October 01, 2019

Determining Local Depth From Structured Light Using A Regular Dot Grid

Hart Levy

Determining Local Depth From Structured Light Using A Regular Dot Grid

Abstract:

This publication describes systems and techniques directed to determining local depth, or depth gradients, of a scene from structured light using a regular dot grid pattern and a single receiving camera. A patterned illuminator with a repeating nature provides a structured light that is used to generate depth information. Because any patch of dots can be expected to repeat at a known horizontal distance, an auto-correlation of the image with itself produces peaks along with these repetitions. By finding the disparity from such a peak to its expected location, we measure the local disparity between pairs of patches in the image. This process produces a local depth map, which is effectively the gradient of the true depth map. This map can be calculated with low complexity, calibrated with simple captures, and is robust to a range of miscalibrations.

Keywords:

structured light, depth map, depth map generation, depth measurement, depth sensing, repeating dot grid, face authentication, facial recognition, biometric, image, calibration, illuminator, camera, autocorrelate, gradient vector, local depth, smartphone, infrared, IR

Background:

Active stereo and structured light are two complementary techniques used in mobile devices to generate depth maps of scenes based on the concept of inferring depth from parallax. Each technique has tradeoffs in computing parallax as a disparity along a baseline from a source scene to a reference scene. Active stereo systems require two cameras to capture a source and reference scene, a simple illuminator dot pattern to provide texture, and a stereo algorithm to determine disparity. Structured light systems require a more precisely engineered illuminator, such as the known structure of a pattern as a reference, and a complex algorithm that requires a coding or lookup of unique local textures to determine the disparity. Both of these techniques require expensive and complicated components and very tight constraints on calibration. They are also very sensitive to errors in the calibration parameters.

For example, dual cameras (stereo cameras) enable the capture of stereo images of a scene because the two cameras are located at different horizontal or vertical locations and capture two slightly different views of the same scene. The two images are compared and a disparity is calculated or generated to obtain depth information about the scene. However, calculating a good depth map requires that the two cameras are correctly calibrated to be rectified and aligned.

Standard structured light systems use a single illuminator and single camera to capture depth maps. In general, high-quality depth from structured light requires a pseudo-random or coded illumination source, in order to accurately identify unique spatial patches in an image and to calculate a disparity from their expected locations. However, this method requires dense structure and also precise calibration.

Description:

This publication describes systems and techniques directed to determining local depth, or depth gradients, of a scene from structured light using a regular dot grid pattern and a single receiving camera. A patterned illuminator with a repeating nature provides a structured light that is used to generate depth information. Since any patch of dots can be expected to repeat at a known horizontal distance, an auto-correlation of the image with itself produces peaks along with these repetitions. By finding the disparity from such a peak to its expected location, we measure the local disparity between pairs of patches in the image. This process produces a local depth map, which is effectively the gradient of the true depth map. This map can be calculated with low complexity, calibrated with simple captures, and is robust to a range of miscalibrations.

More specifically, an image is captured of an illuminated pattern projected on a flat plane. A calibration map is generated to estimate a repetition of a patch in the image along a baseline vector between an illuminator (*e.g.*, an IR dot illuminator) and a camera (*e.g.*, an IR camera) by using an angle and pitch between dots. The calibration map is then applied to a scene (*e.g.,* a user's face) captured with the illuminator, and the image is divided into patches to calculate a z-scored template to be matched. Autocorrelation is applied within +/- n pixels from the baseline vector to find the distance with the best match (local disparity). A gradient vector is determined by repeating the disparity calculation process with calibrations along different initial vectors to generate the local disparity along multiple axes. Finally, a depth map is generated from an iterative integration of the gradient map using an image pyramid technique.

Figure 1 illustrates an example of a repeating dot pattern generated by an IR dot illuminator. In this example, the same four (4) dot patch is repeated approximately 35 pixels away in the x-axis and approximately one (1) pixel away in the y-axis. The predictable repetition of certain IR dot illuminators means that they can be used as their own reference under this disclosure to produce a local depth map.

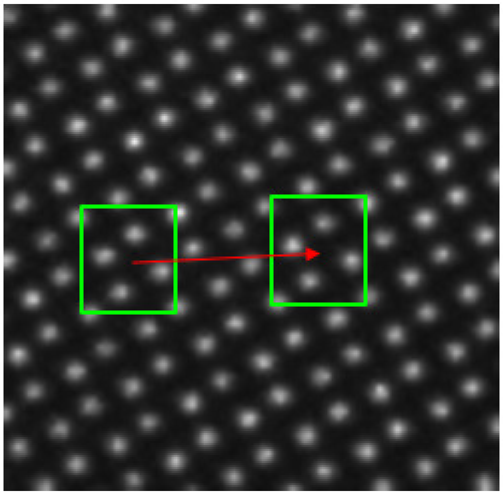

Figure 1

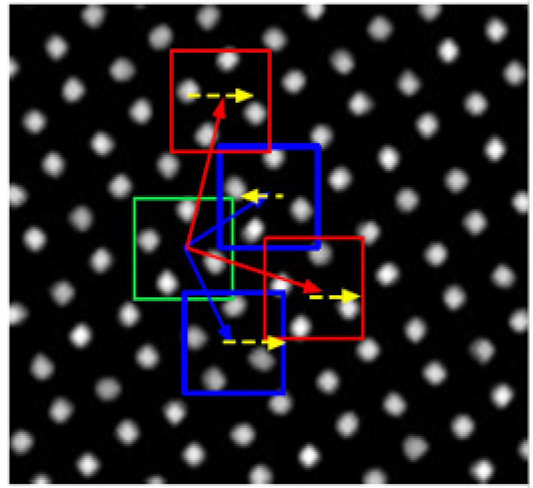

Figure 2

Figure 2 illustrates an example of the four (4) closest repetitions of the dot pattern, in pairs with 90-degree relative orientation. The green box represents an initial reference patch pattern. The blue and red boxes represent the replicated pattern along multiple expected vectors. Yellow arrows show the possible disparity vectors calculated relative to a calibrated repetition location.

For a perfectly repeated pattern, any patch of the pattern will be replicated along multiple expected vectors. If the pattern is imaged by a camera separated along some baseline, it will shift due to stereo parallax (as the conventional structured light signal would be). We use autocorrelation to compute the actual locations of these repetitions. By computing the observed vector, and comparing its position versus the expected known vector along the baseline, we determine the local disparity of the pattern. This local disparity value can be interpreted as the local gradient of depth.

An idealized pattern would need no calibration, save for knowledge of the spacing and rotation of the dots. In reality, there are local non-uniformities due to lens distortion and angular distortion of the pattern. So to calibrate the image, we use our knowledge of the angle and pitch between dots (which can be easily measured from a single capture), to estimate where we will see patch repetition along the baseline vector between illuminator and camera. We calibrate this by taking an image of the pattern on a flat plane at some distance z. Then, we divide the image into patches and compute the local vectors to the nearest repetitions. These vectors should vary slowly over the image field, as depicted in Figure 3.

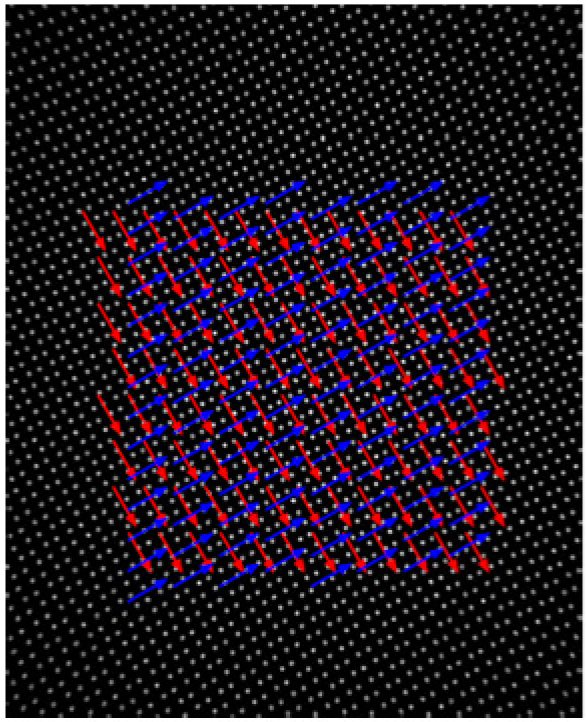

Figure 3

In one example, a flat dot capture image is divided into patches of 16x16 pixels, and template autocorrelation is used to find the (x, y) disparity vectors to the nearest repetition. The resulting maps are shown in Figure 4, as calibration vector maps for x and y. In this example, X values search through [-1, 7], and Y values range [6, 14]. These maps are noisy far from the center, due to the simple nature of the search algorithm. Additionally, each surface appears roughly parabolic/hyperbolic, and fitting a paraboloid can give an accurate and less noisy calibration. Simulation can also be used to model the expected surface (considering illuminator spatial profile and camera distortion), then a small set of parameters may be sufficient to calibrate.

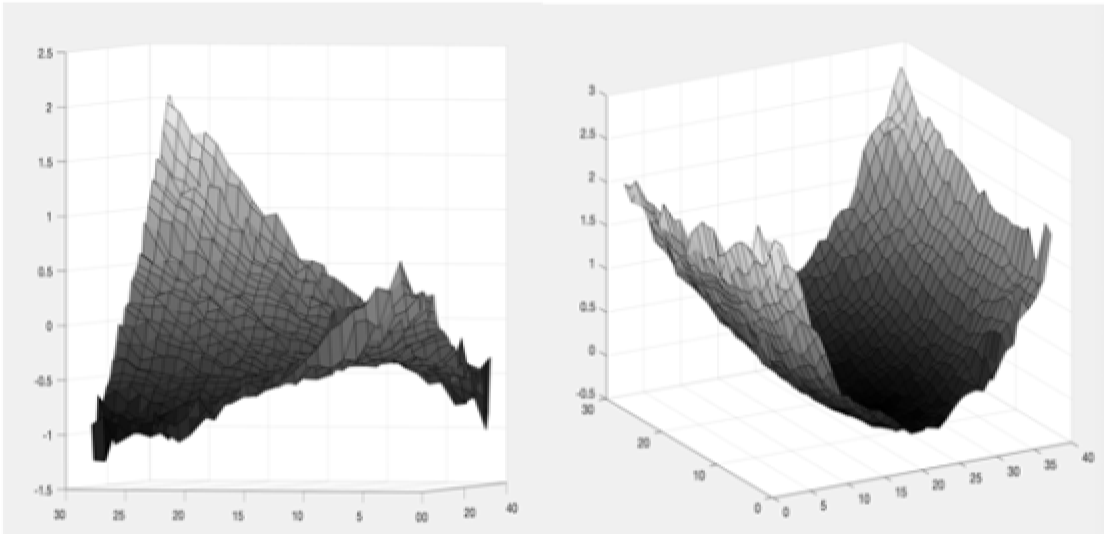

Figure 4

Once the calibration maps have been produced, they can be applied to scenes captured with the same illuminator. To preprocess the captured data, a lens shading (relative illumination (RI)) correction and a dot enhancing kernel (an unsharp mask) are applied. Then, the matching algorithm is very similar to the calibration algorithm itself. The image is divided into patches, which move along an MxM window (*e.g.,* 16x16) in steps of K pixels (*e.g.,* 4). At each patch, the z-scored template is calculated to be matched. For each point, the expected patch vector from a smoothed version of the calibration map is identified, then we autocorrelate along the baseline x vector within +/- n pixels from this vector to find the distance with the best match. This distance is the local disparity. Repeating this with calibrations along different initial vectors is used to generate local disparity along multiple axes, allowing the gradient vector to be determined.

Initializing the calibration map with the four nearest neighbor vectors, and comparing these calibrations to a given scene, generates four different gradient components. Although any two would be enough to generate full normal maps, using four gives better accuracy and can help compensate for noise. Figure 5 demonstrates an example of four different gradient maps of a facial image and associated gradient vectors.

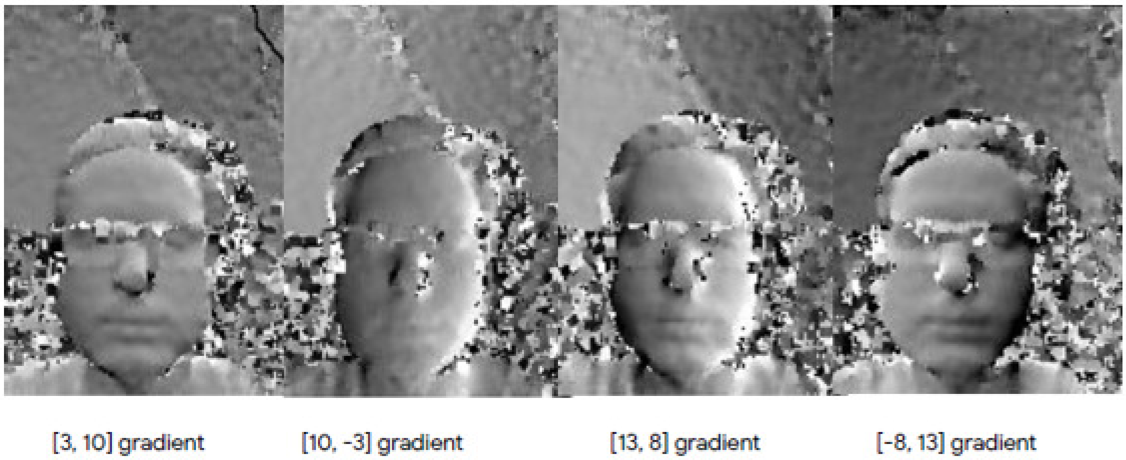

Figure 5

By combining the four gradient maps in proportion to their projections along an arbitrary direction (linear combinations of the maps), we can generate a new gradient map along this direction coordinate. For example, see Figure 6 depicting results using the indicated formula. Additionally, estimating the face orientation allows us to get gradients aligned with the face x-y coordinates, meaning we only need to capture a face in a single orientation.

Also, because the dot pattern is repeated many times in all directions, we can use the same algorithm applied to different repetition vectors to determine the gradient along different axes. This is accomplished by altering the calibration map to use a different starting vector. The original result corresponds to a gradient in x, but by applying the same algorithm with a different calibration map (*e.g.,* initialized along [-3, 33] instead of [35, 0]), we can get a gradient in y instead. In fact, repetitions along arbitrary angles can be used to compute different gradients.

$$G(\theta) = \sum_i G_i \frac{C_i}{|C_i|} \cdot \{cos(\theta), sin(\theta)\}$$

Figure 6

Figure 7

Figure 7 represents an example of producing a standardized gradient map by combining two gradient maps along x and y rotations to give the gradient map corresponding to the image (x, y) vectors. In the example of Figure 7, red represents the x component, cyan represents the y component, and normalized disparity values range from -0.5 to +0.5 pixels.

If we convert the disparity gradient into metric depth gradient, we can calculate precise normal maps from these images. A normal vector of a surface can be calculated directly from a gradient vector as follows:

$$N = \{F_x, F_y, -1\}$$

where F_x and F_y are the derivative of the function in x and y. Note that we are not precisely calculating F_x and F_y, since we only are doing calculations in pixel and disparity space, rather than metric space.

To obtain the metric gradient from the disparity gradient, we first convert from disparity delta, to depth delta, then to first order, as follows (where f = focal length, b = baseline):

$$\Delta z = z_0^2/fb\,\Delta d$$

Next, convert from pixel distance to metric distance, to first order (θ = FOV in x, n = pixels in x):

$$\Delta x = \frac{z_0 tan(\theta/2)}{n/2}\Delta p$$

Then, combine, and use relevant hardware parameters (*e.g.,* n = 480 px, θ = 52°, f = 530 px, b = 40 mm) to convert disparity gradient to z gradient:

$$\frac{\Delta z}{\Delta x} = \frac{z_0 n/2}{tan(\theta/2)fb}\frac{\Delta d}{\Delta p} \simeq 0.23 z_0 \frac{\Delta d}{\Delta p}$$

By using estimated depth to determine the surface normal z coordinate, we generate the full surface normal vector which can be used for face relighting. In this example for faces, we expect z in the range of 30-70 cm, which reasonably constrains the normal z component. Note for photography applications, we would map this to the red, green, blue (RGB) instead of infrared (IR) camera. Also, this solution does not currently handle occlusions (*e.g.,* nose shadow).

Similar algorithms to those used for integrating normal maps in photometric stereo applications apply to generating a surface depth map here, with some parameter tuning and handling of discontinuities. Figure 8 depicts an example of how a depth map is generated from iterative integration of the gradient map, using image pyramids. From the gradient maps it is straightforward to integrate slopes to produce relative depth maps, with an image pyramid/mipmap based recursive algorithm (*e.g.,* adapted from photometric stereo applications). The results for an indoor scenario are very promising, though there are some residual artifacts due to depth offset, low confidence regions, and local discontinuities.

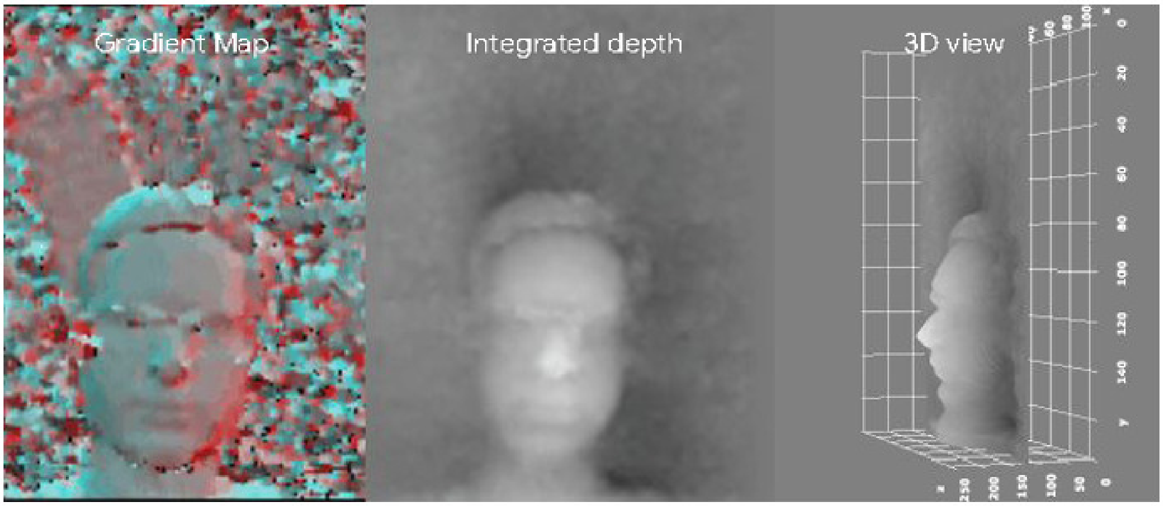

Figure 8

Finally, because the calibration maps used are quite smooth and slowly varying, and epipolar lines need only be approximately known to obtain gradients, this algorithm proves to be very robust to miscalibration, *e.g.,* a wide range of miscalibrations have only minor impact on the gradient result. The most common effect of miscalibration is an overall local offset for the gradient values. This may lead to more significant errors in the integration maps (as integrating a small offset will accumulate), but this may be able to be estimated dynamically. This will have the effect of adding a noticeable slope to an integrated depth map, which may be apparent. However, the gradient map itself does not see significant degradation. As an example, experiments show that there is no substantial degradation in gradient maps for at least three (3) degree shifts in roll, and 1.5-degree shifts in pitch or yaw.

References:

[1] US 20100118123 A1. Depth Mapping Using Projected Patterns. Date of Filing: December 10, 2009.

[2] CN 108924340 A. Verification Method, Verification Device And Computer-Readable Storage Medium. Date of Filing: June 6, 2018.

[3] Saracchini, Rafael Felipe V., Jorge Stolfi, Helena Cristina da Gama Leitão, Gary A. Atkinson and Melvyn L. Smith. "A robust multi-scale integration method to obtain the depth from gradient maps." *Computer Vision and Image Understanding* 116 (2012): 882-895. https://doi.org/10.1016/j.cviu.2012.03.006.