24th International Conference Image and Vision Computing New Zealand (IVCNZ 2009)

Surface Projection for Mixed Pixel Restoration

Robert L. Larkins[1], Michael J. Cree, Adrian A. Dorrington, John P. Godbaz
Department of Engineering, University of Waikato
Hamilton, New Zealand
[1]Email: RLL6@students.waikato.ac.nz

***Abstract*—Amplitude modulated full-field range-imagers are measurement devices that determine the range to an object simultaneously for each pixel in the scene, but due to the nature of this operation, they commonly suffer from the significant problem of mixed pixels. Once mixed pixels are identified a common procedure is to remove them from the scene; this solution is not ideal as the captured point cloud may become damaged. This paper introduces an alternative approach, in which mixed pixels are projected onto the surface that they should belong. This is achieved by breaking the area around an identified mixed pixel into two classes. A parametric surface is then fitted to the class closest to the mixed pixel, with this mixed pixel then being project onto this surface. The restoration procedure was tested using twelve simulated scenes designed to determine its accuracy and robustness. For these simulated scenes, 93% of the mixed pixels were restored to the surface to which they belong. This mixed pixel restoration process is shown to be accurate and robust for both simulated and real world scenes, thus provides a reliable alternative to removing mixed pixels that can be easily adapted to any mixed pixel detection algorithm.**

I. Introduction

Full-field range-imaging cameras provide the simultaneous acquisition of range data at each pixel in an image. Full-field amplitude modulated continuous wave (AMCW) lidar systems [1] achieve this by illuminating a scene with amplitude modulated light, and determine the range by measuring the phase offset between the received light and the transmitted light. The range to an object at each pixel of the camera is determined from the phase difference and knowledge of the speed of light.

The mixed pixel phenomenon occurs in AMCW systems when the light that a pixel captures is contaminated by multiple reflections from the scene. The range calculated for a mixed pixel under the assumption of a single return can erroneously be anything from zero up to the ambiguity distance [2]. The details of mixed pixels are described in greater detail in Section II below.

There has been little research into identifying mixed pixels in a range image despite mixed pixels being a significant source of error. The most common approach reported in literature is to identify mixed pixels from a produced point cloud, as the characteristics of the mixed pixel in a point cloud tend to differ from those that are not mixed. The ability of three mixed pixel identification algorithms, namely the normal-angle filter, edge-length filter and the cone algorithm, are investigated by Tang et al. [3]. They found that none of these three algorithms performed exceptionally well, with the normal-angle method performing the best of the three. Two simplistic methods of dealing with mixed pixels, namely, identifying isolated points in three dimensional coordinate space and median filtering were mentioned by Hebert and Krotkov [2], but were not further elaborated upon. Alternative approaches of detecting and even correcting mixed pixels have been investigated; these include decomposing the mixed pixels into their distinct components [4], detecting discontinuities in the returned signal amplitude [5], and deconvolving the returned signal, to identify the range and intensity of all signal returns seen by each pixel [6].

Once a mixed pixel is identified the general approach is to remove it outright. This deals with the problem of the mixed pixel, but has potential to introduce other errors, such as distorting object edges and creating holes in surfaces due to falsely detected mixed pixels. The method presented in this paper is designed to restore the mixed pixels to the surface that they belong to and has the advantage of not removing points. This paper specifically focusses on the restoration of mixed pixels, but the presented technique can also help reposition points affected by noise by utilising the locations of their neighbours.

In this paper mixed pixel restoration using surface projection is achieved via a series of steps. The first step of mixed pixel identification begins once a point cloud has been produced from a range imaging camera, and is described in Section II. Section III details how Otsu thresholding is used to segment the neighbours of a mixed pixel into two clusters. A parametric surface is fit to each class and each point is projected onto the closest surface. This surface modelling and projection is described in Section IV. Testing of the mixed pixel restoration is carried out by simulating a set of scenes that determine the precision of this process; this is detailed in Section V.

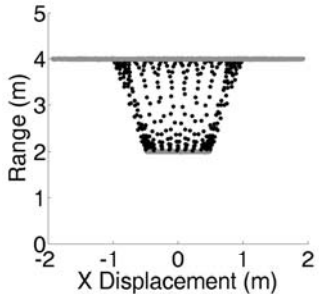
II. Mixed Pixels and Their Identification

Mixed pixels are a significant problem prevalent in full-field AMCW lidar systems which use modulated light to illuminate a set of objects in a scene. Each pixel of the camera sensor captures a piece of the returned light, and by determining the phase of the captured light with respect to the reference modulation the distance to the area viewed by the pixel is determined. Mixed pixels occur when the sensor picks up light that has been reflected back from two or more objects in the scene. This occurs, for example, when a single pixel sees the boundary of two adjoining objects at different ranges.

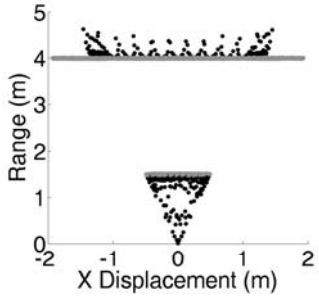
The capture of a scene produces points in a spherical coordinate system centred on the camera, thus the pixel sees

978-1-4244-4698-8/09/$25.00 ©2009 IEEE

light received from a fixed solid angle in a direction projected through the focal point. It is convenient to use the radial distance from the pixel rather than the range. This explains the cone like form of the captured points for the scene shown in Figure 1. Generally, mixed pixels are spread between the objects in a scene, as shown in Figure 1(a), and are due to the previously described problem of a single pixel integrating light measured from two surfaces. Nevertheless there are cases where the mixed pixels extend backwards from the farthest object towards the ambiguity distance, or forward of the front object towards the origin, as shown in Figure 1(b).



(a) Mixed Pixels Between



(b) Mixed Pixels Eitherside

Figure 1. Reconstructed range of a scene consisting of two objects at different ranges showing that mixed pixels can occur between objects or either side of them (grey lines are the objects, black dots are the mixed pixels).

The phenomenon shown in Figure 1(b) is caused by the ambiguity distance λ, given by

$$\lambda = \frac{c}{2f}, \tag{1}$$

where f is the chosen modulation frequency and c is the speed of light. The ambiguity distance is the greatest unambiguous range that an object can be from the camera before the range cycles [7].

The manner in which the range is determined from the returned phase causes mixed pixels to appear in the shortest distance between two objects. If this shortest distance passes through the ambiguity range, then the mixed pixels extend towards 0 and λ as shown in Figure 1(b). A visual example of why the mixed pixels spread like this is shown in Figure 2. Here the distance between the front object O_1 and the back object O_2 is shortest when it passes through λ.

The mixed pixel identification method that is utilised henceforth is similar to the normal-angle filter [3], and was chosen

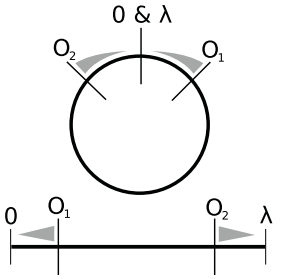


Figure 2. Mixed pixels, depicted by the grey triangles, occur in the shortest distance between two objects, in this case O_1 and O_2. If this distance crosses the ambiguity range λ, then the mixed pixels appear to extend to the scene boundaries.

over other algorithms as it was found to have the best overall performance. The normal-angle filter does have a drawback in that if a surface normal is close to being perpendicular to the optical axis of the camera, the number of false positives is liable to increase.

The algorithm works by taking the line between two points, $\boldsymbol{P_1}$ and $\boldsymbol{P_2}$, and determining if the angle θ between the normal of this line, N, and the line to the origin is greater than a specified threshold, T. If θ is greater than T, then both $\boldsymbol{P_1}$ and $\boldsymbol{P_2}$ are deemed to be mixed. Figure 3 shows how this is carried out. For this operation, N is centred at the point

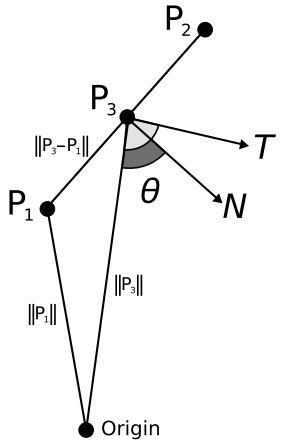


Figure 3. Two points, $\boldsymbol{P_1}$ and $\boldsymbol{P_2}$ are deemed to be mixed if the angle θ of the normal N is greater than a threshold T.

halfway between $\boldsymbol{P_1}$ and $\boldsymbol{P_2}$, but due to the law of cosines N is unnecessary for the calculation of θ, which is given by

$$\theta = \left| \frac{\pi}{2} - \cos^{-1}\left(\frac{a^2 - b^2 - (\|\boldsymbol{P_3} - \boldsymbol{P_1}\|)^2}{-2ab} \right) \right|, \tag{2}$$

where a is the maximum of $\|\boldsymbol{P_1}\|$ and $\|\boldsymbol{P_2}\|$, and b is the minimum. Setting a and b in this fashion ensures that N points in a direction back towards the origin instead of away.

A selected point is tested by first joining itself and its three bottom right neighbours together to produce four boundary lines. A fifth and final line is found as the shortest internal diagonal of the formed polygon. Figure 4 shows an example where $\boldsymbol{P}$ is the selected point and the five lines L_1 to L_5 are

identified. This process is repeated for every point in the grid

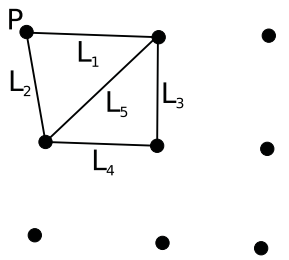


Figure 4. This figure shows how a set of lines are found when given a point $\boldsymbol{P}$ and its neighbours.

except for the last column and row as these points lack some or all of their bottom right neighbours. They are taken care of however by the process being carried out on the previous row or column.

The identified mixed points are subsequently restored by projecting them towards one of the two classes determined in the area around that point. The following section describes the manner in which these two classes are determined.

III. Class Determination

Having identified that a pixel $\boldsymbol{q}$ is mixed, the surface that $\boldsymbol{q}$ should be projected towards is determined from the square neighbourhood of pixels centred on $\boldsymbol{q}$, where this square has a side length of $2l + 1$. The mixed pixels in this neighbourhood, including $\boldsymbol{q}$, are then removed ensuring no bias when determining the two surfaces.

The Otsu threshold [8] is used to determine the two surfaces and is performed on the discretised radial distance r of each point calculated by $r = \|\boldsymbol{p}\|$, where $\boldsymbol{p} = [x, y, z]$ is the three dimensional Cartesian coordinates and is discretised to millimetre points.

The Otsu threshold identifies two distinct classes by iteratively testing each millimetre of the range up to the ambiguity distance as a threshold. The chosen threshold minimises the inter-class variance, where the two classes are the surfaces of the two objects that initially produced the mixed pixel. The class closest to the origin is labelled *class1*, while the more distant class is *class2*. The class that $\boldsymbol{q}$ belongs to is determined by

$$class = \begin{cases} class1 & \text{if } d_1 \leq d_2 \\ class2 & \text{if } d_1 > d_2, \end{cases} \tag{3}$$

where d_1 and d_2 are the distances of the point $\boldsymbol{q}$ to the medians m_1 and m_2 of *class1* and *class2*, respectively. The distance d_1 is given by

$$d_1 = \begin{cases} |r - m_1| & \text{if } r \leq m_2 \\ \lambda - r + m_1 & \text{if } r > m_2, \end{cases} \tag{4}$$

and d_2 is found in a similar fashion by

$$d_2 = \begin{cases} |r - m_2| & \text{if } r \geq m_1 \\ \lambda + r - m_2 & \text{if } r < m_1, \end{cases} \tag{5}$$

where the ambiguity λ is taken into account due to its affect on the placement of points.

Having determined which class the point belongs too, the next step is to construct a parametric surface that models the class, and to that we now turn attention.

IV. Restoration Through Surface Modelling

Parametric surfaces model a set of points in a manner that permits the area between and beyond these points to be described by a single function. We now describe a model for the surface that maps relative pixel coordinates, θ and φ to a radial distance from the camera. Figure 5 shows an example of how the points around $\boldsymbol{q}$ are indexed.

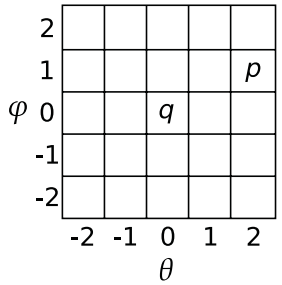


Figure 5. A grid showing how the points around $\boldsymbol{q}$ are indexed, where θ and φ for p would be set to 2 and 1 respectively.

The constructed parametric surface is expressed by

$$r'(\theta, \varphi) = \hat{\beta}_1\theta^2 + \hat{\beta}_2\varphi^2 + \hat{\beta}_3\theta\varphi + \hat{\beta}_4\theta + \hat{\beta}_5\varphi + \hat{\beta}_6, \tag{6}$$

where the vector $\hat{\beta}$ contains the six coefficients that describe the location and shape of the surface. These coefficients are found by

$$\hat{\beta} = (X^T X)^{-1} X^T R. \tag{7}$$

The matrix X is constructed from the n spherical coordinate points p_i where $i = \{1, 2, \ldots, n\}$ from the class $\boldsymbol{q}$ has been determined to belong too. These n points do not include any identified mixed pixels, as mixed pixels adversely bias the shape and position of the fitted surface. The structure of X is given by

$$X = \begin{vmatrix} \theta_1^2 & \varphi_1^2 & \theta_1\varphi_1 & \theta_1 & \varphi_1 & 1 \\ \theta_2^2 & \varphi_2^2 & \theta_2\varphi_2 & \theta_2 & \varphi_2 & 1 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \theta_n^2 & \varphi_n^2 & \theta_n\varphi_n & \theta_n & \varphi_n & 1 \end{vmatrix}, \tag{8}$$

while the the vector R is the radial distance to each of these points, namely

$$R = [r_1, r_2, \ldots, r_n]^T. \tag{9}$$

Having found the solution for vector $\hat{\beta}$, it can then be used in equation (6), but as we only require the radial distance to the modelled surface when θ and φ are 0, the new radial distance, r', for the point $\boldsymbol{q}$ is $\hat{\beta}_6$.

The new position for $\boldsymbol{q}$ is found as $\boldsymbol{q'} = \boldsymbol{q}t$ and lies on the line that passes through both $\boldsymbol{q}$ and the origin of the coordinate

system. The variable t given by

$$t = \sqrt{\frac{r'^2}{x^2 + y^2 + z^2}}, \tag{10}$$

controls where on this line $\boldsymbol{q'}$ is found, where x, y and z are the initial Cartesian coordinates of $\boldsymbol{q}$.

Following this process, a mixed pixel $\boldsymbol{q}$ is projected towards the nearest surface that has been modelled by a set of points that do not contain any identified mixed pixels. This method requires the Cartesian coordinates for each point and the ambiguity distance. The output is a new set of Cartesian coordinates for each identified mixed pixel, which places them with the class they have been determined to belong too.

V. Experiments and Results

Verifying the capability of this projection method cannot be easily carried out with real data as determining the true position of every return in the scene is difficult to perform. Therefore, the evaluation of the mixed pixel restoration is carried out by simulating a set of scenes that replicate a range of scenarios in which mixed pixels occur. From each scene two point clouds are produced: the first, labelled C_1, is the ideal version, containing no mixed pixels or noise, and the second point cloud, C_2, contains both the mixed pixels and noise that would normally occur with a range imaging camera.

The restoration procedure is performed on C_2, and is verified by comparing the radial distance of each point with its counterpart in C_1. The radial differences between these points are determined, allowing a measurement of accuracy to be produced. Because this method is designed to restore the mixed pixels, not the noise, the restored points are tested to ensure they lie within ± 15 mm of the reference points radial distance. There are however points that do not have the projection procedure applied to them as they are too close to the edge of the grid. The projection procedure requires a square neighbourhood around each point to be effective, this can only be applied to points that are at least a distance l from the edge. In these experiments $l = 6$ pixels, thus the side length of the square is $2 \times 6 + 1 = 13$ pixels. It is necessary to take l this high as many of the points in the square area are rejected as mixed or as belonging to the wrong surface, and only the remaining points are used for surface fitting.

A. Testing with Simulated Scenes

Testing of the mixed pixel restoration process was carried out by producing twelve simulated scenes, that range from having two simple up to three complex surfaces. These surfaces were constructed based on equation (6), with $\hat{\beta}$ manually chosen to produce a variety of conditions that determine the robustness of the surface projection process. The range to each pixel was produced by sampling within fixed regions across each scene, determining a mean range. If a region overlaps multiple surfaces, then this range results in a mixed pixel.

Each of the twelve scenes, shown in Figure 6, fall into one of four categories. Scenes 1 to 3 have two surfaces and remain relatively simple. Scenes 4 to 6 have two complex surfaces that are positioned to more deeply test the robustness of the restoration. The remaining six scenes consist of three surfaces, with Scenes 7 to 9 testing the ability of the Otsu threshold when the area around a point potentially captures all three surfaces. The last three scenes are unusual and designed to more deeply test the robustness of the restoration process. Scene 12 contains points that wrap around due to the ambiguity distance.

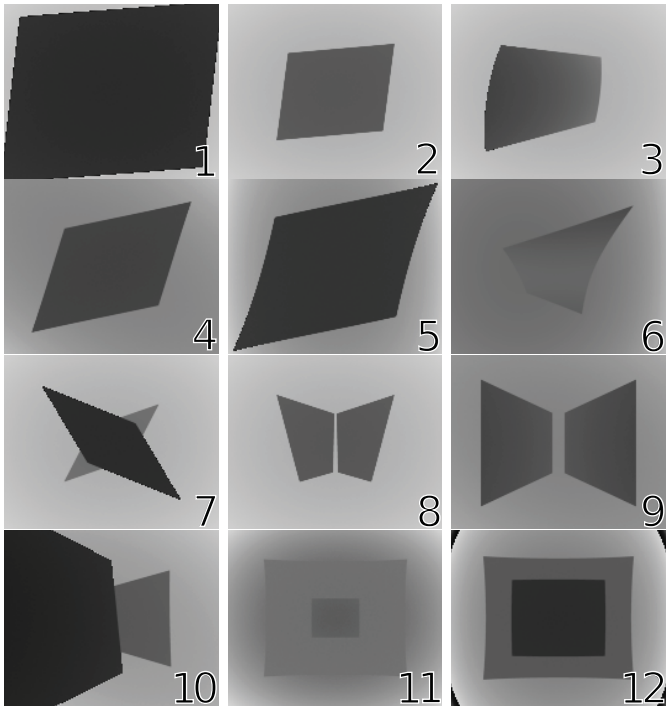
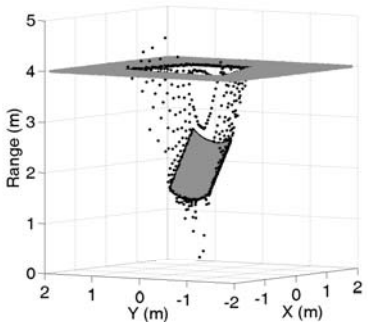


Figure 6. The twelve scenes from the perspective of the camera, where the lighter the intensity the further the point is from the camera.

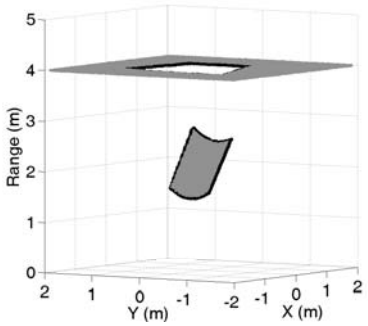
The mixed pixels and the restoration of Scenes 3 and 7 are shown in Figures 7 and 8, respectively. The visual difference between the scene with mixed points and their restored counterparts can easily be seen. Scene 3 had the most mixed pixels accurately restored with only four points projected on to the wrong surface. In contrast, Scene 7 had 127 points incorrectly restored. These are predominately from the area around a mixed pixel of three surfaces and have been projected to a point in an area between the three surfaces. This problem is related to the use of Otsu thresholding which is designed to identify two unique regions. An algorithm designed to identify multiple regions would potentially fair better.

The mean run time of all simulated scenes was 49 seconds, and was carried out using MATLAB R2008a on an Intel Core 2 Duo 3 GHz with 3.25 GB of RAM. If Otsu tests each centimetre instead of each millimetre then this run time drops significantly to 6 seconds.

In Figure 9 the number of points outside the ± 15 mm bound from their correct location are shown for each scene. Scene 8 has the lowest accuracy with 73% of the mixed pixels restored correctly. The best accuracy is 99% from Scene 3 and the mean

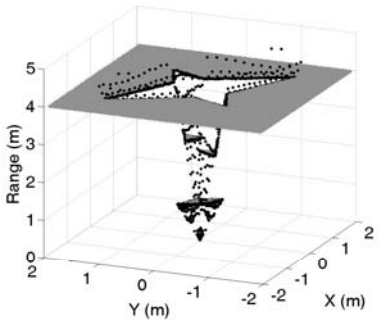


(a) Initial mixed scene.

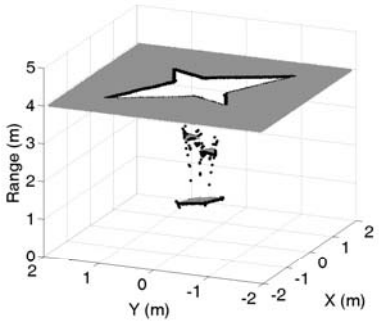


(b) Restored version.

Figure 7. Initially mixed and restored versions of Scene 3. The mixed pixels are black and occur around the surfaces which are grey.



(a) Initial mixed scene.



(b) Restored version.

Figure 8. Initially mixed and restored versions of Scene 7. The mixed pixels are black and occur around the surfaces which are grey.

accuracy for all tested scenes is 93%. Because these scenes contain Gaussian noise generated with a ± 0.01 m variance, they were each reproduced 10 times, each time with different noise. By varying the bounds, the effect of both the noise and the placement ability of the restored points can be measured in relation to their ideal location. This has been carried out for Scene 5, with Figure 10 showing how the number of points decreases as the bounds increase. The point count in the restored scene decreases until it plateaus with the remaining points generally having been projected onto the wrong surface. This plateau tends to occur when the bounds are at about ± 25 mm around each surface. Because these bounds increase, the mixed pixel count in the initial mixed scene also drops, though a significant difference between it and the restored scene continues to remain. From these results, the mixed pixel restoration process is shown to dramatically improve the location of mixed pixels without the need to remove them. This capability is not limited to simulated scenarios, but can also be applied to real data, with the following section demonstrating this.

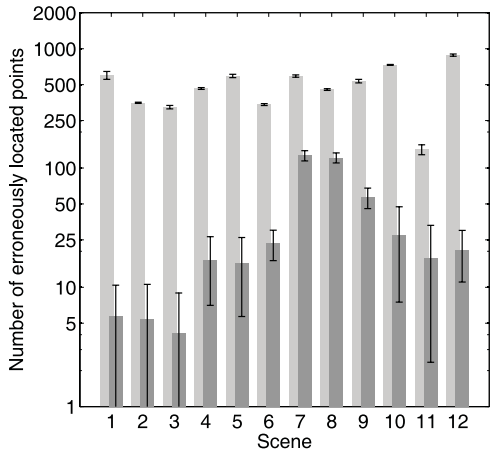


Figure 9. Number of pixels outside the ± 15 mm bounds of the surfaces in each scene. The light grey bar indicates the initial mixed scene, while dark grey shows the restored scene. The bars are set at the mean of each scenes 10 replications, with the error bars being set at two standard deviation.

B. Real Scene Result

Applying the mixed pixel restoration process to real data further tests its ability and demonstrates that it is not limited to simulated scenes. For this test, a frog ornament of approximately 300 mm high and 200 mm wide and deep was captured using a SwissRanger SR4000 range imaging camera, where a large number of mixed pixels were produced from the placement of the frog. The mixed pixel restoration was then applied to the produced point cloud shown in Figure 11(a) resulting in the identified mixed pixels being projected onto one of the two surfaces, which generated the point cloud shown in Figure 11(b). The run time for this scene took approximately 11 seconds to restore all the mixed pixels.

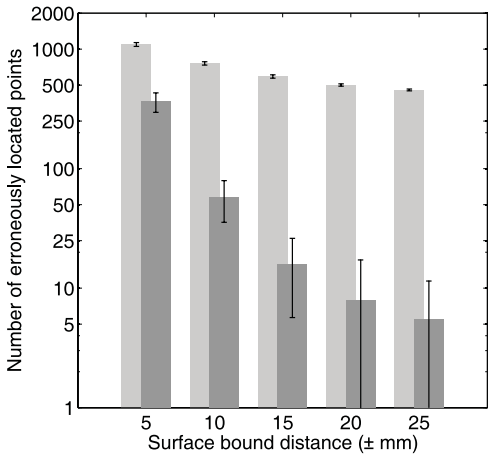
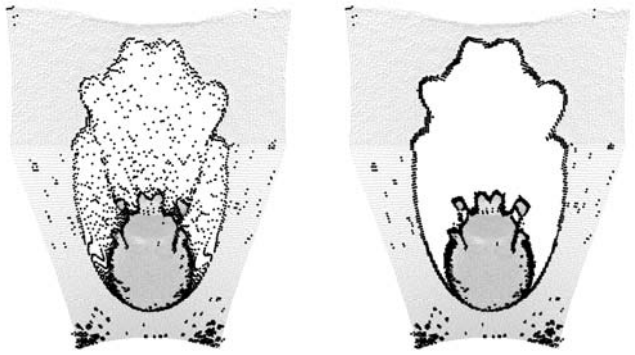


Figure 10. Number of pixels that are outside a particular set of bounds that contain the two surfaces in Scene 5. The light grey bar indicates the initial mixed scene, while dark grey shows the number of points after the restoration process has been applied.



(a) Initial mixed scene. (b) Restored version.

Figure 11. Captured scene of a frog ornament. The black points are the mixed pixels identified in the initial mixed scene.

VI. Conclusion

In a captured point cloud, mixed pixels can be a serious problem affecting how the scene is interpreted and processed for various tasks. Currently, the general method of dealing with detected mixed pixels is to simply discard them; this however has problems that may distort or damage the point cloud. This paper introduced an alternative method of dealing with the mixed pixels, in which the goal is to restore mixed pixels to their correct position. This was achieved by determining the two classes in the area around a mixed pixel and fitting a surface to the points in the nearest class and projecting the mixed pixel on to this surface.

Testing of this technique carried out using a set of simulated scenes found that the majority of the mixed pixels could be restored with a high level of accuracy. The restoration process was most prone to error when the area around a mixed pixel composed of three surfaces, which the Otsu thresholding algorithm was incapable of handling. If an alternative and better segmentation algorithm is utilised, the incorrectly placed mixed pixels would likely be placed correctly, making the projection process both more robust and further increasing its already high level of accuracy.

Acknowledgment

Robert Larkins acknowledges the support provided by both the Range Imaging and Waikato Doctoral Scholarships.

References

[1] M. Cree, A. Dorrington, R. Conroy, A. Payne, and D. Carnegie, "The Waikato range imager," in *Proceedings of Image and Vision Computing New Zealand (IVCNZ'06)*, 2006, pp. 233–238.

[2] M. Hebert and E. Krotkov, "3D measurements from imaging laser radars: how good are they?" in *IEEE/RSJ International Workshop on Intelligent Robots and Systems' 91.'Intelligence for Mechanical Systems, Proceedings IROS'91*, 1991, pp. 359–364.

[3] P. Tang, D. Huber, and B. Akinci, "A Comparative Analysis of Depth-Discontinuity and Mixed-Pixel Detection Algorithms," in *3-D Digital Imaging and Modeling, 2007. 3DIM'07. Sixth International Conference on*, 2007, pp. 29–38.

[4] M. Adams and P. Probert, "The interpretation of phase and intensity data from AMCW light detection sensors for reliable ranging," *The International Journal of Robotics Research*, vol. 15, no. 5, p. 441, 1996.

[5] M. Gebbinck and T. Schouten, "Decomposition of mixed pixels," *Image and Signal Processing for Remote Sensing II*, pp. 104–115, 1995.

[6] J. Godbaz, M. Cree, and A. Dorrington, "Mixed pixel return separation for a full-field ranger," in *Image and Vision Computing New Zealand, 2008. IVCNZ 2008. 23rd International Conference*, 2008, pp. 1–6.

[7] A. Dorrington, M. Cree, A. Payne, R. Conroy, and D. Carnegie, "Achieving sub-millimetre precision with a solid-state full-field heterodyning range imaging camera," *Measurement Science and Technology*, vol. 18, no. 9, pp. 2809–2816, 2007.

[8] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *Automatica*, vol. 11, pp. 285–296, 1975.