A. Chandler, J. Finney, C. Lewis, and A. Dix, "Toward emergent technology for blended public displays," in *Ubicomp '09: Proceedings of the 11th international conference on Ubiquitous computing*. New York, NY, USA: ACM, 2009, pp. 101–104.

[2] S. Seitinger, D. S. Perry, and W. J. Mitchell, "Urban pixels: painting the city with light," in *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*. New York, NY, USA: ACM, 2009, pp. 839–848.

[3] M. Sato, "Particle display system: a real world display with physically distributable pixels," in *CHI '08: CHI '08 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2008, pp. 3771–3776.

[4] R. Bohne, "Luminet: An organic, interactive, illumination network," Master's thesis, Aachen University, 2009. [Online]. Available: http://hci.rwth-aachen.de/materials/publications/bohne2009a.pdf

[5] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008.

[6] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: exploring photo collections in 3d," in *ACM SIGGRAPH 2006 Papers*, ser. SIGGRAPH '06. New York, NY, USA: ACM, 2006, pp. 835–846.

[7] C. O. Conaire, M. Blighe, and N. E. O'Connor, "Sensecam image localisation using hierarchical surf trees," in *Proceedings of the 15th International Multimedia Modeling Conference on Advances in Multimedia Modeling*, ser. MMM '09. Berlin, Heidelberg: Springer-Verlag, 2008, pp. 15–26.

[8] Microsoft, "Kinect," accessed June, 2011. [Online]. Available: http://www.xbox.com/kinect

[9] H. Uchiyama, M. Yoshino, H. Saito, M. Nakagawa, S. Haruyama, T. Kakehashi, and N. Nagamoto, "Photogrammetric system using visible light communication," in *Proc. 34th Annual Conf. of IEEE Industrial Electronics IECON 2008*, 2008, pp. 1771–1776.

[10] M. Yoshino, S. Haruyama, and M. Nakagawa, "High-accuracy positioning system using visible led lights and image sensor," in *Proc. IEEE Radio and Wireless Symp*, 2008, pp. 439–442.

[11] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision, Second Edition*. Cambridge University Press, 2003.

[12] Microsoft, "Photosynth," accessed July, 2011. [Online]. Available: http://photosynth.net/

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004, 10.1023/B:VISI.0000029664.99615.94.

[14] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. Seitz, "Multi-view stereo for community photo collections," in *Proc. ICCV*. Citeseer, 2007, pp. 1–8.