Contactless seismic acquisition using stereo vision

Thomas Rapstine & Paul Sava
Center for Wave Phenomena, Colorado School of Mines

ABSTRACT
Environmental impact and cost of acquiring land seismic data are major factors to consider when designing a seismic survey. We explore means to record seismic data without disturbing, or contacting, the ground surface to reduce environmental impact and cost, while speeding up land data acquisition. Recent developments in computer vision techniques and drone technology lead us to propose passively observing the ground with a drone-borne stereo video camera system for recording seismic data. Two cameras separated by a known distance are used to monitor the distance to a collection of points, using stereo vision theory, representing a textured ground surface. The textured ground surface is vertically displaced according to a known displacement with time; the stereo vision system monitors the distance to each point on the ground surface with time. The recovered displacement from the collection of points is used to recover probability distribution functions (PDF's) of a vertical displacement time series. Using the PDF, we can leverage mean and variance of displacement values to understand the uncertainty of our measurements. Examples are provided that illustrate the effects of a variable drone height and variable sub-pixel disparity precision. We observe that the stereo vision displacement measurements greatly benefit from low drone heights and sub-pixel disparity precision. We conclude that within reasonable, and currently available, camera system parameters we are able to monitor sub-millimeter ground motion within several meters above the ground.

Key words: contactless seismic, stereo vision, acquisition

1 INTRODUCTION

Environmental impact, access, and acquisition cost are all practical factors to consider when designing a geophysical survey. In environmentally sensitive areas, for example, it may not be possible to perform a ground-based survey without disrupting or damaging a fragile environment or ecosystem. Treacherous terrain in mountainous or remote areas can lead to access and logistical issues for equipment and personnel. Limitations due to access and environmental concerns are accompanied by financial limitations. For example, deploying geophone arrays for exploration can involve a large amount of equipment, personnel, and time; all of which compound into large acquisition costs. In practice, cost can be reduced by optimizing acquisition design parameters such as receiver spacing, seismic source geometry, source signal length, and frequency content. Technological advancements can also reduce costs of acquisition and potentially open new opportunities for exploration; an excellent example is distributed acoustic systems (DAS). We advocate in this paper that recent developments in drones and computer vision have the potential to impact seismic data acquisition, in principle allowing for airborne seismic surveying systems at high speed and low cost.

Monitoring motion without contacting a surface can be performed using passive or active systems. Laser Doppler vibrometers (LDV) are an example of an active system, in which a known source signal is emitted and the phase of the return signal is analyzed to deduce the distance the signal travelled. LDV's have been used in the past to investigate the feasibility of remotely detecting ground motion from seismic waves (Berni, 1991, 1994). Another similar active system uses microwaves, and interferometry, to deduce vibrations and resonant frequencies in a structural engineering context such as in Stanbridge et al. (2000). In contrast to active systems, passive systems do not require a source for deducing motion and leverage high speed video cameras and ambient lighting. Here we focus on passive methods for deducing motion.

In this work, we advocate measuring distance variations to the ground with time using stereo vision theory. Stereo vision has been used for a long time to passively measure distance using two images recorded from offset cameras (Quam et al., 1972). Our left and right eyes allow for depth perception, just as two images taken from offset cameras allow for a distance estimate. To perform stereo vision, we record two digital images taken at the same time from calibrated cameras with a known offset. We then find points in each image that corre-

spond to each-other and measure the lateral shift, or disparity, between the points as observed from the two offset cameras. As we show later, this disparity is inversely proportional to the distance from the camera to the points observed in the images. Therefore, the process of finding distance from images using stereo vision requires us to (1) accurately identify correspondence between points in an image, and (2) accurately compute perceived shifts between corresponding points from two vintage points. This process is outlined in more detail in the Theory section.

A drawback of stereo vision is that small disparity errors result in large distance errors; we therefore must be able to precisely measure shifts between images in order to obtain a reliable distance measurement. To improve our ability to measure small shifts between images, we can amplify the shifts using a recent and exciting advancement in computer vision called motion magnification. Motion magnification allows us to remotely monitor visually imperceivable vibrations using passive high speed cameras (Wahbeh et al., 2003; Wadhwa et al., 2014; Rubinstein et al., 2014). The technique has been used by many others to solve various scientific engineering problems. Chen et al. (2014) uses the method for deducing structural information of a steel beam using video; a similar approach is taken by Shariati et al. (2015) to deduce natural structural vibration modes of other structures. Remotely monitoring heartbeats using video is another intriguing use of motion magnification (Wu et al., 2012). Davis et al. (2014) impressively recover sound from video alone by filming a bag of chips subject to tiny vibrations caused by someone speaking in its vicinity. It is our intention to apply motion magnification, in conjunction with stereo vision, in order to acquire seismic data without contacting the ground. In this report, we do not include motion magnification, however we note that contactless motion can benefit significantly from this new development.

In this paper, we demonstrate a method for sampling a seismic wavefield that has potential to reduce costs while providing exciting new opportunities for geophysical exploration. The method anticipates using drones and cameras as a means to acquire seismic data without contacting the ground surface. First, we discuss the underlying stereo vision theory used to determine ground displacement variations with time. Ground displacement values are recovered for a collection of points representing a vertically vibrating ground surface. Using the collection of ground displacement values, we recover a PDF of ground displacement, from which we deduce displacement mean and variance as a function of time. The mean and variance of the displacement PDF allow insight on the uncertainty of observations provided by the method. Using stereo vision theory, we then demonstrate motion recovery from video using synthetic stereo videos of a vibrating textured surface.

2 THEORY

We aspire to measure surface displacement versus time without making direct contact with the ground. Surface displacement can be described as the change in distance from a fixed point to a surface with time. In order to achieve our goal, we prepare to use a stereo video camera system which measures the distance to a vibrating textured ground surface. The stereo camera system is assumed to be at a known height above the ground, with each camera being aligned and separated by a known distance b. The vibrating textured surface is represented by a collection of points of varying initial height that move vertically according to a known displacement time series. The points representing the textured surface are defined in a coordinate system separate from the coordinate system of the camera; we refer to the coordinate system of the points as the world coordinate system. The camera may be moving or tilting as the ground moves. In our work, we do not assume a specific camera orientation in relation to the ground. To be clear, during our simulations the points representing the ground are displaced vertically with time in the world coordinate system.

The displaced points are observed at each time of the simulation by stereo cameras, which we simulate using a pinhole camera model, Figure 1. This model assumes an image sensor, or image plane, that is a distance f, the camera focal length, away from the focal point. The optical axis defines the Z-axis of the camera coordinate system and passes through the focal point and the center of the image sensor. The X and Y directions of the camera coordinate system are defined to run in two orthogonal directions along the image sensor. We project a collection of 3D points, which represent the textured ground surface, onto a 2D image sensor using this pinhole camera model. Appendix A outlines the theory behind projecting a 3D point in camera coordinates onto the 2D image plane using the pinhole camera model. In short, the position of a point in 3D space must firstly be expressed in camera coordinates; this can be achieved by using an appropriate rotation and translation from the world to the camera coordinate systems. The point is then projected onto the image plane using geometry derived from the pinhole camera model shown in Figure 1, which changes based on the specific camera used. The point projects to different pixel locations when viewed using a stereo camera system, and the difference in the pixel locations allows us to infer the depth to the point from the camera.

To demonstrate the stereo vision theory, we consider a stereo camera system observing a single point on the ground, as shown in Figure 2. The point in this schematic represents one of the many points we can use to represent a vibrating textured surface at a given time. The schematic in Figure 2 is only shown for the xz-plane of the camera coordinate system, the pixel location of a point on each camera only differs in the direction of the camera separation. To be clear, there are three coordinate systems in our simulations (1) the left camera, (2) right camera, and (3) world coordinate systems. The world coordinate system origin is defined at:

$$\mathbf{o}_w = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \tag{1}$$

with unit vectors in x, y, and z being $[1,0,0]^T$, $[0,1,0]^T$, and $[0,0,1]^T$, respectively. In the world coordinate system, the

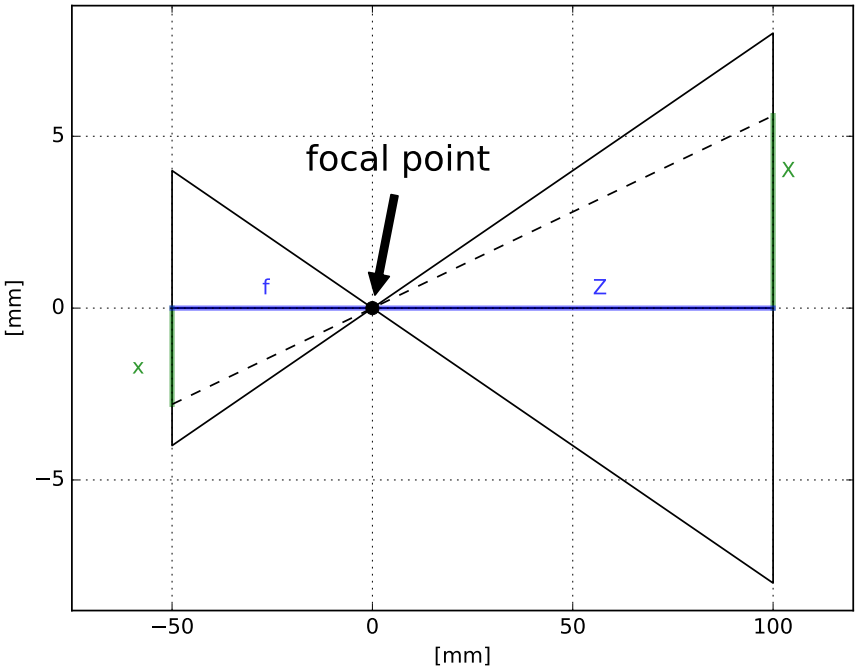


Figure 1. The pinhole camera model used to project a 3D point (X, Y, Z) onto a 2D image plane at point (x, y). The schematic for the xz-plane is shown here, the same schematic for the yz-plane exists. The image plane is on the left portion of the figure, a distance of f away from the focal point. The dashed ray shows the X and Z location of the 3D point in the camera coodinate system being projected onto the image plane to find the pixel location x. The same image in the yz-plane can be used to find the pixel location y. Any point along the dashed path projects onto the same point on the image plane.

positive z-direction is defined to be pointing down into the ground. We may define the left and right cameras to be any location in the world coordinate system; in our examples we define the left camera origin in the world coordinate system at:

$$ {}^{w}\mathbf{o}_l = \begin{bmatrix} 0 \\ 0 \\ -h \end{bmatrix}, \tag{2} $$

where the preceding superscript in the vector ${}^{w}\mathbf{o}_r$ indicates that the vector is in the *world* camera coordinate system. We note that h is the height of both the left and right cameras at time zero. The right camera origin is defined to be a distance b away from the left camera in the positive x-direction of the left camera coordinate system:

$$ {}^{l}\mathbf{o}_r = \begin{bmatrix} b \\ 0 \\ -h \end{bmatrix}, \tag{3} $$

where the superscript in the vector ${}^{l}\mathbf{o}_r$ indicates that the vector is in the *left* camera coordinate system. We can project a homogeneous point, denoted using the notation $\tilde{\mathbf{x}}$, in the left or right camera system to the world system using a known rotation and translation (see Appendix A). The homogeneous point representing the right camera origin in the world frame ${}^{w}\tilde{\mathbf{o}}_r$ may be represented as a 3D point ${}^{w}\mathbf{o}_r$ by dividing all of the elements of by the fourth element:

$$ {}^{w}\tilde{\mathbf{o}}_r = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \rightarrow {}^{w}\mathbf{o}_r = \begin{bmatrix} \frac{x_1}{x_4} \\ \frac{x_2}{x_4} \\ \frac{x_3}{x_4} \end{bmatrix} = \begin{bmatrix} o_{r,x} \\ o_{r,y} \\ o_{r,z} \end{bmatrix}, \tag{4} $$

where we use $o_{r,x}$, $o_{r,y}$, and $o_{r,z}$ to denote x, y, and z coordinates of the right camera in the world coordinate system. The right camera origin in left camera coordinates can be written in the world coordinate system as:

$$ {}^{w}\tilde{\mathbf{o}}_r = {}^{w}_{l}\mathbf{H}\,{}^{l}\tilde{\mathbf{o}}_r. \tag{5} $$

We assume that the rotation and translation from the world to camera coordinate system is known, which is used to form the transformation ${}^{w}_{l}\mathbf{H}$ from left camera to world coordinates (see Appendix A). Any point in the left coordinate system can be transformed to the world coordinate system using ${}^{w}_{l}\mathbf{H}$; a transformation from world to left camera coordinates can be performed using ${}^{l}_{w}\mathbf{H} = {}^{w}_{l}\mathbf{H}^{-1}$. Equivalently, we may transform any point in the right camera to the left using ${}^{l}_{r}\mathbf{H}$ since the two camera coordinate systems are aligned and separated by a known distance. For completeness, a homogeneous point in the right camera ${}^{r}\tilde{\mathbf{x}}$ may be transformed to the world coordinate system by:

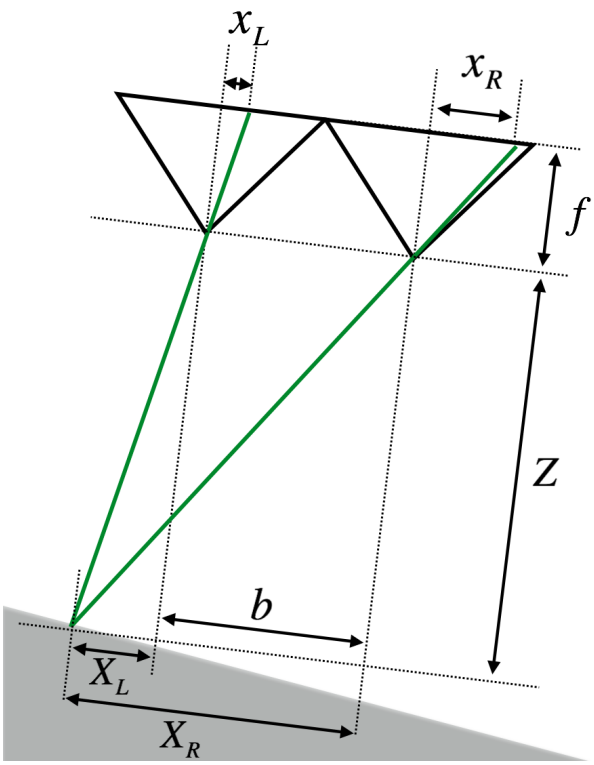


Figure 2. A schematic of stereo cameras viewing the ground from above. The two cameras have optical axis that are aligned but separated by a distance b. The dashed paths show where a point on the ground projects on the left and right cameras. Note that the cameras do not have to be aligned with the ground surface, however the recovered depth Z is measured from the cameras orthogonal to the lines connecting the pinholes.

$$ {}^{w}\tilde{\mathbf{x}} = {}^{w}_{l}\mathbf{H}\,{}^{l}_{r}\mathbf{H}\,{}^{r}\tilde{\mathbf{x}}. \tag{6} $$

Consider a point $\mathbf{p}$ on the ground, which we project onto each camera, located at ${}^{l}\mathbf{p} = [X_l, Y_l, Z_l]^T$ in the left camera coordinate system and ${}^{r}\mathbf{p} = [X_r, Y_r, Z_r]^T$ in the right camera coordinate system. Both of these coordinates can be transformed to the world coordinate system using the known transformations ${}^{w}_{l}\mathbf{H}$ and ${}^{w}_{r}\mathbf{H}$, respectively. Note, that the definition of the point in the left and right coordinate system still holds if the cameras are rotated, pitched, or tilted, as long as the cameras remain aligned and separated by a known distance b in the x-direction of the left camera coordinate system. Since the camera axes are only separated in the x-direction of the camera coordinate systems, we know that $Y_l = Y_r = Y$, $Z_l = Z_r = Z$, and $y_l = y_r$; the pixel location of the point is only different in the x-direction of the camera (i.e. along the horizontal direction of the image sensors on each camera). Here, Z represents the distance measured along the z-axis of the left and right camera. The pixel locations of the point projected onto the left and right cameras are denoted $[x_l, y_l]$ and $[x_r, y_r]$, respectively. In the plane defined using the point $\mathbf{p}$ in 3D space and its projections on the left and right images at pixel locations $[x_l, y_l]$ and $[x_r, y_r]$, (Figure 2) we find the following relationships based on triangle similarity:

$$ \begin{aligned} \frac{x_l}{f} &= \frac{X_l}{Z} \rightarrow x_l = \frac{fX_l}{Z} \\ \frac{x_r}{f} &= \frac{X_r}{Z} \rightarrow x_r = \frac{fX_r}{Z}. \end{aligned} \tag{7} $$

Since we know the cameras are separated by a distance $b = X_r - X_l$, we know that the difference in the x-pixel location on the camera image planes is:

$$ d = x_r - x_l = \frac{fX_r - f(X_r - b)}{Z} = \frac{fb}{Z}. \tag{8} $$

The difference in pixel projection locations between the left and right images is called disparity. In this case, the disparity is proportional to the focal length and camera separation, but is inversely proportional to the distance Z, as shown in equation 8. For each time, we deduce the distance from each point to the cameras representing the vibrating ground surface using the disparity-depth relationship in equation 8.

Given disparity, we find the depth a point is from the cameras Z, then compute the points position in the right camera coordinate system using the relationships in equation 7. For the position of the point in the right camera coordinate system, we obtain:

$$X_r = \frac{Zx_r}{f}$$
$$Y_r = \frac{Zy_r}{f}. \tag{9}$$

Finally, we transform the point ${}^r\mathbf{p}$, in homogeneous coordinates, from the right camera coordinate system to the world coordinate system using the known transformation ${}^w_r\mathbf{H}$:

$${}^r\mathbf{p} = \begin{bmatrix} X_r \\ Y_r \\ Z \end{bmatrix} \rightarrow {}^w\tilde{\mathbf{p}} = {}^w_r\mathbf{H}\,{}^r\tilde{\mathbf{p}} \tag{10}$$

The relationship between disparity and distance shows that a small error in disparity results in a large error in distance. In practice this poses a problem since the pixel coordinates are discrete. The pixel size is therefore a function of the camera resolution and the distance the cameras are from the point on the ground. As the camera resolution increases, the representative pixel size decreases which allows us to more accurately represent disparity and therefore distance. The same beneficial decrease in pixel size can be achieved by moving the cameras closer to the vibrating points. We therefore seek to have the highest resolution cameras as close to the vibrating ground surface as possible. In practice, if the cameras were mounted on a drone, a height between 1-10 m would be reasonable; we also easily expect resolutions close to that of a compact camera ranging from 1-10 megapixels.

The derived relationship between disparity and distance in equation 8 is a special case based on additional assumptions about camera geometry and calibration, point visibility and correspondence. The relationship becomes more complicated when the left and right camera orientations are not aligned; however the simpler geometry we assume here is realizable in many stereo camera systems. Also, we assume a pinhole camera model for two aligned and calibrated cameras with known separation, and therefore rely on correct camera calibration, eliminating lens distortion, for this theory to be applicable. We also assume that the point we estimate distance to is visible from both cameras. If the point is occluded or out of the field of view of a camera, then we are not able to compute a disparity. In practice, we view the vibrating points on the ground from above which minimizes occlusion of points. Another implicit assumption is that sufficient lighting allows us to view the point using the cameras. After all, we would not be able to record an image in a pitch black room. The last and largest assumption we have made is the knowledge of correspondence of points in both images. When performing stereo vision using many points, we cannot hope to achieve accurate results if the disparity is computed for points that do not correspond to one another.

Practically speaking, there are two main strategies for computing disparities in stereo images using (1) sparse correspondence of points and (2) dense correspondence. Sparse correspondence methods have been in development since the 1970's and tend to focus on tracking several features within images, such as edges (Szeliski, 2011). In contrast, dense correspondence methods focus on identifying a much larger number of disparity calculations within an entire image. An excellent summary of many dense correspondence algorithms can be found in Scharstein and Szeliski (2002). In our work, we simulate a dense set of disparities and assume the correspondence between points is known; we leave the correspondence problem for future work. The theory presented here is general enough to be applied when using either sparse or dense correspondence methods. However, we anticipate using dense correspondence in images to estimate the distance to the ground in each frame of a video. Here, we use a large number of distance estimates to help describe the motion of the ground in a statistical way using a probability density function (PDF) of the recovered ground displacement.

We note that pixel coordinates $[x_l, y_l]$ and $[x_r, y_r]$ in the left and right camera images are represented as integers, and therefore limit the precision of the disparity, and therefore distance, we may measure. Sub-pixel disparity precision allows for a more precise recovery of distance. Sub-pixel disparity can be computed by interpolating between disparities, using a best local planar fit, or by area matching of template, to name a few methods (Birchfield and Tomasi, 1998; Scharstein and Szeliski, 2002; Shimizu and Okutomi, 2005). An alternative to sub-pixel dense disparity algorithms would be to magnify the shifts between the left and right images. We may use motion magnification as a way to magnify subtle sub-pixel shifts between left and right videos (Wahbeh et al., 2003; Wadhwa et al., 2014; Rubinstein et al., 2014; Davis et al., 2014). The stereo vision process discussed here does not change as, we would simply magnify motion in the left and right videos before computing disparities using equation 8. However, the amount of motion magnification we use would have to be undone before computing distances from the disparities. For example, if we magnify motion by 100% then the motion disparities would twice too large, therefore we would divide the disparities by two before computing distances using equation 8.

Before performing numerical experiments to find PDF's of displacement, we analytically determine how error in projected pixel locations translates to error in the disparity. We may quantify the change in disparity given a small change, or error, in the pixel locations Δx_r or Δx_l using total derivative of equation 8. Total derivative of a function $z = f(x, y)$ can be approximated as:

$$\Delta z \approx \frac{\delta f}{\delta x}\Delta x + \frac{\delta f}{\delta y}\Delta y, \tag{11}$$

for small changes in x and y denoted as Δx and Δy, respectively. Considering, the error in disparity, Δd, from equation 8 we then have:

$$\begin{aligned} \Delta d &= \frac{\delta d}{\delta x_r}\Delta x_r + \frac{\delta d}{\delta x_l}\Delta x_l \\ &= \Delta x_r - \Delta x_l, \end{aligned} \tag{12}$$

where Δx_r and Δx_l are errors in the point projection on the right and left camera, respectively. We expect the errors in the

left and right cameras to be independent and zero mean. The expected value of the disparity error then becomes:

$$\mu_d = E[\Delta d] = E[\Delta x_r - \Delta x_l] = E[\Delta x_r] - E[\Delta x_l] = 0. \tag{13}$$

The variance of the disparity error becomes:

$$\begin{aligned} \sigma_d^2 &= E[(\Delta d - \mu_d)^2] \\ &= E[(\Delta d^2)] \\ &= E[(\Delta x_r - \Delta x_l)^2] \\ &= E[{\Delta x_r}^2] + E[{\Delta x_l}^2] - 2E[\Delta x_r \Delta x_l] \\ &= \sigma_L^2 + \sigma_R^2 - 2\sigma_{LR}^2 \\ &= \sigma_L^2 + \sigma_R^2. \end{aligned} \tag{14}$$

We conclude that the mean of the disparity error to be zero given independent and zero mean errors in the left and right camera pixel projection coordinates. The variance of the disparity is the sum of the variance of the errors in the left and right cameras.

We express the error in the depth estimate given an error in disparity defined in equation 8, as:

$$\Delta Z = \frac{\delta Z}{\delta d}\Delta d = -\frac{fb}{d^2}\Delta d. \tag{15}$$

We see that the absolute value of the distance error is proportional to $\frac{fb}{d^2}$. Intuitively, this result implies that an increase in camera separation b or focal length f increases the distance error for a given disparity d. However, the disparity dominates the depth error as $\frac{1}{d^2}$, implying that large disparities can drastically decrease the depth error. The mean of the error in depth Z becomes,

$$\mu_Z = E[\Delta Z] = -\frac{fb}{d^2}E[\Delta d] = 0, \tag{16}$$

while the variance of the error in depth Z is,

$$\begin{aligned} \sigma_Z^2 &= E[(\Delta Z - \mu_Z)^2] \\ &= E[\Delta Z^2] \\ &= E\left[\left(-\frac{fb}{d^2}\right)^2 \Delta d^2\right] \\ &= \left(\frac{fb}{d^2}\right)^2 \sigma_d^2. \end{aligned} \tag{17}$$

The standard deviation in the depth error is proportional to the quantity $\left(\frac{fb}{d^2}\right)^2$. This quantity is dominated by the effects of d^{-4}; if the disparity increases, then the variance in the depth error rapidly decreases.

Mean and variance summarize the behavior of the distance error, however a more complete representation of the displacement can be expressed using PDF's. At every time, the cameras observe the ground, represented by a number of points on the ground, and use the disparity of points to compute the distance to the ground. Using many points on the ground, the distance measurements which can be represented as a PDF characterizing ground displacement. Displacement is computed by subtracting the distance estimate for each point at the initial time. Therefore, we recover many PDF's of displacement measurements, representing the recovered displacement versus time. In the next section, we analyze the recovered PDF's of displacement in order to better understand the statistical behavior of the proposed method for observing motion without contact.

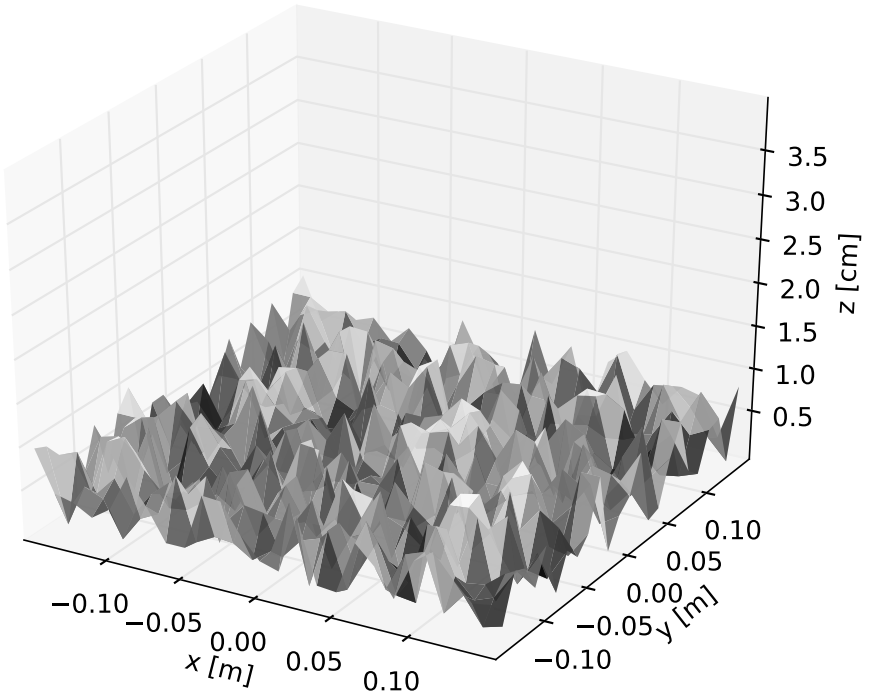
3 NUMERICAL EXAMPLES

We simulate stereo cameras viewing a textured surface to analyze the feasibility of detecting small displacements. The textured ground surface is represented using a collection of points of varying initial height. Vertical ground motion is simulated by moving the points vertically according to a known vertical displacement signal. The textured surface points are projected onto two cameras with known orientation above the ground as described in the previous Theory section. From the recovered vertical displacement for each point, we recover a distribution of displacement at each time step of the simulation. We analyze the effects of (1) the number of points considered on the vibrating ground, (2) the accuracy of the disparity, or shifts, and (3) camera height above ground.

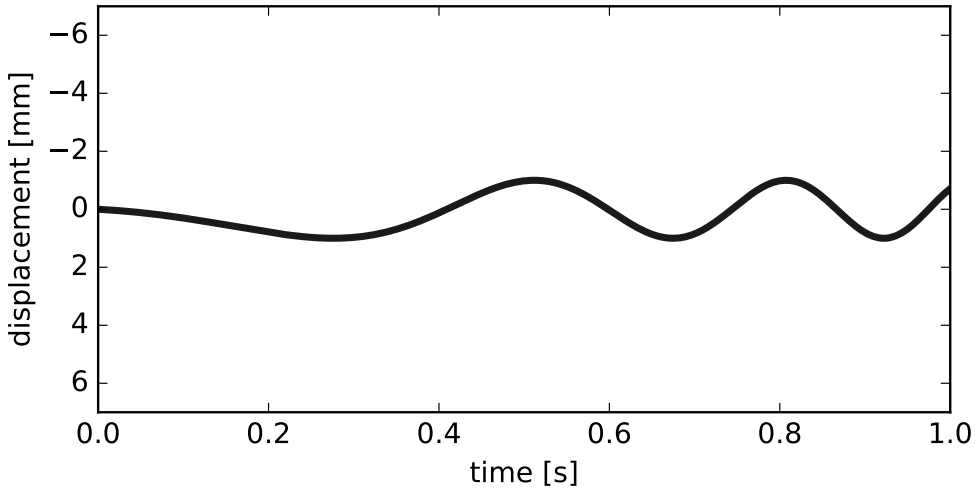
We simulate results using specifications for a commercially available stereo camera, the DUO MC by Code Laboratories, which has a focal length of 2 mm, is capable of a resolution 752×480 pixels, and has a pixel sensor size of 6 μm. Initially, we utilize 1000 points to representing a textured ground surface spanning 0.3 m in both x and y directions. The initial height of all points is randomly distributed between +/- 1 cm. The points are displaced vertically according to a chirp from 0.25 Hz to 1.0 Hz shown in Figure 3(b).

We investigate the effects of including an increasing number of points to represent the textured ground surface. As discussed in the Theory section, the number of points we choose to consider when performing stereo vision depends on our choice of correspondence method; sparse methods use fewer points than dense methods. We consider having 10, 100, and 1000 points tracked during our stereo vision simulation in Figure 4. In Figure 4 we display the PDF of the recovered displacement using stereo vision, with the true displacement dashed, the mean of the PDF at each time traced in solid, and one standard deviation above and below the mean in light gray. As the number of points increases, mean of the displacement PDF converges to the true displacement. This behavior of the mean aligns with the error discussion in the previous Theory section, namely that the expected error of the recovered distance tends to zero as the number of points increases.

A notable characteristic of the recovered displacement PDF in Figure 4 is its horizontal banding. The banding is due to a precision limitation which effectively discretized the possible displacements we may recover. Considering a focal length f, pixel sensor size s, and height h of the DUO camera, we know that one pixel spans a distance on the ground x_{pixel} corresponding to:



(a)

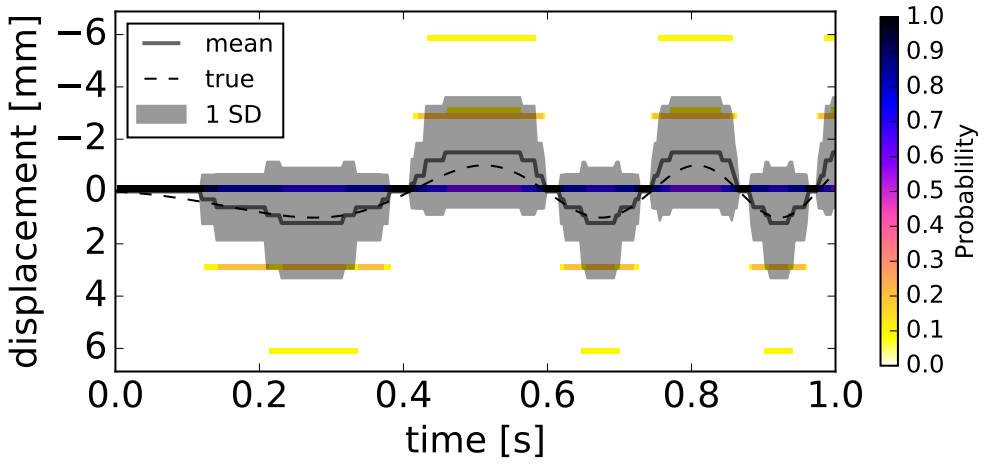


(b)

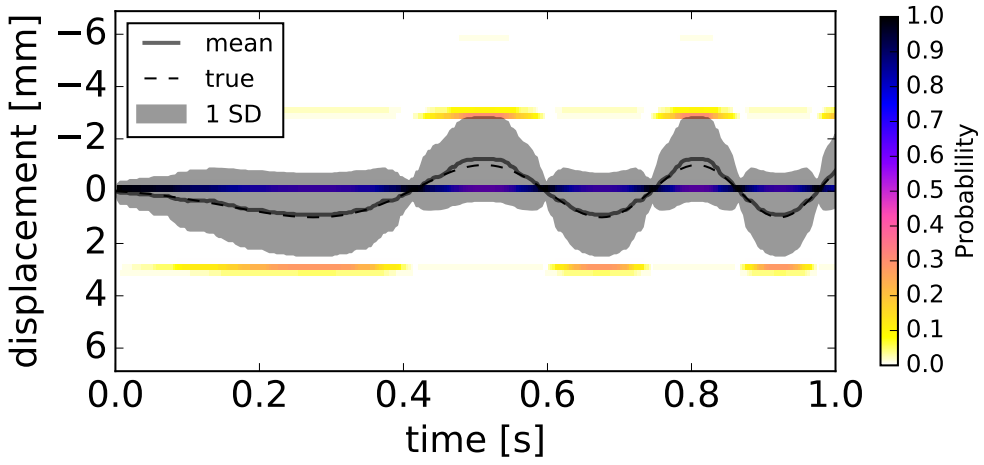
Figure 3. (a) Textured ground representation and (b) true displacement of textured ground as a sweep from 0.25 Hz to 1.0 Hz. Lower displacement frequencies are chosen for display purposes only. The points representing the textured ground are perturbed from their initial height using the displayed sweep.

$$\frac{x_{pixel}}{2} = \frac{sh}{f} = \frac{\left(6\ \mu\text{m} \times \frac{1\ \text{mm}}{1000\ \mu\text{m}}\right) \times 1000\ \text{mm}}{2\ \text{mm}} = 3\ \text{mm}$$
$$\rightarrow x_{pixel} = 6\ \text{mm}. \tag{18}$$

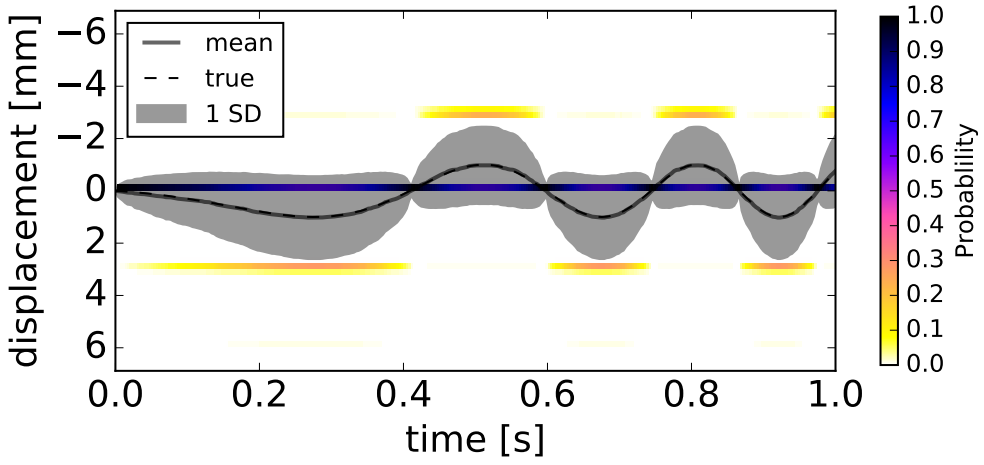
As we use the pinhole camera model geometry in Figure 1 with $x = \frac{x_{pixel}}{2}$, all distance estimates we make are integer multiples of x_{pixel}, since this is our lowest precision. However, subpixel disparity estimate methods do exist and we can round off our disparity estimates to see their benefits (Scharstein and



(a)



(b)



(c)

Figure 4. Displacement recovery at a camera height of 1 m with a half of a pixel precision using (a) 10, (b) 100, and (c) 1000 points representing the textured ground. Each point is tracked using stereo vision; tracking more points on the ground lowers the variance in the recovered displacement PDF.

Szeliski, 2002). To observe the benefits of sub-pixel precision, we show the recovered displacement for no sub-pixel precision, half of a pixel, and one tenth of a pixel precision in Figure 5. We observe the banding at integer multiples of 6 mm when no sub-pixel precision is used in Figure 5(a); whereas Figure 5(b) and Figure 5(c) show an increased number of bands spaced closer together due to sub-pixel disparity precision. The bands for sub-pixel estimates are closer together because of the increased precision in recoverable disparity values. Similar to the previous simulation results in Figure 4, in all three precision cases the mean value of the recovered displacement PDF is near the true value. However, the standard deviation of the recovered PDF greatly decreases when we allow sub-pixel precision. This decrease in standard deviation is expected, given our error variance estimates for displacement in equation 17.

The camera height directly effects the effective pixel size, and therefore impacts the discretization of displacement estimates we recover. We desire to understand the feasibility of using stereo vision on drones, and therefore investigate heights a drone could easily obtain. For a scenario with one tenth of a pixel precision, Figure 7, as the camera height decreases, the number of bands in the displacement increases which is caused by a decrease in effective pixel size. In Figure 7, we observe that the mean of the displacement PDF tracks the true displacement for all heights; lower heights exhibit similar mean values, but higher variance. The same behavior is observed when the camera separation is increased (Figure 6). For a given camera to ground distance Z, as the camera separation b increases the disparity also increases (equation 8), greatly lowering the error in variance of the recovered distance (equation 17).

Based on our results, we believe that a stereo camera is a viable option for monitoring sub-millimeter ground motion. The simulation parameters we use reflect realistic drone and camera parameters. Here, we only show recovered displacement for vertical ground motion +/- 1 mm. In order to observe smaller ground motions, we must decrease the effective pixel size by either increasing the camera resolution, lowering the camera height, or using sub-pixel disparity precision methods. Alternatively, we may use motion magnification in order to amplify the shifts in the left and right images for computing displacement with higher precision.

4 CONCLUSIONS

Our simulations show that a stereo camera is a viable option for monitoring sub-millimeter vertical ground motion at realistic camera heights. The methods we use assume a known camera orientation with respect to the ground. The assumption is realistic given that orientation information of a drone-camera system is routinely monitored via GPS and Inertial Measurement Units (IMUs) which record accelerometer, magnetometer, and gyroscope data. In terms of acquisition geometry and hardware, in order to improve our motion estimates using stereo vision we must decrease the effective pixel size by moving the camera closer to the ground or by increasing the resolution of the camera. We may further improve the motion estimates using processing techniques that recover many dense disparity values, and importantly recover sub-pixel shifts. Recovering many disparity values allows us to form a PDF of ground displacement, which can be used to more completely understand the statistics of the measured data in terms of mean or variance, as opposed to conventional geophone data. Future work on the processing side includes motion magnification to improve disparity estimates.

5 ACKNOWLEDGEMENTS

We thank the sponsor companies of the Center for Wave Phenomena, whose support made this research possible.

REFERENCES

Berni, A. J., 1991, Remote seismic sensing. (US Patent 5,070,483).

———, 1994, Remote sensing of seismic vibrations by laser Doppler interferometry: Geophysics, **59**, 1856–1867.

Birchfield, S., and C. Tomasi, 1998, A pixel dissimilarity measure that is insensitive to image sampling: Pattern Analysis and Machine Intelligence, IEEE Transactions on, **20**, 401–406.

Chen, J. G., N. Wadhwa, Y.-J. Cha, F. Durand, W. T. Freeman, and O. Buyukozturk, 2014, Structural modal identification through high speed camera video: Motion magnification, *in* Topics in Modal Analysis I: Springer, **7**, 191–197.

Davis, A., M. Rubinstein, N. Wadhwa, G. J. Mysore, F. Durand, and W. T. Freeman, 2014, The visual microphone: Passive recovery of sound from video: ACM Transaction on Graphics, **33 (4)**, 79:1–79:10.

Quam, L. H., R. B. Tucker, S. Liebes Jr, M. J. Hannah, and B. G. Eross, 1972, Computer interactive picture processing: Technical Report 166, Stanford Artificial Intelligence Project Memo.

Rubinstein, M., et al., 2014, Analysis and visualization of temporal variations in video: PhD thesis, Massachusetts Institute of Technology.

Scharstein, D., and R. Szeliski, 2002, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms: International journal of computer vision, **47**, 7–42.

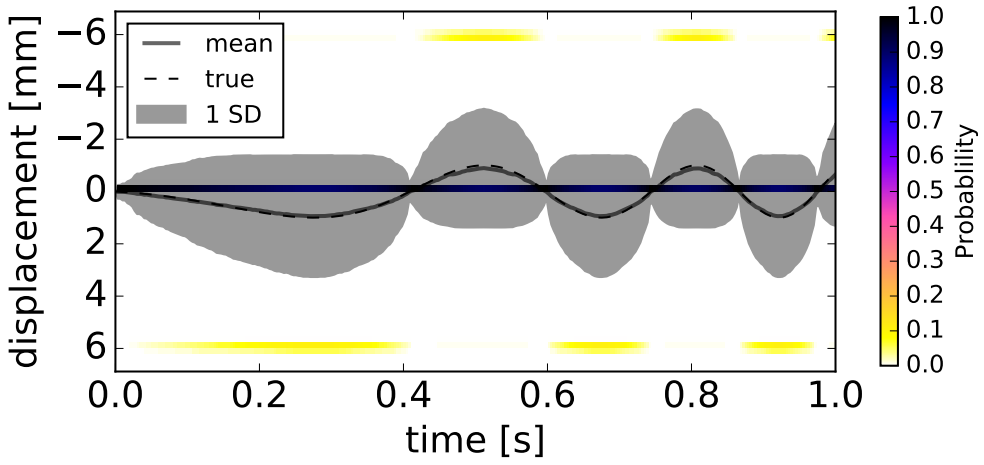
Shariati, A., T. Schumacher, and N. Ramanna, 2015, Eulerian-based virtual visual sensors to detect natural frequencies of structures: Journal of Civil Structural Health Monitoring, 1–12.

Shimizu, M., and M. Okutomi, 2005, Two-dimensional simultaneous subpixel estimation for area-based matching: Systems and Computers in Japan, **36**, 1–11.

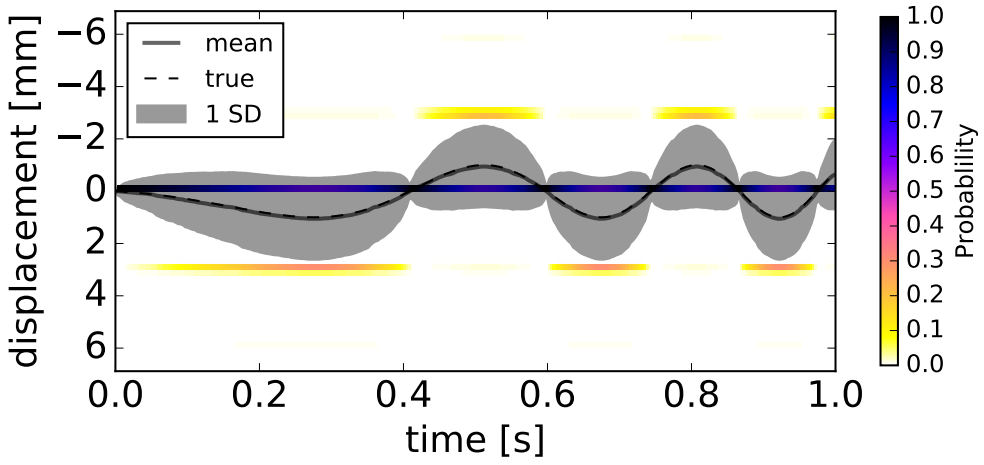
Stanbridge, A., D. Ewins, and A. Khan, 2000, Modal testing using impact excitation and a scanning ldv: Shock and Vibration, **7**, 91–100.

Szeliski, R., 2011, Computer vision: Algorithms and applications: Springer.

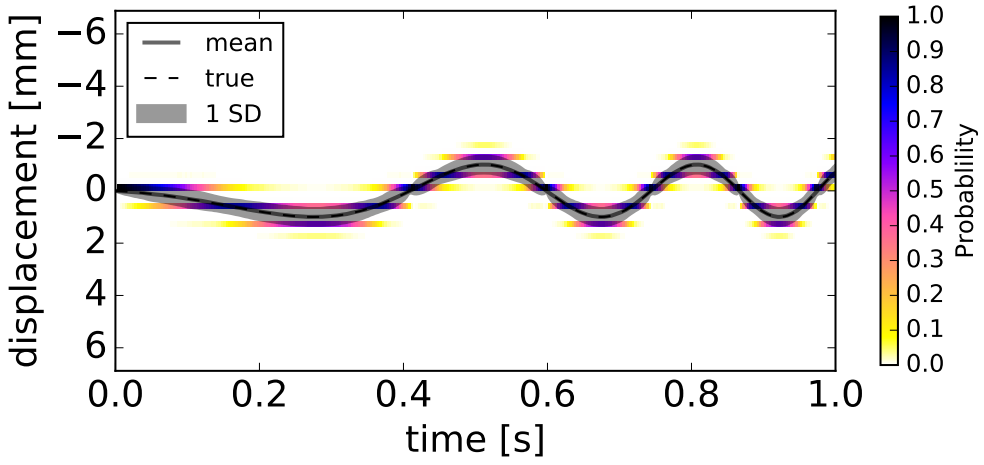
Wadhwa, N., M. Rubinstein, F. Durand, and W. T. Freeman,



(a)



(b)



(c)

Figure 5. Displacement recovery at a camera height of 1 m with 1000 points representing the ground and (a) no sub-pixel precision, (b) sub-pixel precision of one half of a pixel, and (c) sub-pixel precision of one 10th of a pixel. Increasing sub-pixel disparity precision decreases the variance of the recovered motion.

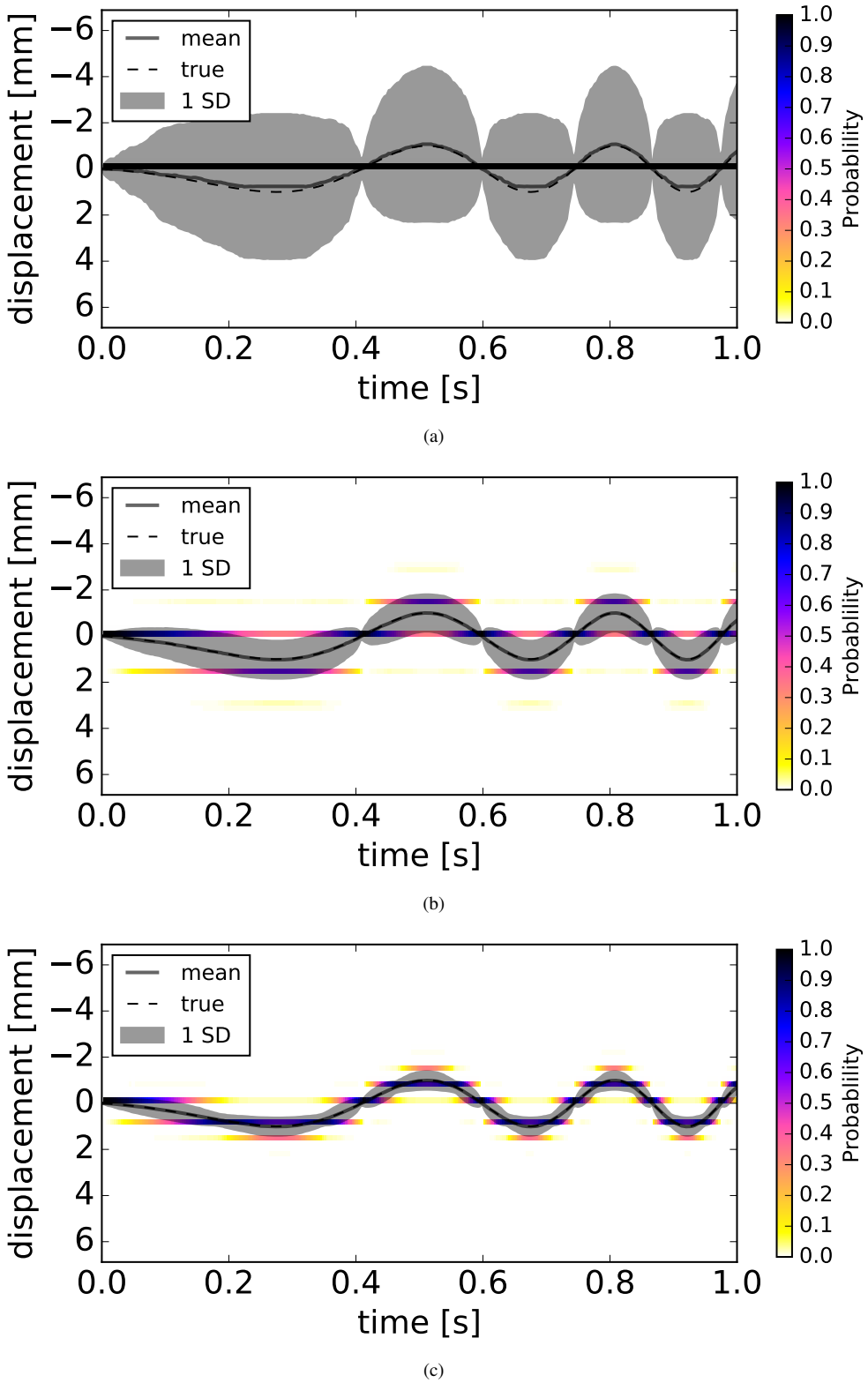
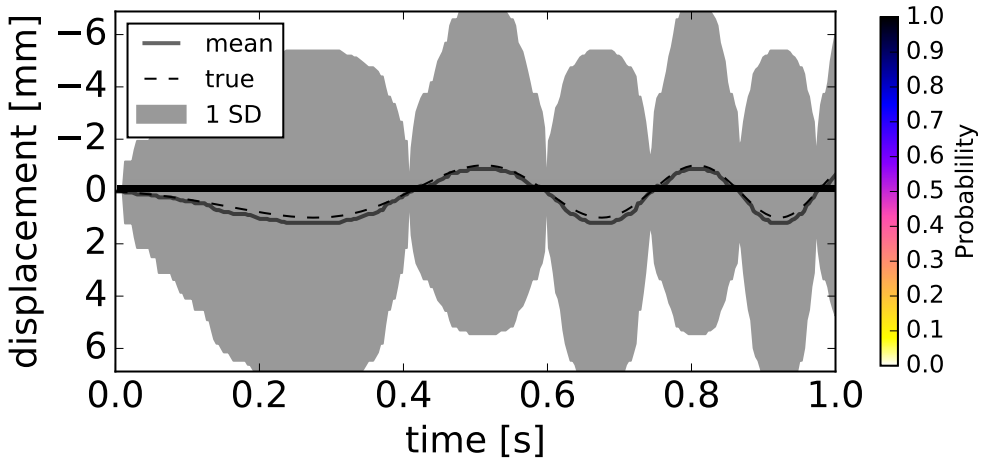
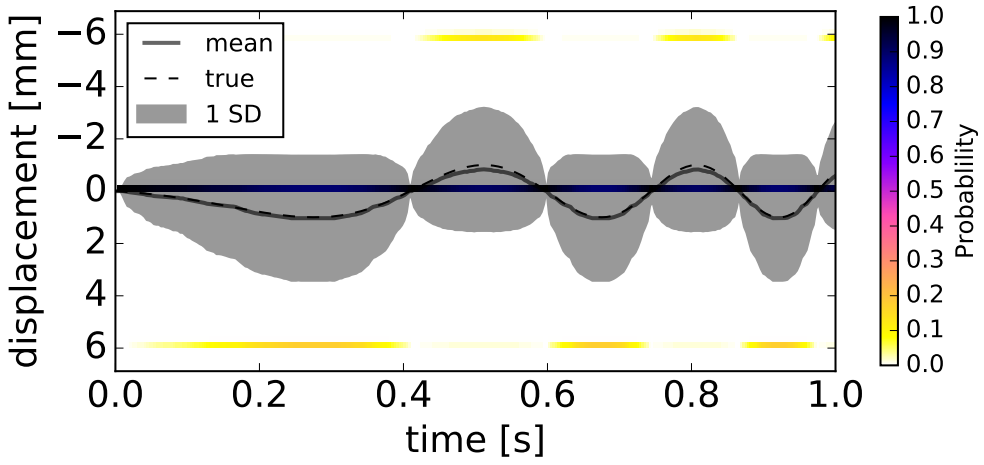


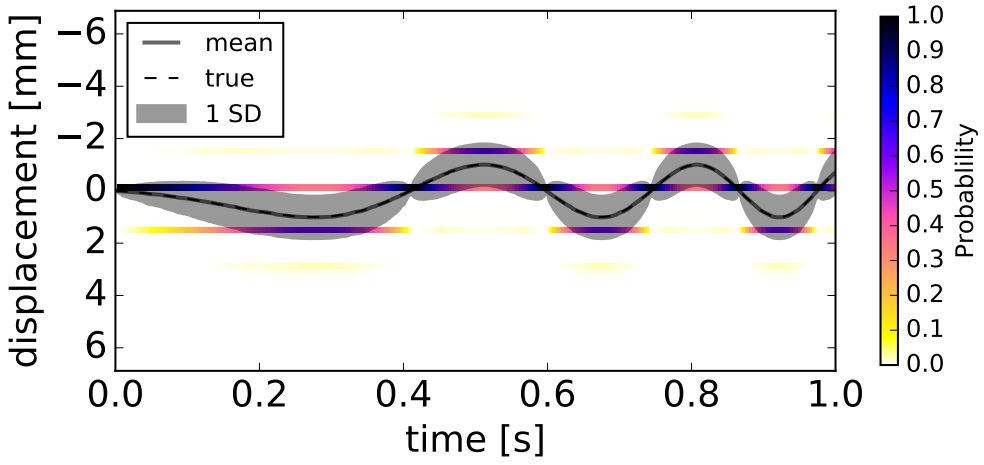
Figure 6. Displacement recovery at a camera height of 1 m with 1000 points and a quarter of a pixel disparity precision for a camera separation b of (a) 100 cm, (b) 0.5 m, and (c) 1 m. Increasing the camera separation decreases the variance of the recovered motion as expected from equation 17.



(a)



(b)



(c)

Figure 7. Displacement recovery at a camera height of 1 m with a quarter of a pixel precision for heights (a) 5 m, (b) 2 m, and (c) 1 m. Lower camera heights exhibit smaller representative pixel size, increasing precision of recovered motion.

2014, Riesz pyramids for fast phase-based video magnification.

Wahbeh, A. M., J. P. Caffrey, and S. F. Masri, 2003, A vision-based approach for the direct measurement of displacements in vibrating systems: Smart Materials and Structures, **12**, 785.

Wu, H.-Y., M. Rubinstein, E. Shih, J. V. Guttag, F. Durand, and W. T. Freeman, 2012, Eulerian video magnification for revealing subtle changes in the world.: ACM Trans. Graph., **31**, 65.

6 APPENDIX A: POINT PROJECTION AND STEREO VISION

This appendix provides, the theory behind projecting a 3D point onto a 2D image. This theory is not new, and can be found in standard computer vision textbooks; the notation we use here aligns with that used in Szeliski (2011). To begin, define a 2D rotation matrix, ${}^{b}_{a}\mathbf{R} \in \mathbb{R}^{2\times 2}$, that performs a rotation from coordinate system, or frame, $\{A\}$ to frame $\{B\}$ as:

$$ {}^{b}_{a}\mathbf{R} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}, \tag{19} $$

where θ is the angle from the x-axis of frame $\{A\}$ measured relative to the x-axis of frame $\{B\}$. The notation we use indicates "from" frame $\{A\}$ in the preceding subscript and "to" frame $\{B\}$ in the preceding superscript. The rotation matrix is an orthonormal matrix. The rotation matrix from $\{B\}$ to $\{A\}$ is:

$$ {}^{a}_{b}\mathbf{R} = \left({}^{b}_{a}\mathbf{R}\right)^{-1} = \left({}^{b}_{a}\mathbf{R}\right)^{T} \tag{20} $$

When transforming from one coordinate frame to another, we must know the vector pointing from the origin of frame $\{A\}$ to the $\{B\}$ frame origin. We denote this point as ${}^{b}\mathbf{t}_{aorg}$, which can be read "the $\{A\}$ frame origin expressed in the $\{B\}$ frame." So the transformation of a point in the $\{A\}$ frame, ${}^{a}\mathbf{x}$, to the $\{B\}$ frame can be performed by applying the rotation then translation:

$$ {}^{b}\mathbf{x} = {}^{b}_{a}\mathbf{R}\,{}^{a}\mathbf{x} + {}^{b}\mathbf{t}_{aorg}. \tag{21} $$

More simply put, one can represent a transformed coordinate $\mathbf{x}_{trans}$ using rotation followed by a translation a vector $\mathbf{x}$ as:

$$ \mathbf{x}_{trans} = \mathbf{R}\mathbf{x} + \mathbf{t}. \tag{22} $$

We represent points using homogeneous coordinates to make notation simpler. When using homogeneous coordinates, we append 1 as an extra element to the vector defining a given 2D point. We may represent a given homogeneous point $\tilde{\mathbf{x}}$ as:

$$ \tilde{\mathbf{x}} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} sx \\ sy \\ s \end{bmatrix}. \tag{23} $$

Homogeneous vectors differing by scale alone are considered equivalent. This allows us to rewrite the 2D-2D transformation:

$$ {}^{b}\mathbf{x} = {}^{b}_{a}\mathbf{R}\,{}^{a}\mathbf{x} + {}^{b}\mathbf{t}_{aorg} \tag{24} $$

as,

$$ {}^{b}\tilde{\mathbf{x}} = \begin{bmatrix} {}^{b}_{a}\mathbf{R} & {}^{b}\mathbf{t}_{aorg} \\ \mathbf{0} & 1 \end{bmatrix} {}^{a}\tilde{\mathbf{x}}, \tag{25} $$

or,

$$ {}^{b}\tilde{\mathbf{x}} = {}^{b}_{a}\mathbf{H}\,{}^{a}\tilde{\mathbf{x}}. \tag{26} $$

Here, $\mathbf{0}$ (boldface zero) indicates a row vector with the same length as the number of columns of $\mathbf{R}$. The notation used here seems cumbersome at first, however it simplifies the notation for transformations between multiple frames. For example, if we know the rotation and translation between the $\{A\}$, $\{B\}$, and $\{C\}$ frames we can transform a point in frame $\{C\}$ to $\{A\}$ by:

$$ {}^{a}\tilde{\mathbf{x}} = {}^{a}_{b}\mathbf{H}\,{}^{b}_{c}\mathbf{H}\,{}^{c}\tilde{\mathbf{x}} \tag{27} $$

Note that the preceding superscripts and subscripts cancel when using multiple transformations. Projections from 3D to 3D points is similar to 2D-2D transformations only there are more degrees of freedom for the transformation. A rotation and translation of homogeneous coordinates is still performed in a 3D-3D transform as in 2D-2D transforms. So a 3D-3D transformation of ${}^{a}\mathbf{x}$ in frame $\{A\}$ to ${}^{b}\mathbf{x}$ in frame $\{B\}$ could be performed by:

$$ {}^{b}\tilde{\mathbf{x}} = {}^{b}_{a}\mathbf{H}\,{}^{a}\tilde{\mathbf{x}}. \tag{28} $$

Here, the rotation matrix, ${}^{a}_{b}\mathbf{R} \in \mathbb{R}^{3\times 3}$, from B-to-A is still an orthogonal matrix. The matrix can be defined in a variety of ways. One way is to define rotation about the x, y, and z-axis and then form rotations in x, y, and z in a predefined order (the order differs with convention). Another way is to define a single vector, or direction, about which all coordinate axis are rotated. This second way is the so-called "equivalent angle-axis" and is more complicated but more robust. Forming the 3D rotation matrix is a common operation, however the details are outside the scope of this document.

One nifty property of the matrix ${}^{b}_{a}\mathbf{R}$ is that it's columns are the unit vectors of $\{A\}$ projected onto the $\{B\}$ frame. Denoting this property using dot products we can say:

$$ {}^{b}_{a}\mathbf{R} = \left[{}^{b}\hat{\mathbf{x}}_{a},\, {}^{b}\hat{\mathbf{y}}_{a},\, {}^{b}\hat{\mathbf{z}}_{a}\right]. \tag{29} $$

Alternatively, the rows of ${}^{a}_{b}\mathbf{R}$ are the unit vectors of $\{B\}$ projected onto the $\{A\}$ frame. We can write this as:

$$ {}^{b}_{a}\mathbf{R} = \begin{bmatrix} {}^{a}\hat{\mathbf{x}}_{b}^{T} \\ {}^{a}\hat{\mathbf{y}}_{b}^{T} \\ {}^{a}\hat{\mathbf{z}}_{b}^{T} \end{bmatrix}. \tag{30} $$

Depending on the information available, sometimes defining the rotation matrix by the projection of unit vectors is convenient. In our numerical examples, this property of the rotation matrix is used.

In order to project 3D points onto a 2D image, we adopt a 3D-2D transform of homogeneous coordinates. The transformation of a 3D point $[X, Y, Z, 1]^T$ to a 2D homogeneous point $\tilde{\mathbf{x}}$ can be performed by:

$$ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \rightarrow \tilde{\mathbf{x}} = \begin{bmatrix} X/Z \\ Y/Z \\ 1 \end{bmatrix}. \tag{31} $$

This transformation is valid because we are using homogeneous coordinates. Following the 3D-2D transformation, we must compensate for the geometry of a given camera recording the image; namely the focal length and image sensor size. We assume a simple pinhole camera model shown in Figure 1 which, by similar triangle geometry, allows us to describe a transform a 3D point in the camera coordinate frame $[X, Y, Z]$ into an image point $[x, y]$ by:

$$ x = f_x(X/Z) + c_x, \text{ and } y = f_y(Y/Z) + c_y, \tag{32} $$

where f_x, f_y, c_x, and c_y are the focal lengths and image center coordinates, respectively. This transformation can be performed using a so-called intrinsic camera matrix $\mathbf{K}$ following a 3D-2D transformation as described in equation 31. We show this below:

$$ \begin{aligned} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} &= \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \\ &= \begin{bmatrix} f_x X + c_x Z \\ f_y Y + c_y Z \\ Z \end{bmatrix} \\ &\approx \begin{bmatrix} f_x X/Z + c_x \\ f_y Y/Z + c_y \\ 1 \end{bmatrix}. \end{aligned} \tag{33} $$

Recall, that to use the 3D-2D transform the 3D point must be expressed in the camera coordinate frame. Therefore, we must perform a 3D-3D transform (see equation 28) from world coordinates to camera coordinates before deploying the 3D-2D transform in equation 31. We coalesce these operations into a single matrix:

$$ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} {}^{c}_{w}\mathbf{R} & {}^{c}\mathbf{t}_{worg} \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} {}^{c}_{w}\mathbf{R} & {}^{c}\mathbf{t}_{worg} \end{bmatrix} = \mathbf{M}_{ext} \tag{34} $$

In summary, performing 3D-2D transformation of world points (in 3D) denoted as ${}^{w}\tilde{\mathbf{x}}$ to camera points ${}^{c}\tilde{\mathbf{x}}$ we must (1) rotate from world to camera using ${}^{c}_{w}\mathbf{R}$, (2) translate from world to camera using $\mathbf{{}^{c}t_{worg}}$, (3) transform from 3D-2D, and finally (4) project based on intrinsic camera parameters using $\mathbf{K}$. The equation for a 3D-2D transformation becomes:

$$ {}^{c}\tilde{\mathbf{x}} = \mathbf{K}\mathbf{M}_{ext}\,{}^{w}\tilde{\mathbf{x}}. \tag{35} $$