

A Structured Light Range Imaging System Using a Moving Correlation Code

Frank Pipitone

Navy Center for Applied Research in Artificial Intelligence
Naval Research Laboratory
Washington, DC 20375-5337 USA
pipitone@aic.nrl.navy.mil

Ralph Hartley

Navy Center for Applied Research in Artificial Intelligence
Naval Research Laboratory
Washington, DC 20375-5337 USA
hartley@aic.nrl.navy.mil

Glen Creamer

Naval Center for Space Technology Naval Research Laboratory
Washington, DC 20375-5337 USA
creamers@space.nrl.navy.mil

Abstract

We describe a system which produces a dense accurate range image using 8 consecutive frames of camera data in conjunction with a special projector. A prototype has been built and tested, and yields a typical range error of about 0.2mm at 2 meters range with a baseline just over one meter.

The camera is directed at a scene, along with a stripe projector consisting of a thin light source (xenon tube and slit) on the axis of a turntable, and a binary mask conforming to a cylinder coaxial with this. The mask has alternate opaque and transparent stripes parallel to the axis. It forms a sequence in which each sub-sequence of given length n (8 here) is different.

No lens is used in the projector, deliberately smoothing the resulting illumination in a shadowing process. In operation, the turntable rotates, and images are taken at uniform angular intervals for several consecutive frames. In the consecutive frames, a given pixel records samples of the brightness of a fixed surface point. The vector consisting of those samples, when normalized, is unique to the place in the sequence from which it came, thus enabling the computation, via a fast indexing process, of the 3D position of the surface point. The code is similar to a DeBruijn sequence, but

modified to reduce range error. Several types of calibration and error compensation were used to produce an accurate range image.

1 Introduction

Recent years have seen accelerating progress in methods for acquiring range images. We cite four surveys [5] [2] [3] [1] to generally reference triangulation range imaging. The most closely related prior work is [4], which introduced DeBruijn sequences to triangulation range imaging. It also introduced the use of a moving fixed code pattern and the use of shadow optics. Here we extend this work by introducing codes with fewer uniform regions than DeBruijn sequences, improving the calibration, and using a fast matching procedure.

Some of the natural measures of performance of a range imaging sensor are range resolution, the density of range pixels, the frame rate of range images, and the tolerance of relative motion of sensor and scene.

We are interested here in giving priority to resolution and density, while retaining moderately good speed and motion tolerance. The range images produced by our scanner are dense, in the sense that each

pixel is an independent range measurement, in contrast to some methods, such as stereo, that assign a range to a neighborhood. Our range resolution is limited by the pixel size with subpixel accuracy typically achieved in uniform regions.

We describe a system which produces a dense accurate range image using a 8 consecutive frames of camera data in conjunction with a special projector. A prototype has been built and tested, and yields a typical range error of about 0.2mm at 2 meters range, using a baseline of just over a meter.

Figure 1 shows the basic design of the scanner. The

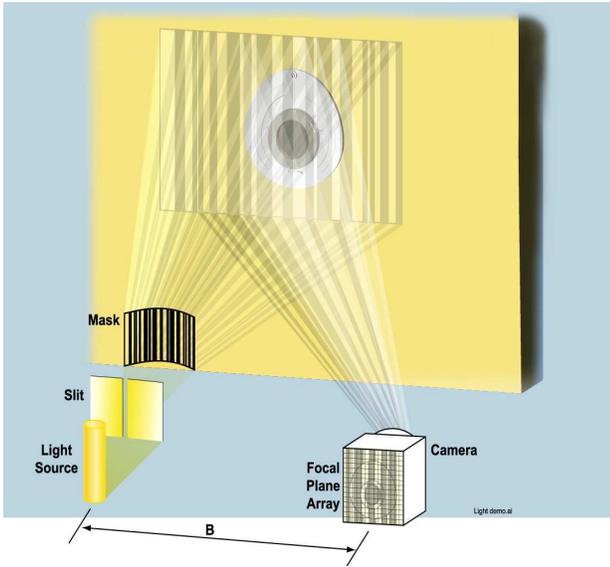


Figure 1. Basic layout of the scanner.

method resembles prior spatial encoding systems in that a camera and a stripe projector are directed at a scene, but the projector is unusual. It consists of a thin light source (xenon tube and slit) on the axis of a turntable, and a binary mask conforming to a cylinder coaxial with the rotation axis. The mask has alternate transparent and opaque stripes parallel to the axis. It forms a sequence in which each sub-sequence of given length n is different.

No lens is used in the projector, deliberately smoothing the resulting illumination in a shadowing process. In operation, the turntable rotates, and images are taken at uniform angular intervals for several consecutive frames. The interval is the angular width of one code stripe. No switchable mask is needed; a single rigid pattern mask is simply rotated. In the consecutive frames, a given pixel records samples of the intensity of the projected light, as reflected from the point on the scene observed by the pixel.

The main advantages of the lenseless system are near infinite depth of field, simplicity, and low cost. It also lacks spherical and chromatic aberration, and radial distortion, making the structured illumination field nearly a one-dimensional pattern; i.e., dependent (in slit-centered cylindrical coordinates) on the angular coordinate only, facilitating the code matching and calibration.

The vector consisting of the consecutive pixel values, when normalized, is unique to the place in the sequence from which it came. Thus we can compute the position in 3-space of the surface point at which the pixel is looking. The code is similar to a DeBruijn sequence, which was used in related work [4], but modified to provide smaller and more uniform error.

Several types of calibration and error compensation were used to produce an accurate range image. These include camera calibration, flash brightness variation correction, and compensations for the small variation of the projector's light field waveform with distance from the projector axis. The projector geometry is captured with reference images of the code projected onto a white plane in known position relative to the camera.

2 Normalization

Even small variations in flash brightness from frame to frame would introduce substantial errors in the code vectors. Normalizing each frame as a whole before extracting the code vectors suffices. To correct for this

Each point in the target reflects a different proportion of the light from the projector back to the camera. This depends on the target's albedo and, for specular surfaces, on the surface normal and the directions to the projector and camera.

There are also two different sources of background light: ambient light, and scattered light from the flash. Ambient light is mostly eliminated by using a short exposure coordinated with the flash, and by subtracting images taken without any flash. It is difficult to completely eliminate light from the flash reflected from parts of the projector.

To produce code vectors that are independent of these effects the 8 brightness values for each pixel are normalized by multiplying by a factor and adding a constant so that the sum of the 8 components of the code is zero and the sum of their squares is one.

Normalization reduces the number of degrees of freedom in the code vectors by two. If the number of frames were too small, this would result in ambiguities in the code, with six or more frames this is not a problem, as

long as the normalization is taken into account when designing the code.

3 Code Selection

The starting point for constructing our code is an order eight DeBruijn sequence, a sequence of bits such that each eight bit subsequence appears exactly once. It is not immediately obvious that smoothing a binary code would give a good continuous code, but it is so.

The continuous code is a vector valued function $V_i(x)$. There are two desirable properties for this function: good local accuracy, and low global error rate.

Local accuracy increases in proportion to the magnitude of the gradient of the code. It is greatest where many of the components are changing. In terms of the images, that corresponds to the pixel being near intensity transitions in as many of the eight frames as possible. If the gradient in frame i at a given pixel is $\dot{V}_i(x)$, the location error at that pixel is proportional to the intensity error and inversely proportional to $\sqrt{\sum_{i=0}^8 \dot{V}_i(x)}$.

The global error rate is determined by how often, and how closely, different parts of the 8 dimensional curve of code vectors approach each other. If two points have nearly the same code vector they can not be reliably distinguished from each other.

We want these properties to the greatest extent possible along the whole length of the curve. If, as turns out to be the case, most of the curve is good but there are a few bad spots, it can be greatly improved by deleting a few subsequences from the original DeBruijn sequence.

The all 0 and all 1 subsequences are the most obvious subsequences to leave out, since they contain the fewest transitions. In general it helps to avoid long uniform sequences, but the longest ones are the worst.

The other particularly bad subsequences are where 0 and 1 alternate. The gradient is large most places, but when the samples fall exactly at the middle of the original bits, all the components reach their maximums simultaneously.

Simulations using modeled noise confirmed that the 8th order code with the worst subsequences removed had good error properties.

For lower order codes it was possible to optimize the sequence for a given noise model. For each subsequence of length Order+2 (to take the smoothing into account), a simulation measured the local accuracy. For each pair of such sequences we measured the global error rate that would result from including them both in the same code.

This tabulated data permitted the efficient evaluation of full length codes, and rapid search. Optimized codes of order up to six were found in this way, though the actual hardware used an unoptimized order 8 sequence. An important advantage of these codes over some other structured light sequences is that most pixels fall on an intensity gradient in multiple frames. For the optimized codes, all pixels fall on a steep intensity gradient for many frames. In comparison, using two phase shifted sinusoids and a binary Grey code sequence would only effectively provide one frame with a unit gradient for each pixel.

4 Hardware Implementation

The projector (Figure 2) is the only part of the scan-

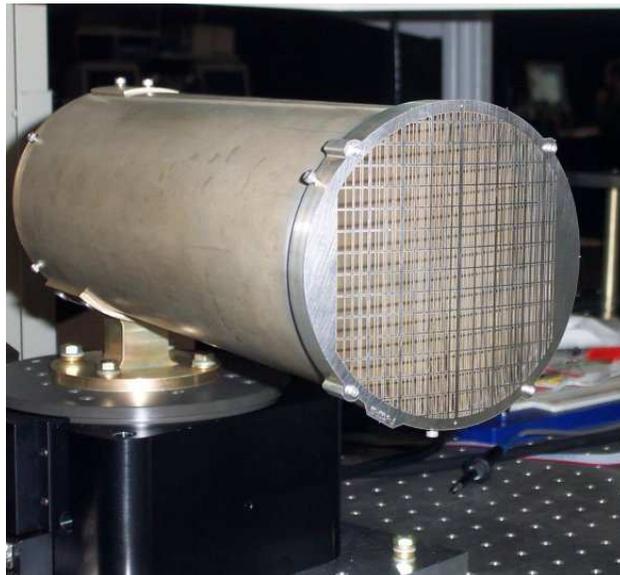


Figure 2. The projector. The mask is visible in front. The bulb assembly is at the far end on the vertical axis of the motor.

ner that is not basically off the shelf.

The flash tube is mounted in a cylindrical reflector with the slit built in, on the axis of the motor. The mask is made of etched stainless steel sheet. The slit is 0.03" wide, and each bit of the mask is 0.024", the mask is held 12" from the slit at the end of a 4.5" tube. The round tube spreads light reflected from its inside surface evenly, flat surfaces would produce reflected images of the code pattern.

The motor is substantially over-sized. It turns less than half a degree, with essentially no load, during the scan, but the encoder and bearings need to ensure that

the positions are the same each time.

All the tests used a 1024x1024 pixel, 12 bit/pixel Dalsa camera with a 50mm f1.4 lens. For most of our experiments (all those shown) the baseline was 44". It is important to note that the range accuracy was camera-limited, not projector-limited. It was easy, using lensless shadow optics, to obtain intensity gradients which contribute smaller code matching error than the camera related error arising from reflectance variation (see below).

Figure 3 shows an image of the code as produced by



Figure 3. The pattern as projected. Part of one of a set of images used for calibration.

the projector.

5 Calibration

Camera calibration was performed by conventional methods [6]. Calibration of the camera is important for absolute accuracy but the scanner will function without it. The same is not true for calibration of the projector. In addition to geometrical properties, the pattern of illumination from the projector, for each frame, needs to be known very accurately.

The illumination pattern is generated by the mask pattern smoothed by the slit light source. Considering the slit as uniform, isotropic, parallel to the mask, and at a known position suffices for evaluating different masks, but not for accurate matching. The actual mask is not exactly parallel to the slit, resulting in additional smoothing. The geometry of the bulb and reflector behind the slit results in the intensity of light not being uniform across its width, and it is far from isotropic (which would not be desirable anyway, as it would result in more light being wasted). A small change in the distance from the mask to the slit would result in a significant rescaling of the pattern. In addition the projector was only designed to deliver flashes at *repeatable* angles, the angles between frames may not all be exactly the same.

It might be possible to measure each of these effects separately and compensate for them, but it would be difficult. Fortunately, it is not necessary. They can all be calibrated together by taking one set of images of a reference plane, a planar target of uniform reflectivity. The reference images provide a sample of the light pattern from the projector, against which target pixels

can be matched. All the above effects are the same for target and the reference plane, so they need not be determined explicitly. It suffices to determine the position in the reference image along the same epipolar line to which a target pixel matches best, the range can then be computed by simple geometry.

A section of one of the reference images is shown in Figure 3.

If the target were actually the same as the reference plane, the resulting range image would be perfectly flat. So deviations from flatness of the reference plane will be directly reflected in systematic errors in range. Because of normalization, slow variations in albedo in the reference plane would have little effect, but high frequency variations would interfere with subpixel accuracy.

Because the projector pattern is very nearly uniform in the direction parallel to the slit, which is approximately parallel with the camera's vertical direction, it suffices to record the central horizontal line of the reference image, together with the point on that line to which each other point matches best.

To reduce noise in the reference, which would degrade all range images, multiple (in our experiments 10) sets of images are taken of the reference plane and averaged. In addition, each point on the center line is averaged with a small (20 pixel) vertical neighborhood. This reduces noise further, and reduces the effects of local deviations from planarity or uniform reflectance.

If the reference plane has a known distance and orientation, and either the baseline or absolute angle are known, then no additional geometric calibration is needed. The match position, together with the camera calibration, completely determines the position of a pixel in three dimensions.

6. Matching

Normalization of each pixel in a set of target images produces an eight dimensional code vector for each pixel, which is matched against the reference, which is an eight dimensional curve parameterized by position in the reference images.

The code vector is quantized and looked up in a pre-computed table to obtain a small set of possible approximate match positions. Each of those is iteratively refined by least squares fitting to short linear segments of the reference, and the closest match is found.

7 Subpixel Compensation

To obtain the maximum possible accuracy, especially when the range is much larger than the baseline,

finding the match position to the nearest pixel is not sufficient. Even linear interpolation between adjacent pixels results in noticeable systematic error.

Since the reference curve is a slowly varying function, it can be sub-sampled to finer resolution. This could in principle be done as part of the matching process, but it is more efficient to do it while pre-processing the reference images.

7.1 Focus

If either the reference images or the target images are out of focus, the code vectors will not match correctly. It does not matter so much that they be *in* focus, so much as that they be out of focus by the same amount.

To compensate a small amount of blurring was performed on the reference data before matching. This does not compensate perfectly, because the point spread function of the true blurring is not known exactly, also if the surface of the target is not parallel to the reference plane, the required amount of blurring will be affected by foreshortening.

For very short ranges, relative to the distance from the mask to the slit, the smoothing of the mask by the slit will be less than the far field asymptotic amount of smoothing. For targets that close, the pattern projected on the target may be sharper than the pattern observed on the reference plane, even though the reference images are in perfect focus. Our tests were at ranges for which this was not an important effect. It is undesirable to blur the target images, because that would increase the effective pixel size (see next subsection).

A more complete way to compensate for focus would be to store reference data for different focus settings, and different ranges. This would give better compensation, at all ranges, at the cost of more work during the calibration phase.

Since real lenses have limited depth of field, the degree of focus will sometimes vary from point to point. A first pass producing an approximate range image could be used to determine the correct focus compensation for each point, at little extra computational cost. Most of our tests were done with targets at large ranges relative to their sizes, so this was not done.

7.2 Reflectance Variation

Rapid reflectance variations on the target (e.g. albedo edges) are another source of error in the range image. If the reflectance is not constant across a pixel,

the light seen in each image will represent the center of the pixel, but will instead come from the brighter side.

The effective position of a pixel with true position $x_0 = 0$ is

$$\Delta x = \frac{\int_{-\infty}^{\infty} xI(x)\phi(x) dx}{\int_{-\infty}^{\infty} I(x)\phi(x) dx} \quad (1)$$

Were $I(x)$ is the reflectance, and $\phi(x)$ is the pixel's point spread function. This would be zero if the reflectance were a constant or the pixel was only sensitive to light from a single geometric line. If the effective width of the pixel is w there will be a range error equal to

$$\Delta R = \frac{R^2}{B} \left(\frac{w}{fI(0)} \frac{dI}{dx} + O(w^2) \right) \quad (2)$$

Where R is the range, B is the baseline distance and f is the focal length,

If reflectance varies smoothly over the image, this error can be at least partially compensated for. However the width of the pixels still determines the limit of range resolution in many of our images. Subpixel resolution depends on the target having certain properties, and in their absence there is a limit to the resolution of any triangulation based ranging method.

8 Results

The scanner was tested on numerous targets, only a few of which are shown here. The stripe-like error, highlighted here by our pseudo-Lambertian rendering, is due primarily to imperfect compensation for code match error due to the range dependence of both camera focus and code waveform. We believe that these systematic errors can be more completely removed with further effort.

Figure 4 shows a part of a spacecraft mock-up. Images of the same target were taken at ranges from 3 to 20 meters. Figure 5 is a shaded view of a face. Because the images are taken sequentially, the scanner is sensitive to target motion, and that may be a major source of error in this image, but the images are taken quickly enough that it is not a major problem.

9 Conclusions and Future Work

From a sequence of images of a single smooth coded pattern we extract a dense accurate range image. The projector is simple, and can be built at low cost.

The code used was not completely optimized. Improvements in the code could result in even less sensitivity to noise in the images, or allow the use of fewer images to produce the same range image. Heuristic

methods, such as Genetic Algorithms or Simulated annealing, could be used to find optimal, or near optimal codes of greater length.

By making reasonable assumptions about the surface, e.g. that it is piecewise smooth, etc. and combining information from a local region it may be possible to increase the level of subpixel resolution.

References

- [1] P. Besl. Active optical range imaging sensors. *Machine Vision and Applications*, 1(2):127–152, 1988.
- [2] F. Blais. Review of 20 years of range sensor development. *Journal of Electronic Imaging*, 13(1):231–243, jan 2004.
- [3] H. Everett, D. DeMuth, and E. Stitz. Survey of collision avoidance and ranging sensors for mobile robots. NC-COSC Technical Report 1194, Naval Command Control and Ocean Surveillance Center, RDT&E Division, dec 1992. Revision 1.
- [4] F. Pipitone and R. Hartley. Moving-correlation-code triangulation range imaging. In B. Nnaji and A. Wang, editors, *Sensors and Controls for Advanced Manufacturing*, volume 3201, pages 91–97. SPIE, oct 1997.
- [5] J. Salvi, J. Pages, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, apr 2004.
- [6] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

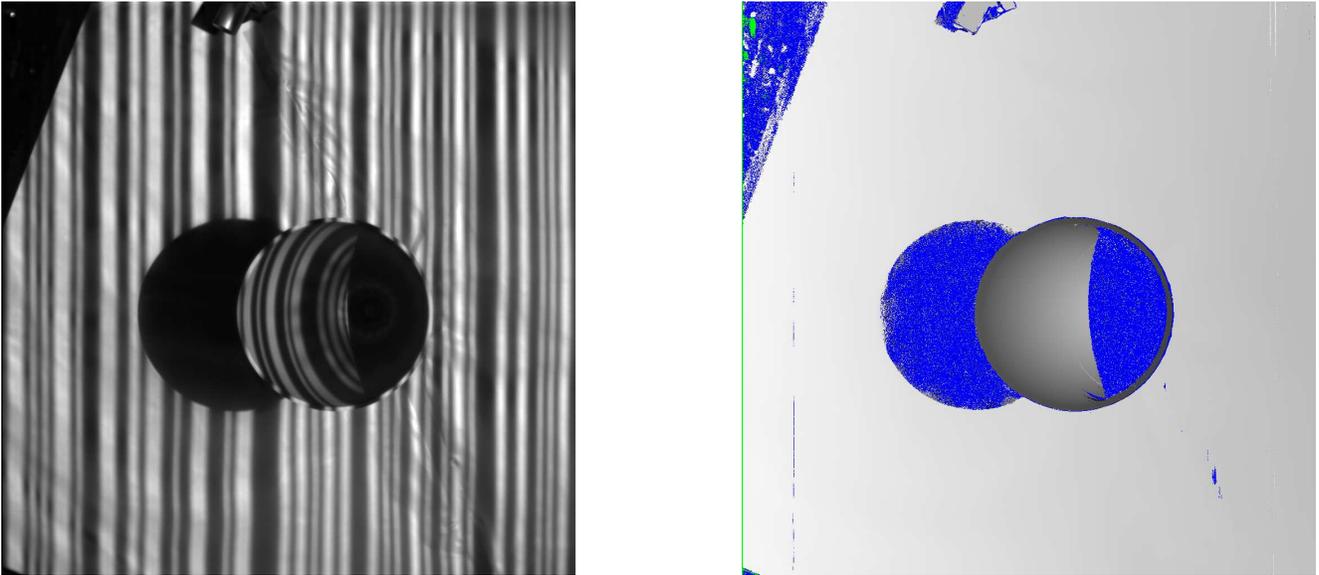


Figure 4. Left - One of the raw images of a nozzle mock-up, looking at the inside surface of the conical nozzle. Range is about 4m. Right - Range image. dark = close, light = far, blue = no range



Figure 5. A shaded view of a range image of a face. Range is about 2.5 meters.