

# Angular Domain Reconstruction of Dynamic 3D Fluid Surfaces

Jinwei Ye      Yu Ji      Feng Li      Jingyi Yu  
University of Delaware, Newark, DE 19716, USA  
[{jye,yuji,feli,yu}@cis.udel.edu](mailto:{jye,yuji,feli,yu}@cis.udel.edu)

## Abstract

We present a novel and simple computational imaging solution to robustly and accurately recover 3D dynamic fluid surfaces. Traditional specular surface reconstruction schemes place special patterns (checkerboard or color patterns) beneath the fluid surface to establish point-pixel correspondences. However, point-pixel correspondences alone are insufficient to recover surface normal or height and they rely on additional constraints to resolve the ambiguity. In this paper, we exploit using Bokode - a computational optical device that emulates a pinhole projector - for capturing ray-ray correspondences which can then be used to directly recover the surface normals. We further develop a robust feature matching algorithm based on the Active-Appearance Model to robustly establishing ray-ray correspondences. Our solution results in an angularly sampled normal field and we derive a new angular-domain surface integration scheme to recover the surface from the normal fields. Specifically, we reformulate the problem as an over-constrained linear system under spherical coordinate and solve it using Singular Value Decomposition. Experiments results on real and synthetic surfaces demonstrate that our approach is robust and accurate, and is easier to implement than state-of-the-art multi-camera based approaches.

## 1. Introduction

The problem of modeling and reconstructing time-varying specular surfaces such as dynamic 3D fluid wavefront has attracted much attention in recent years [16, 6, 23]. Successful solutions can benefit numerous applications in oceanology[10], fluid mechanism [2] and computer graphics [7] as well as lead to new insights towards shape reconstruction algorithms. The problem, however, is inherently difficult for a number of reasons. First specular surface does not have its own image. Instead, it borrows appearance from nearby diffuse objects. Second, determining the light path for shape reconstruction is non-trivial since refractions or reflections non-linearly alter the light paths. Finally, dynamic specular surfaces often exhibit spatially and tem-

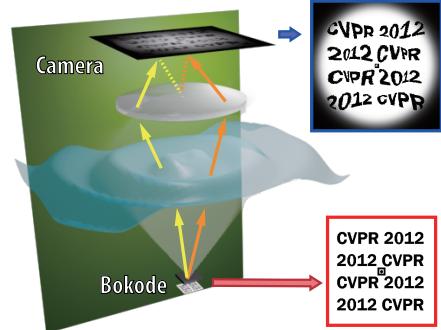


Figure 1. A Bokode projects a pattern toward the fluid surface and by matching the pattern with pixels, we obtain ray-ray correspondences for reconstructing angularly sampled surface normals.

porally varying distortions that are hard to correct.

Existing solutions on specular surface reconstruction can be essentially viewed as a special class of multi-view reconstruction algorithms. Often a known pattern such as a checkerboard is positioned near the surface and conceptually one can analyze the corresponding feature points in the observed cameras views and then apply stereo [16] or volumetric reconstruction [6] techniques for recovering the surface. In reality, point-pixel correspondences are *under-constrained* even for single reflection or refraction: to determine the surface normal, it is necessary to know both the incident and the exit ray directions; the pixel location provides the exit direction but the 3D point does not provide the incident direction, unless the surface height is known in prior. To resolve this ambiguity, additional constraints such as the planarity assumption [9, 17], surface smoothness prior [20] and surface integrability constraints [21] have been proposed.

In this paper, we present a novel and simple solution for resolving the point-pixel ambiguity. Our solution leverages recent advances in computational photography. Specifically, we place a special optical device called the Bokode [15] beneath the surface where the Bokode behaves as a pinhole projector that emits rays from a common 3D point. We then mount a high resolution camera on top of the fluid surface to observe the distorted projection pattern. By associating the projected and the observed patterns, we instantly obtain ray-

ray correspondences that resolve the previous point-pixel ambiguity as shown in Fig.1.

In theory, the ray-ray intersections should directly correspond to the wavefront surface whereas the two ray directions would provide the surface normal. In reality, the surface obtained from ray-ray intersections can be highly noisy due to calibration and numerical errors. We therefore only use the ray directions to recover the surface normal. The resulting normal field, however, is sampled angularly. We therefore derive a new angular-domain surface integration scheme. We first show that traditional spatial-domain surface integration problem, i.e., Poisson surface completion, can be reformulated as to solve an over-determined linear system. Likewise, we formulate the angular-domain surface reconstruction as a similar linear system under spherical coordinate and solve it using Singular Value Decomposition. We experiment our new fluid surface reconstruction approach on both synthetic and real data. Experimental results show that our technique is robust and accurate and is easier to implement than state-of-the-art multi-camera based solutions.

## 2. Related Work

Existing specular (reflective and refractive) surface reconstruction algorithms have generally followed the correspondence-based approaches and we classify them into two categories based on different types of correspondences.

**Point-Pixel Correspondences.** Most existing solutions for specular surface reconstruction build upon point-pixel correspondences where a special planar pattern such as a checkerboard is placed near the surface and a single or multiple cameras are used to acquire the distorted pattern for shape reconstruction. Murase [17] analyzed the optical flow between the distortion image and the original one and used the center of trajectory to establish point-point correspondences. Blake [3] examined the variation of reflected highlight by changing the viewing position to recover the surface geometry and reflective properties Bonfort and Sturm [4] used images captured by multiple calibrated cameras to reconstruct specular surface with voxels. Recently Sankaranarayanan *et al.* [19] used standard SIFT algorithm to match point-pixel correspondences resulting from specular flow and further used quadrics approximation to recover mirror-type surfaces from sparse samples. A common issue in point-pixel based solutions is ambiguity: a pixel corresponds to a ray from the camera while the specular surface can lie at any position along the ray. Tremendous efforts have been focused on adding additional constraints [9, 17, 20, 21] for resolving this ambiguity.

The problem of acquiring dynamic specular surfaces is relatively new to computer vision. Morris and Kutulakos [16] tracked the corners of the checkerboard pattern over

time to establish point-pixel correspondences and then imposed the refractive disparity constraint to iteratively solve for surface height and surface normal. However, robustly tracking feature points on dynamic surfaces is challenging as the observed image can exhibit severe distortions and motion blurs. In a similar fashion, Ding *et al.* [6] recently constructed a camera array system to obtain multi-view point-pixel correspondences. When one of the cameras loses track, the rest of the cameras can still recover the surface and the result can be used to warp the lost-track feature points back to the camera.

**Ray-Ray Correspondences.** A different class of solutions that can directly resolve the point-pixel ambiguity is to use ray-ray correspondences. The earlier work of Sanderson *et al.* [18] controlled the illumination direction and coupled it with the observed specular highlights to form ray-ray correspondences. Kutulakos and Steger [12] recovered complex-shaped static specular objects by computing the light paths from the specular object to the camera. By studying indirect projection of 3D points, they formulated the problem of recovering the light path as a general triangulation problem. However their framework by far can only handle static objects as it requires acquiring the object twice whereas we present a simpler solution to directly handle dynamic 3D fluid surfaces.

Closely to our solution, Wetzstein *et al.* [23] recently to replace the conventional checkerboard pattern with a light field probe which encodes 4D spatial and angular information. In their setup, they used color gradients to code the 2D incident ray direction and 1D (vertical) feature point position. The second (horizontal) dimension of the feature point can be recovered through geometric constraints. Their approach can achieve highly accurate ray-ray correspondences. Our solution differs from theirs in a number of ways. First, we use a much simpler and affordable device, a Bokode that can be easily constructed from a webcam, in place of the light field probes. Our ray-ray correspondences, however, are less accurate than the ones obtained by the light field probe and they cannot be used to directly recover the surface. We therefore only use the recovered normal field and develop an angular-domain normal field integration scheme. Finally, dynamic fluid surfaces often cause strong chromatic aberrations and intensity changes due to caustics. Therefore, we choose not to use the color-coded pattern but a special monochromatic pattern and apply Active Appearance Model (AAM) for correspondence matching.

## 3. Bokode-based Acquisition System

Fig.2 shows the algorithm flow our proposed Bokode-based fluid surface reconstruction framework. We use Bokode projecting out pattern towards fluid surface and capture the distorted pattern. By associating the distorted

pattern with the projected one using AAM matching, we then obtain the incident-exit ray correspondences for computing surface normals. Since the normals are sampled in angular-domain, we therefore reconstruct the surface with our new spherical coordinate based surface integration algorithm.



Figure 2. A block diagram that shows the pipeline of our Bokode based fluid surface reconstruction framework.

Bokode is an optical device that resembles a pinhole projector [15]. In essence, a Bokode emits lights originating from the common 3D point over different angles as shown in Fig.3(a). When capturing a Bokode using a camera with a small aperture, the Bokode would appear as a single dot as the camera only captures a specific angle of rays. In contrast, if a Bokode is captured by a camera with a large aperture, the pattern emitted by the Bokode can be partially captured as shown in Fig.3(b). Using this unique feature, Mohan *et al.* [15] proposed to use Bokode as an invisible identification tag. In the similar vein, we explore using Bokode as a special active illumination device for fluid surface reconstruction.

To physically implement a Bokode, the simplest approach is to construct a pinhole type of device that only allows the lights to pass through the hole. However, similar to pinhole cameras constructed as such, this design suffers from blurry images and insufficient lights. In reality, a Bokode can be approximated using a small lens projector with the projection pattern positioned at the focal length. The viewing camera is then positioned relatively faraway from the Bokode and focuses at infinity to effectively sample the angular rays emitted from the Bokode as shown in Fig.3(a).

It is important to note that a commodity projector cannot be directly used as a Bokode. Although both Bokode and commodity projectors use back light to illuminate the projection pattern, the Bokode requires the pattern be placed at the depth of the focal length whereas the projector places the pattern much farther away from the lens for magnifying the pattern. Further, the Bokode uses a much smaller aperture to effectively emulate a pinhole system while the aperture of a commodity projector is set ultra large to ensure the brightness of projection. In our setup, we construct a lens-based Bokode that contains four layers: lens, pattern, diffuser and light source. We use the lens of a commodity web camera as the Bokode lens. The lens has a diameter of 2mm and a focal length of 8mm. We print a special monochrome pattern on a transparency at a resolution of 5080 dpi. We further use the diffuser to ensure that lights emit towards all directions. Finally, to increase the brightness, we use an ultra-bright LED flash light of 400 lumen as the back light to the Bokode.

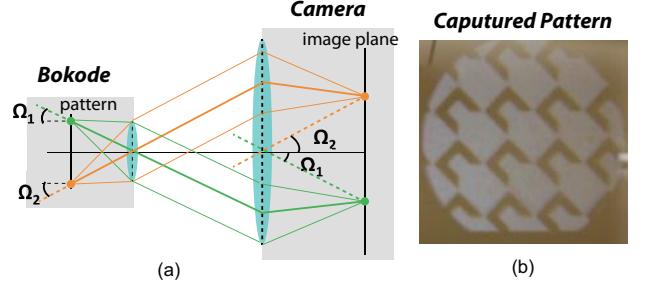


Figure 3. How Bokode is viewed by a camera. (a) Parallel lights emitting from a point on the Bokode pattern converge to the same pixel on image plane of a camera focusing at infinity; (b) A typical Bokode image captured by a camera with wide aperture focusing at infinity.

Fig.4 shows our Bokode-based fluid surface acquisition system. We place the Bokode underneath a water tank to project lights towards the fluid surface. Since the bottom of water tank is flat and thin, we ignore the refraction effects and view the Bokode as if it sits directly at the bottom of the tank. We assume each feature on the projection pattern corresponds to a thin beam of parallel light rays and the top (wavefront) fluid surface interacting with each light beam is nearly flat. Therefore, their corresponding exit rays remain approximately parallel.

Let  $P_i(x_i, y_i)$  be a point on the projection pattern on the Bokode, where  $(x_i, y_i)$  are the relative coordinates of  $P_i$  to the lens' optical center, and  $f_b$  be the focal length of the Bokode lens.  $P_i$  then maps to a light beam with direction  $\alpha_i = \arctan \sqrt{x_i^2 + y_i^2} / f_b$ . We call the ray direction from the Bokode towards the fluid surface the incident ray direction.

On the camera side, we use a calibrated camera focusing at infinity to capture the light rays after being refracted by the fluid surface. For each pixel  $P'_i(x'_i, y'_i)$  on the captured image, we can use the camera parameters to obtain its corresponding ray direction as  $\beta_i = \arctan \sqrt{x'_i^2 + y'_i^2} / f_c$  where  $f_c$  is the focal length of the camera lens. We call  $\beta_i$  the exit ray direction. Notice that once we obtain  $P_i$  and  $P'_i$  correspondences, we can simply intersect the rays to obtain the fluid surface position and normal, although in reality only the directions of the rays are useful. In this paper, we call  $P_i$  and  $P'_i$  correspondences the Incident-Exit-Ray (IER) correspondences.

To calibrate our system, we first calibrate the camera and align its optical axis with the Bokode's main axis. We then capture a Bokode image without adding any fluid to the tank. In this case, the exit ray directions captured by the camera are identical to the incident ray directions. In our implementation, the Bokode projects a special pattern with many feature points and the calibration process associates the feature points with the incident ray directions from the Bokode. Once the tank is filled with fluid, we then find the matching feature points (see Sec.4) which would directly

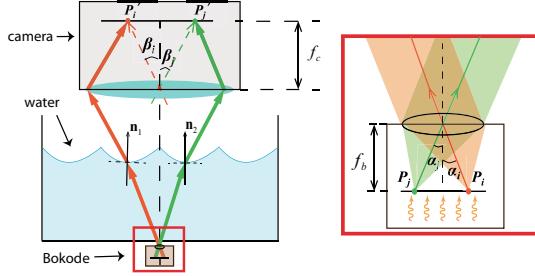


Figure 4. Each point on the Bokode pattern maps to a beam of parallel rays. These rays are refracted by fluid surface and gathered by the viewing camera at a pixel.

provide IER correspondences.

#### 4. Correspondence Matching

The choice of the projection pattern is important for reliable correspondence matching and hence surface reconstruction. Most previous approaches use a checkerboard pattern and track the feature points (corners). For our Bokode-based solution, checkerboard pattern is less suitable. This is because existing two-view or multi-view based approaches assume that the cameras can view the complete checkerboard, which greatly helps tracking the pattern over time and across views. In our case, the Bokode has a much wider field of view with  $130^\circ$  compared to the viewing camera with merely  $20^\circ$ . Therefore, the camera can only view part of the pattern and since checkerboard patterns are highly symmetric, tracking the features consistently is challenging. Another option is to use the color-based pattern. For example, Wetzstein *et al.* [23] red, blue, and green gradient to encode the 2D directions and 1D vertical positions. The color-based pattern is suitable for static surface but can cast challenge to dynamic fluid surfaces due to chromatic aberrations and caustics. Chromatic abberations destroys color calibration results and caustics changes the intensity, making it difficult to match colors. Further, the color patterns are usually of a much lower resolution. The common resolution of color printer is 1440 dpi while our Bokode pattern is printed at 5080 dpi.

We choose to use an irregular monochrome patterns as shown in Fig.3(b) and apply Active Appearance Model for correspondence matching. Our pattern consists of an array of tiled asymmetric symbols. Each symbol has the size of  $80\mu m$  and the entire pattern is of dimension  $1cm \times 1cm$ . The symbol has sharp corners and edges to provide effective feature points. Specifically, we use the corners as the major feature points and interpolate along the edges between every pair of major features to generate secondary feature points. Further, we place a checkerboard square marker at the center of the pattern calibrating its position and add a dot to the top-corner of the square to identify its orientation.

To track the feature points, we apply the Active Appearance Model (AAM) [5, 14] that was originally devel-

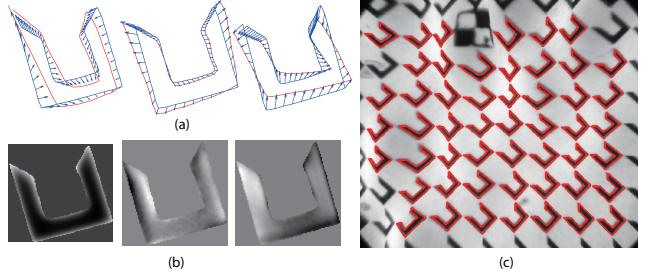


Figure 5. Active appearance model for pattern-pixel matching. (a) Shape model: the shape in blue is the mean shape  $s_0$  and the ones in red are shape variations estimated by PCA; (b) Appearance model: the left shows the mean appearance  $A_0$  and the first two eigen appearance bases; (c) An example of AAM matched pattern.

oped for pattern recognition. An AAM is composed of two components: a shape model and an appearance model. The shape model is described using as a set of  $N$  feature points  $(x_1, y_1, x_2, y_2, \dots, x_N, y_N)$  and is represented as a mean shape  $s_0$  with a linear combination  $p_i$  of variations on  $n$  shape basis  $\{s_i\}$ :

$$s(p) = s_0 + \sum_{i=1}^n p_i s_i \quad (1)$$

The appearance model is defined as the intensity of image patches surrounding the mean shape. Similar to the shape model, we model the appearance model with a mean appearance  $A_0$  plus a linear combination  $\lambda_i$  of variations on  $m$  appearance basis  $\{A_i\}$ :

$$A(p) = A_0 + \sum_{i=1}^m \lambda_i A_i \quad (2)$$

Same as classical AAM-based recognition techniques [5, 14], we obtain the mean shape  $s_0$ , the mean appearance  $A_0$ , the shape variations  $\{p_i\}$  and the appearance variations  $\{\lambda_i\}$  by applying the Principal Component Analysis (PCA) to our training data. Fig.5(a) shows the mean shape and some shape variations; Fig.5(b) shows the mean appearance and two appearance bases. To match the shape to a specific image, the AAM technique finds the optimal shape parameters and appearance parameters that minimize the appearance variations:

$$E_a(\mathbf{x}) = \sum_{\mathbf{x} \in s_0} [A_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i A_i(\mathbf{x}) - I(W(\mathbf{x}; \mathbf{p}))]^2 \quad (3)$$

where  $W(\mathbf{x}; \mathbf{p})$  is the affine warping defined by a shape model  $s(p)$  and the mean shape  $s_0$  that maps every pixel from the model coordinate to the corresponding image coordinate.

We generate our training data by rendering a large set of distorted patterns using varying fluid surface normals, height, and orientations. To match a new distorted pattern, we first segment the captured image to small patches, each containing a single symbol. We then perform AAM search on each symbol to robustly handle non-uniform intensity

caused by the caustics. In the first frame of the video sequence, we initialize the match by aligning the model and the capture symbols at their centroid. For the consecutive frames, the matched shape from the previous frame is used to initialize the matching process. Since our symbols are have high contrast to the background, we further generate a distance map  $M_d$  based on contour of the shape as additional cost constraint to guide the AAM search:

$$E_d(\mathbf{x}) = \sum_{\mathbf{x} \in s_0} (M_d(W(\mathbf{x}; \mathbf{p})))^2 \quad (4)$$

Therefore, the total cost function for shape matching hence becomes  $E = E_a + wE_d$ , where  $w$  is the weighting factor to the distance map constraint. Fig.5(c) shows an AAM matched result of captured image.

Once we match the captured symbol with the projected one, we instantly obtain IER correspondences. Recall that the incident ray directions  $\mathbf{d}_{in}$  are encoded in the projected Bokode image and are precomputed in the calibration step and the exit ray directions  $\mathbf{d}_{exit}$  can be calculated with the viewing camera parameters. We assume that the refraction indices of air and the fluid  $n_1$  and  $n_2$  are known in prior respectively and we can solve for the surface normal using the Snell's law as:

$$\mathbf{n} = n_2 \mathbf{d}_{in} - n_1 \mathbf{d}_{exit} \quad (5)$$

Since we only sample a relatively sparse set of feature points on the pattern, we obtain a sparsely and irregularly sampled normal field. We then apply the Radial-Basis Function (RBF) function to interpolate the normal field.

## 5. Surface Reconstruction from Angularly Sampled Normals

Next, we show how to reconstruct the fluid surface from the angularly sampled normal field recovered from Bokode discussed in Sec.4. Recall that the key advantage of our approach is that it directly recovers the normal direction from the IER correspondences. The resulting normal field, however, is very different from the classical height-field based one. To elaborate, if we model the surface as a height field  $z(x, y)$ , the scaled normal vector at each point  $(x, y)$  is simply  $(z_x, z_y, -1)$ ; when given the boundary condition (Neumann or Dirichlet) and the height-field based normal field, the problem of integrating the normal field to recover the same can be formulated to find the optimal surface  $f$  where:

$$\min_f \iint ((f_x - z_x)^2 + (f_y - z_y)^2) dx dy \quad (6)$$

Previous approaches [11, 1] have shown that solving this optimization problem is equivalent Poisson equation:  $\Delta f = z_{xx} + z_{yy}$ , where  $\Delta$  is the Laplacian operator:  $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2$ . In the discrete case, one can linearize the Laplacian and the derivative operator and form an linear system in  $f$  and directly solve for  $f$  as shown in the supplementary material.

Our angular normal sampling scheme, in contrast, measures normals at discrete inclination and azimuth angles. These angular domain normals cannot be directly mapped to spatial domain normals, i.e., we cannot perform ray-surface intersection as the surface is unknown. Further, by using a wide aperture viewing camera, we can only recover the exit ray direction rather than the ray itself, i.e., we cannot perform ray-ray intersections [22]. We therefore derive a new angular-domain surface integration scheme. We parameterize the surface in spherical coordinates as  $r(\theta, \phi)$  where the origin coincides with the Bokode's pinhole,  $r$  is the radial distance from the origin, and  $\theta$  and  $\phi$  correspond to the azimuthal and polar angles respectively. Our sampling essential recovers the normals at discrete samples of  $\theta$  and  $\phi$  and our goal is to recover the radius  $r$  from the sampled normal field.

At each surface point  $r(\theta, \phi)$ , we can compute its gradients under spherical coordinate as  $(\partial r/\partial\theta, \partial r/\partial\phi)$ . In reality, we only have sampled normal directions measured in the Cartesian coordinate. We therefore need to further convert the gradients in Cartesian coordinate to spherical coordinate.

Recall that  $x = r \sin \phi \cos \theta$ ,  $y = r \sin \phi \sin \theta$ , and  $z = r \cos \phi$ , we have:

$$\begin{cases} \frac{\partial r}{\partial\theta} = r \cdot \frac{\sin\phi(\frac{\partial z}{\partial x}\sin\theta - \frac{\partial z}{\partial y}\cos\theta)}{\sin\phi(\frac{\partial z}{\partial x}\cos\theta + \frac{\partial z}{\partial y}\sin\theta) - \cos\phi}, \\ \frac{\partial r}{\partial\phi} = r \cdot \frac{\sin\phi + \cos\phi(\frac{\partial z}{\partial x}\cos\theta + \frac{\partial z}{\partial y}\sin\theta)}{\cos\phi - \sin\phi(\frac{\partial z}{\partial x}\cos\theta + \frac{\partial z}{\partial y}\sin\theta)}. \end{cases} \quad (7)$$

Eq.(7) illustrates the constraint between radius gradient to surface normal. Our goal is to find the optimal surface  $r$  (in spherical coordinate) under the constraints.

We first discretize the surface  $r$  in discrete  $(\theta_i, \phi_j)$ . We can then approximate Eq.(7) using finite difference as:

$$\begin{cases} r_{i+1,j} - r_{ij} = r_{ij} \cdot P_{ij} \\ r_{i,j+1} - r_{ij} = r_{ij} \cdot Q_{ij} \end{cases} \quad (8)$$

where

$$\begin{cases} P_{ij} = \Delta\theta \cdot \frac{\sin\phi_j(\frac{\partial z}{\partial x}(i,j)\sin\theta_i - \frac{\partial z}{\partial y}(i,j)\cos\theta_i)}{\sin\phi_j(\frac{\partial z}{\partial x}(i,j)\cos\theta_i + \frac{\partial z}{\partial y}(i,j)\sin\theta_i) - \cos\phi_j} \\ Q_{ij} = \Delta\phi \cdot \frac{\sin\phi_j + \cos\phi_j(\frac{\partial z}{\partial x}(i,j)\cos\theta_i + \frac{\partial z}{\partial y}(i,j)\sin\theta_i)}{\cos\phi_j - \sin\phi_j(\frac{\partial z}{\partial x}(i,j)\cos\theta_i + \frac{\partial z}{\partial y}(i,j)\sin\theta_i)} \end{cases}$$

In  $\{P_{ij}\}$  and  $\{Q_{ij}\}$ ,  $\frac{\partial z}{\partial x}(i,j)$  and  $\frac{\partial z}{\partial y}(i,j)$  are observed normals in Cartesian coordinate corresponding to  $(\theta_i, \phi_j)$ . Assume we have discretized  $\theta$  and  $\phi$  into a  $m \times n$  grid, we then form an over-constrained linear system from Eq.(8): we

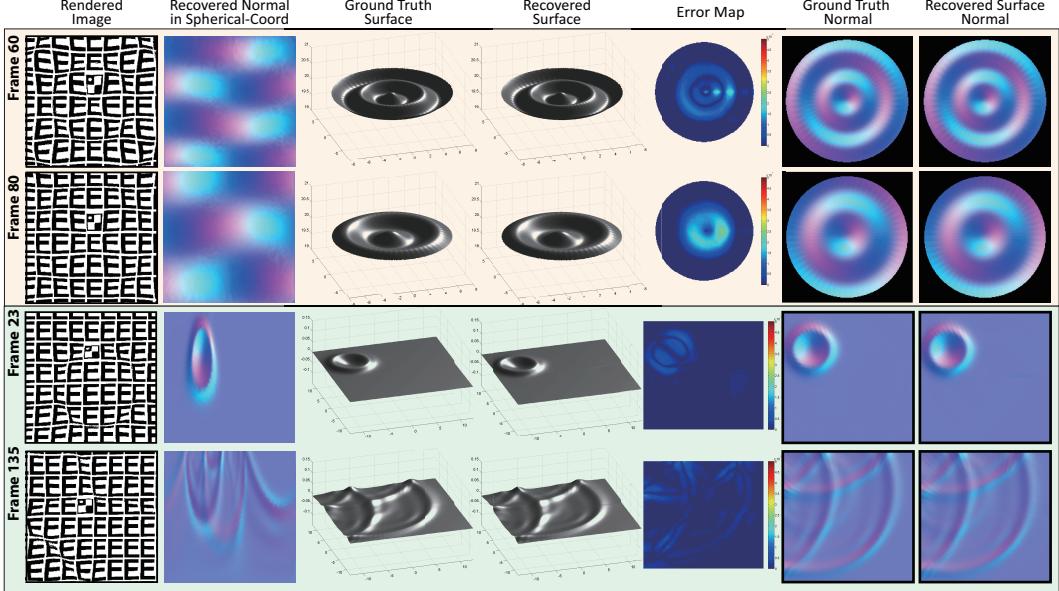


Figure 6. Results on a synthetic sinusoid wave (top rows) and on a Helmholtz wave (bottom two rows). We show the cropped Bokode pattern used in reconstruction. In particular, column 2 are sampled surface normals under spherical coordinate in respect to  $\theta$  and  $\phi$  and we further compute normal maps in Cartesian coordinate and compare with the ground truth ones to demonstrate the accuracy of our method as shown in column 6 and 7. The complete sequences can be found in the supplementary video.

have two equations for each radius  $r_{ij}$  except for the boundary. For the  $\theta$  boundaries, we have  $r(0, \phi_j) = r(2\pi, \phi_j)$ , i.e.,  $r_{0j} = r_{mj}$ . However, for the  $\phi$  boundaries, it is not easy to acquire radius at those points. We solve this problem by using the current frame’s reconstruction result to predict the boundary of later frames. In all we have  $mn$  unknowns and  $2mn$  linear equations. By stacking these together, we obtain a linear system with equations  $A\Theta = 0$ , where  $A$  is the coefficient matrix formed by  $\{P_{ij}, Q_{ij}, i = 1, \dots, m; j = 1, \dots, n\}$  and  $\Theta = \{r_{ij}\}, i = 1, \dots, m; j = 1, \dots, n$ . Then we apply Singular Value Decomposition (SVD) on our linear system to obtain the least square solution of surface radiiuses.. In the supplementary material, we prove that this is a valid approach as traditional spatial-domain surface completion can also be formulated and solved using a similar over-constrained linear system.

## 6. Experiments

We have validated our approach on both synthetic and real fluid surfaces. For synthetic surfaces, we have implemented a Ray-tracer that back-traces feature points from the Bokode pattern to pixels in the viewing camera. For real surfaces, we capture video streams of dynamic fluid surfaces using our acquisition system (see Sec.3).

### 6.1. Synthetic Scene Simulation

We first conduct experiments on a synthetic sinusoidal wave:  $z(x, y, t) = 20 + \cos(\pi t \sqrt{(x - w/2)^2 + (y - h/2)^2}/200)$ , where  $w = h = 300$ . On the Bokode side, we use a pattern of physical size

$300 \times 300$  with 48 symbols on it. In our setup, the viewing camera captures 32 symbols. We also use the Helmholtz Equation to propagate the of the same wavefront at  $t = 0$ , to synthesize more realistic fluid effects and test the robustness of our algorithm. In the Helmholtz wave case, we use a higher resolution pattern of  $600 \times 600$  with 220 symbols to improve the angular resolution. The refraction index of the fluid is set to be 1.33 to emulate water.

To generate the AAM training data, we render 100 distorted pattern image on randomly sampled normals. Since we use ray-tracing, we obtain ground-truth feature points at both the corners and along edges. In our experiment, we use 8 corner points and 10 edge points in between each pair of neighboring corners on the symbol. We apply AAM matching to the synthesize fluid images using the training data and obtain the ray-ray correspondences. We then compute the surface normal at each feature point and interpolate a dense normal field. For the sinusoidal wave, the angularly resolution of our sampled normal map is  $0.33^\circ$  and for the Helmholtz wave, we generate a higher angular resolution of  $0.2^\circ$  to recover fine details. Finally, we apply our spherical coordinate surface integration scheme. In both cases, we use the ground truth wave boundary for integration.

Fig.6 shows our recovered wavefronts at different time instances. The video sequence of the results can be found in supplementary material. We further compute the reconstruction error to illustrate the accuracy of our method. The amplitude of sinusoidal wave is in range of [19.9, 20.1] and our average reconstruction error is  $9.782 \times 10^{-4}$ . Helmholtz wave is in a similar range and our reconstruction error is

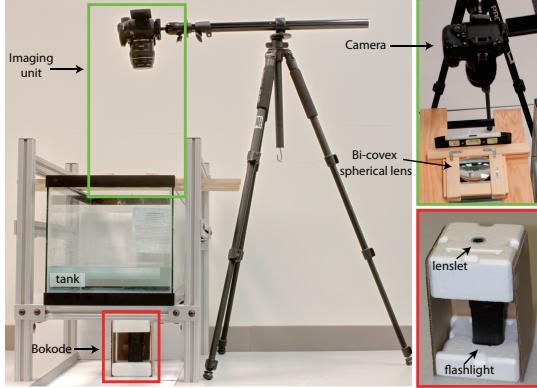


Figure 7. Our experimental setup. We construct a Bokode using a flashlight, a diffuser, a high-resolution pattern and webcam lens. We also use an auxiliary bi-convex spherical lens to collect lights refracted by the fluid surface (top right).

$7.773 \times 10^{-4}$ . This implies that using a denser pattern would improve accuracy. In reality, however, producing a dense pattern for the Bokode is challenging due to the resolution limit on commodity printer.

## 6.2. Real Scene Experiments

To capture real fluid surfaces, we set up our system for capturing real fluid surface as shown in Fig.7. We construct a Bokode that consists of a bright flash light of 400 lumen, a micro-pattern and a lenslet with 2mm aperture and 8mm focal length dissembled from a cheap conventional web camera. We print a monochrome Bokode pattern at a resolution of 5080 dpi on a  $1cm \times 1cm$  transparency using the professional printing service provided by Page-Works (<http://www.pageworks.com>). On the viewing side, we couple a high resolution DSLR camera (Canon 60D, lens 85f1.8) with an auxiliary bi-convex spherical lens of 100mm with focal length 170mm to capture a wider angular range of rays. We adjust the viewing camera to focus at the focal plane of the auxiliary lens to record an HD video at a resolution of  $960 \times 720$  at 30 fps. Our water tank is of size  $24cm \times 18cm \times 36cm$  and the viewing camera under our lens and aperture setting can observe an area of around  $600mm \times 600mm$ . We pre-calibrate the viewing camera using Zhang’s algorithm[24] and then capture the image of the Bokode pattern with water for obtaining the incident ray direction with respect to each feature point on the pattern.

We experiment our method on two types of wavefront. The first one is created by randomly perturbing the fluid at one end of the water tank to propagate the wave towards the other end; The second is a “ring-type” wave that is created by blowing air into towards the fluid. One major challenge that we observe in the real fluid surface case but not in the synthetic one is the effects of caustics which changes the intensity of the observed patterns. To reuse the training data, we use only the binary gradient map of the rendered images as the appearance model and then apply

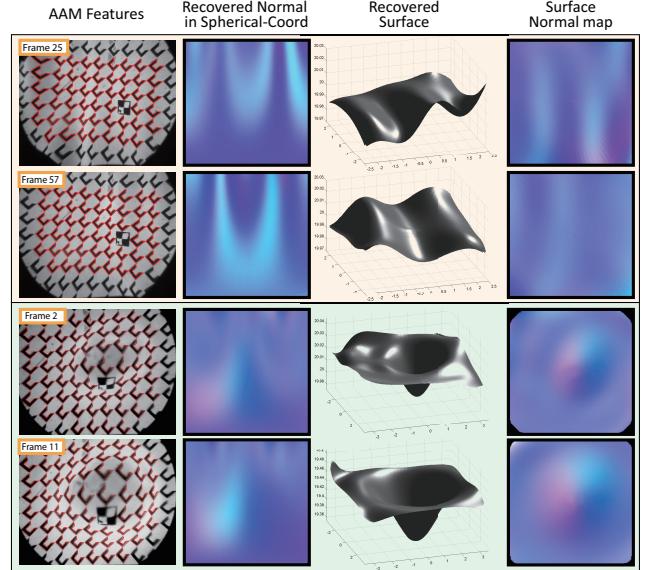


Figure 8. Results on two sets of real data (a perturbed wave and a ring wave). From left to right: we show the captured image with matched features, the sampled normal under spherical coordinate, the reconstructed surface, and the surface normal field computed from the reconstructed surface.

**AAM matching.** In some cases, the acquired image can exhibit motion blurs and we need to apply manual alignments. To integrate the surface, we assume that the fluid boundary is flat in the first frame and then apply the Navier Stokes (NS) model to propagate the boundary [13]. Fig.8 shows our acquired raw data, the AAM tracked results, and our reconstructions a number of frames of real fluid surfaces. In the “ring-type” wavefronts, several acquired patches exhibit strong distortions. Our technique is able to reasonably align the distorted pattern using AAM and our reconstruction results are consistent with the observed distortions. In fact, the quality of our reconstruction can be further improved by using more training samples in AAM. We refer the reviewers to the supplementary videos for the completely reconstructed sequences.

## 7. Discussions and Conclusions

We have presented a novel and affordable solution for reconstructing dynamic fluid surfaces by using a special optical device called the Bokode to emulate a pinhole projector. By associating the projection pattern with the observed pixels, we directly obtain ray-ray correspondences that can be used to recover the surface normal field. Our method hence is one of the few that directly resolves the point-pixel ambiguity in single-view based solution. Another unique feature of our approach is that it provides an angular reconstruction of the normal field whereas most, if not all, previous approaches recover a spatial (height-field) sampling. We have hence developed a tailored surface integration algorithm for

integrating the normal field.

Our technique has a number of limitations. First, we rely the AAM technique for feature alignment. We chose not to use color patterns as chromatic aberrations caused by refraction can greatly affect color registration. The quality of AAM, however, depends heavily on the training data. In our first few trials on acquiring real surfaces, we were unable to match many symbols due to distortions and we had to render a much larger set of training data and occasionally need to conduct manual alignment. In the future, we plan to explore more robust feature matching algorithms. For example, one possible solution is to use temporally coded patterns[8], which would provide a reliable and much denser set of feature correspondences. The challenge there, however, would be the frame rate as we aim to acquire dynamic surfaces.

Another important future direction we plan to explore is on surface reconstruction from angularly sampled normals. In our implementation, we only approximate the boundary condition for integrating the surface. In the future we will investigate how to acquire the ground truth boundary, e.g., by using auxiliary cameras or other types of sensors. Further, it is important to note that our surface integration scheme only provides an approximation. Previous spatial-domain surface completion scheme finds the global optimal surface that best matches the normal field (in  $L_2$  normal). Ours finds the local optimal by discretizing the constraints into piecewise linear ones, although our results show that this approximation is highly effective and accurate. In the future, we plan to conduct a more comprehensive study on angular-domain surface completion by using the Variational method in a similar fashion to Poisson surface/image completion.

Our current setup only allows us to capture a small area of the fluid surface. The Bokode itself has a very wide field-of-view of up to  $160^\circ$  (depending on the pattern size and the lens' focal length) and can a single Bokode can cover a large surface. The limitation is on the viewing camera side whose aperture is usually much smaller. Our current solution is to an auxiliary lens to refocus the rays toward the camera. The range of acquisition hence is restricted by the size of the auxiliary lens. A simple solution is to use an ultra-large lens but at a much higher cost. A more practical solution is to construct an auxiliary lens array to emulate the large lens. Finally, compared with the recent light field probe based solution[23] which inspired our work, our ray-ray correspondences are less accurate and cannot be directly used for recovering the surface height. In the future, we plan to work with the authors to compare the reconstruction results on a number of benchmark surfaces and explore possible integrations of the two systems.

## Acknowledgement

This project was partially supported by the National Science Foundation under grants IIS-CAREER-0845268 and IIS-RI-1016395, and by the Air Force Office of Science Research under the YIP Award.

## References

- [1] A. K. Agrawal, R. Raskar, and R. Chellappa. What is the range of surface reconstructions from a gradient field? In *ECCV*, 2006.
- [2] W. J. D. Bateman, C. Swan, and P. H. Taylor. On the efficient numerical simulation of directionally spread surface water waves. *J. Comput. Phys.*, 174:277–305, November 2001.
- [3] A. Blake. Specular stereo. In *Proceedings of the 9th international joint conference on Artificial intelligence - Volume 2*, 1985.
- [4] T. Bonfort and P. Sturm. Voxel carving for specular surfaces. In *ICCV*, pages 591 –596 vol.1, Oct. 2003.
- [5] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. *IEEE TPAMI*, 23(6):681 –685, June 2001.
- [6] Y. Ding, F. Li, Y. Ji, and J. Yu. Dynamic Fluid Surface Acquisition Using a Camera Array. In *ICCV*, 2011.
- [7] D. Enright, S. Marschner, and R. Fedkiw. Animation and rendering of complex water surfaces. In *SIGGRAPH*, 2002.
- [8] J. Gu, T. Kobayashi, M. Gupta, and S. K. Nayar. Multiplexed Illumination for Scene Recovery in the Presence of Global Illumination. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1–8, Nov 2011.
- [9] K. Ikeuchi. Determining surface orientations of specular surfaces by using the photometric stereo method. *IEEE TPAMI*, PAMI-3(6):661 –669, Nov. 1981.
- [10] B. Jähne, J. Klinken, and S. Waas. Imaging of short ocean wind waves: a critical theoretical review. *J. Opt. Soc. Am. A*, 11(8):2197–2209, Aug. 1994.
- [11] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, pages 61–70, 2006.
- [12] K. Kutulakos and E. Steger. A theory of refractive and specular 3d shape by light-path triangulation. In *ICCV*, 2005.
- [13] F. Li, L. Xu, P. Guyenne, and J. Yu. Recovering fluid-type motions using navier-stokes potential flow. In *CVPR*, 2010.
- [14] I. Matthews and S. Baker. Active appearance models revisited. *Int. J. Comput. Vision*, 60:135–164, November 2004.
- [15] A. Mohan, G. Woo, S. Hiura, Q. Smithwick, and R. Raskar. Bokode: imperceptible visual tags for camera based interaction from a distance. In *SIGGRAPH*, 2009.
- [16] N. Morris and K. Kutulakos. Dynamic refraction stereo. In *ICCV*, 2005.
- [17] H. Murase. Surface shape reconstruction of an undulating transparent object. In *ICCV*, 1990.
- [18] A. Sanderson, L. Weiss, and S. Nayar. Structured highlight inspection of specular surfaces. *IEEE TPAMI*, 10(1):44 –55, Jan. 1988.
- [19] A. Sankaranarayanan, A. Veeraraghavan, O. Tuzel, and A. Agrawal. Specular surface reconstruction from sparse reflection correspondences. In *CVPR*, 2010.
- [20] S. Savarese and P. Perona. Local analysis for 3d reconstruction of specular surfaces. In *CVPR*, 2001.
- [21] M. Tarini, H. P. A. Lensch, M. Goesele, and H.-P. Seidel. 3d acquisition of mirroring objects using striped patterns. *Graph. Models*, 67:233–259, July 2005.
- [22] G. Wetzstein, R. Raskar, and W. Heidrich. Hand-held schlieren photography with light field probes. In *ICCP*, 2011.
- [23] G. Wetzstein, D. Roodnick, R. Raskar, and W. Heidrich. Refractive Shape from Light Field Distortion. In *ICCV*, 2011.
- [24] Z. Zhang. A flexible new technique for camera calibration. *IEEE TPAMI*, 22(11):1330 – 1334, Nov. 2000.