https://doi.org/10.2352/EI.2022.34.14.COIMG-306

Inferring surface properties of oscillating fluids from video by inversion of physics models

Bob Price[1] ***(bprice@parc.com) Svyatoslav Korneev***[1] ***Adrian Lew***[1] ***Christoforos Somarakis***[1] ***Raja Bala***[1] ***Jonathan (Shengtai) Ju***[1]
1) Palo Alto Research Center - A Xerox Company, Palo Alto, CA, 94304, USA

Abstract

Measuring the shape, motion, and physical properties of oscillating fluids is critical for understanding the physics of fluid systems and optimizing and controlling them in real-time. Conventional surface measurement techniques such as profile analysis or stereo reconstruction are not effective for monitoring fluids in industrial processes due to occluding structures, extreme heat, and complex light interactions at the fluid surface. We propose a video-based method comprising forward and inverse transforms. The forward transform employs a physics-based fluid surface model combined with a ray-traced renderer to map shape and motion parameters to synthetic video frames. The inverse transform uses machine learning models to recover surface parameters from video. The inverse models are trained on synthetic data generated by the forward transform. We illustrate the method on an industrial 3D printer for which we recover the motion and surface of a molten aluminum alloy oscillating inside a microscopic nozzle. The inverse transform is ill-posed but can be regularized. We show that surface properties can be reliably inferred with either a suitably regularized nearest neighbor regressor or a deep convolutional network whose results are less stable but faster to compute.

Introduction

Understanding the shape, motion, and physical properties of oscillating fluids can provide insight into the physics of fluid systems, enable the optimization of designs in industrial fluid processes and enable real-time optimization of industrial mechanisms operating on such fluids. However, fluids in industrial processes present many challenges to conventional measurement techniques, such as occluding structures, extremes of heat and pressure, and complex light interactions at the fluid surface. We develop and illustrate an approach to this important class of problems for a specific application in monitoring and controlling the liquid metal jet on the Xerox ElemX 3D metal alloy printer (Figure 1). A critical problem in maximizing print quality is controlling the settling time of molten metallic fluid in the nozzle between ejection events. Material properties, nozzle design, and pump control can influence the settling time. The optimization of some of these parameters will benefit from ways to quantify the oscillation of the fluid. Here we use visual methods to monitor the oscillation by inferring the shape of the metal-air interface from high-speed video frames. The highly specular metallic surface of the molten metal and the obstructive nature of the nozzle precludes conventional depth-inference techniques. We develop two physics-informed techniques that overcome these challenges.

Figure 1. *Xerox ElemX Printer 3D Aluminum Alloy Printer*

Related Work

The recovery of shape from images has a long history in computer vision. In the “shape from shading” [3] approach, one uses the changes in reflectance with angle found on Lambertian surfaces to infer the surface normal at each pixel in the image. One can then integrate over the slopes to get a shape. Unfortunately, these techniques do not work well on specular surfaces. The specular highlights are very sparse so that there is not much information about the surface, and when there are gradients, the transitions tend to be fairly abrupt between minimum and maximum brightness levels.

Another common method is the “silhouette” method which builds up a shape by looking at the shadow cast by an object when viewed from the side. The object can be rotated to build up a three-dimensional shape. It only works on convex shapes, but our oscillating surface is approximately convex. Unfortunately, the surface of interest is entirely occluded by the nozzle for half of each oscillation and largely occluded by the heat shield the rest of the time.

A very popular method for depth estimation in robotics is the stereo algorithm [5], which recovers shape by finding matching image patches in two different views of the same scene, computing their disparity (the difference in the position in each image) and then solving a set of constraints based on epi-polar lines to figure out the depth of the objects in the images. Unfortunately, this method requires two different views. Given our high frame rate, we would need to synchronize these views to less than 100 microseconds. This might be addressed by using a split mirror to image two different views with the same camera. However, more

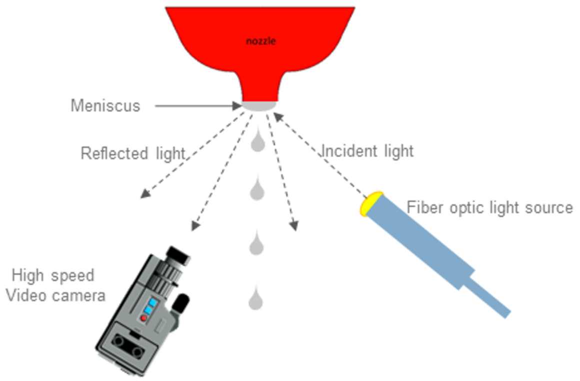

Figure 2. *Apparatus setup showing light source and high-speed video camera*

fundamentally, highly-specular surfaces break one of the assumptions of stereo — that you can match patches between images and compute disparity. Given highly specular objects, the light can change dramatically with a small change in viewpoint, preventing matching. The sparseness of highlights again makes it difficult to gather information about more than a tiny portion of the scene.

LIDAR measures shape based on time of flight of reflected light [7]. LIDAR has rapidly improved and fallen in cost and is now available in compact solid state implementations, but there are none that work with 0.5mm objects. This might be addressable through mirrors or optics, however, the specularity again plays a problematic role. The rays of the LIDAR are scattered off the mirror-like surface of the molten metal droplet preventing an image from being formed.

Recent work in the deep learning community has shown that one can use a trained network to infer depth. It is possible to take sparse depth information and complete shapes [2]. Unfortunately, our information is very sparse and we do not actually have any explicit depth information. Alternatively, a number of researchers have demonstrated monocular depth reconstruction from single images [10] which make use of a large training set of image pairs with matching RGB and depth images. The network learns prior information about the likely depths of everyday scenes such as bedrooms which allow it to predict depth even though the depth in monocular images are inherently unidentifiable (e.g., we do not know if we are looking at a small close object or a large distant object). We do not have the training data required to train these models on the dynamics of the metallic droplets. In addition, the density of information again relies on most surfaces having diffuse Lambertian reflectance models so that we get light from most parts of the scene. There has been a small amount of work with non Lambertian surfaces (e.g., [6]), but these scenes are only weakly non-Lambertian and have dense shape cues. Our droplet scenes have very sparse, highly specular surfaces.

We were not able to find any existing work on estimating depth that would be directly applicable to these highly specialized images of sparse, highly specular, dynamic surfaces.

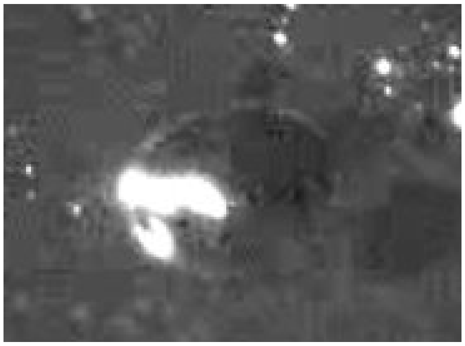

Figure 3. *Closeup of video frame showing approximately 0.5mm circular nozzle, metallic fluid inside and some spatter*

Method

The lab setup includes a high-speed camera mounted beneath and to the side of the nozzle, and pointed up at the aperture. We use the Phantom Veo 710 high-speed camera. The nozzle diameter is approximately 0.5 mm. The scene is illuminated by a Sugarcube Ultra White LED coupled to a fiber optic guide. The apparatus appears in Figure 2. Images are captured using a Mitutoyu 2x macro lens at 640 × 480 pixels and 19,000 frames per second. A high-magnification macro lens is used to make the aperture visible. An example of a closeup of the nozzle can be seen in Figure 3.

Our approach overcomes the limited information in sparse highlights through a strong physics-based prior. We then describe two different ways of using this prior to estimate model state from images.

Physics Model Prior

To introduce a prior from physics, we model the fluid interface as a damped oscillating circular membrane under the force of gravity [1]. In this model, material particles in the membrane are assumed to move mainly along the direction normal to its flat configuration, here the vertical direction. Thus, only vertical displacements from a flat configuration are needed to describe the membrane shape at any time. This problem has a closed-form solution in terms of a series, where the spatial part is expressed using Bessel functions and the temporal part using damped harmonic functions. Since the oscillating surface is approximately convex, we consider the membrane deformation as axisymmetric, and we take only a few low-order terms from the series. With these assumptions, the membrane's vertical displacements is given by a function $D(r,t)$ for each radius $r \in [0,R]$ at time t of the form

$$\begin{aligned} D(r,t) \quad &= \quad d + \left[pR - \frac{p}{R}r^2\right] \\ &\quad +e^{-\gamma_1 t}\sin(\omega_1 t)\left[aRJ_0\left(\lambda_{01}\frac{r}{R}\right)\right] \\ &\quad +e^{-\gamma_2 t}\sin(\omega_2 t)\left[bRJ_0\left(\lambda_{02}\frac{r}{R}\right)\right], \end{aligned} \tag{1}$$

where $\lambda_{01} \approx 2.4048, \lambda_{02} \approx 5.52$ are the first and the second roots of the zeroth order Bessel function J_0. The term $d + \left[pR - \frac{p}{R}r^2\right]$ represents steady-state membrane's shape under the force of gravity, $\gamma_1, \gamma_2 > 0$ are the rates of the exponential decay, and $\omega_1, \omega_2 > 0$ are the corresponding oscillation frequencies. Coefficients a and b determine the relative weights of the two Bessel function

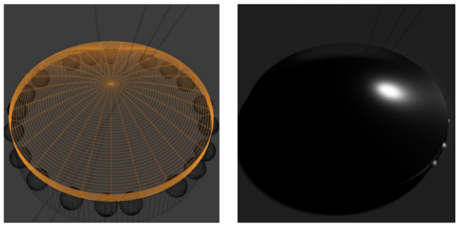

Figure 4. *Radial mesh calculated from equation 1 (left) and ray-traced specular image (right).*

modes.

Inference from Images

Our ultimate goal is to recover the process parameters from lab video of the fluid motions. The physics model (1) gives us a way of going from high-level parameters, such as the oscillation frequency and exponential decay constants, to a liquid air surface shape at some point in time.

To translate the surface shape into an image, we construct a radially symmetric mesh whose height is given by the Bessel approximated vibrating membrane model in Eq. (1). We found that the radial mesh gave a smoother render with fewer artifacts than a rectangular mesh. We use Blender's "SUN" light model which generates parallel rays to approximate the focused LED light source illumination of the droplet. We obtained the approximate position of the light source relative to the camera in terms of azimuth and elevation angles by measuring angles in the lab. The camera model used is the default Blender pinhole camera model. We use the open source Blender "Cycle" ray tracing engine to create an image of the mesh. We observed that the images from the lab of the nozzle showed small highlights around the edge of the nozzle, likely due to machining defects in manufacturing. We observe that the highlights from these defects are active in specific phases of oscillation and therefore are informative. We added a number of randomized spherical artifacts around the outside of our nozzle model to generate similar artifacts in simulated images.

We now have a way of going from the physics model to membrane shape to rendered image. What we need is a method to invert the process, i.e., going from images to shape coefficients (p,d,a,b). This could be done as an optimization where one directly optimizes shape coefficients to alter the mesh and the rendered image until it matches recorded video frames from lab data. A natural measure would be the mean-squared error (MSE) in pixel intensities between the synthesized image and the lab video frame image.

Unfortunately, the sparse nature of specular highlights makes the MSE objective function non-smooth in shape coefficients. To illustrate this, consider the following argument. As it might be the case with a random initialization, suppose that the location of the highlight in one image is distant from the location of the highlight in the lab image, so that they do not overlap at all. If we alter the parameters of the simulator to move the highlight in the synthesized image by a small amount, it will still not overlap with the lab image highlight. This means that the pixel-wise MSE error will not change, and thus that there will be no signal for gradient-based methods to optimize the shape coefficients. It may be that Wasserstein or "earth mover" distance instead of MSE could be used, but these metrics are expensive to calculate. Over and above the cost of computing the metric, the rendering pipeline to generate new images is also very expensive, making this general approach infeasible.

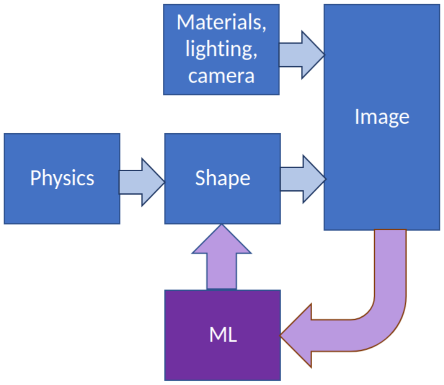

Figure 5. *Forward-model-based rendering and machine learning inversion.*

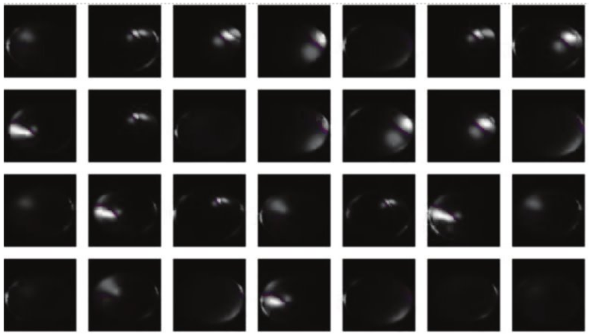

Figure 6. *Subset of dataset showing renderings of membranes corresponding to various shape parameters.*

We decided to employ a machine learning algorithm to go from the images to shape coefficients of the model that generated the images. To simplify the matching process, we ignore the temporal dimension initially and just look at matching frame by frame. The basic idea is to pick some shape coefficients and then ray-trace an image from them and then train a machine learning model using these supervised pairs to go from image to shape coefficients (see Figure 5). We investigate two different machine learning approaches that have distinct trade offs between computational complexity and convenience of introducing temporal priors. The first method is a non-parametric nearest-neighbor method, and the second is a deep convolutional network approach.

We generated data for the machine learning approaches by sampling shape coefficients from uniform ranges determined by looking at video and bounding the reasonable extremes of these parameters. We also sampled a small amount of variation in camera angle and relative light position to make the system robust to small errors in calibration. We sampled batches of 20,000 images and 100,000 images. Examples of renders appear in Figure 6. These images were also augmented by various transforms includ-

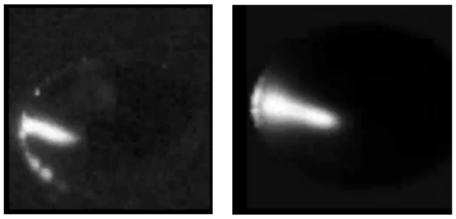

Figure 7. *Lab video frame (left) and closest ray-traced simulation (right).*

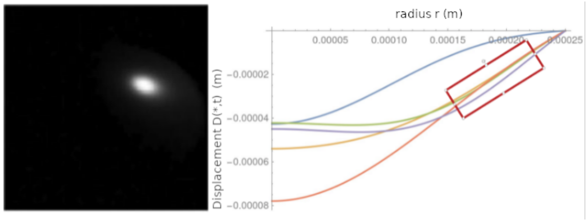

Figure 8. *A lab image and plotted radial profiles ($D(\cdot,t)$) of membranes with resembling rendered images. All of rendered images have similar slopes at the one location where the highlight appears in the lab image, but different slopes at other places where there is no illuminated structure to constrain the model.*

ing small displacements, contrast, noise and blur.

Regularized Nearest-Neighbor-Based Inference

In the nearest-neighbor method, we match a video frame from the lab camera to each of the ray-traced synthetic images and return the ten closest synthetic images along with the shape coefficients used to generate these images. We experimented with various possible metrics for determining the closeness between a lab image and a synthetic image. We compared the matrix 2-norm, Euclidean distance and the perceptual computer vision metric SSIM [9]. We found that matrix 2-norm and Euclidean distances outperformed SSIM. Moroever, Euclidean distance worked as well as matrix 2-norm, so we proceeded with the former, being the most computationally tractable. An example of this identification is shown in Figure 7.

In our analysis of retrieved images we found that images that appear to be similar could have very different shape coefficients. We compared the shape of the radially symmetric membranes used to generate these images by plotting their radial profiles ($D(\cdot,t)$) on a single graph. As shown in Figure 8, all of the radial profiles have a similar slope at one particular location. This means that a fixed light source bounces off all of these shapes in the same way at this location. As a result, the sparse highlight in the lab image provides a constraint for one point on the surface. In contrast, the shape is unconstrained where the image is dark, so there are multiple possible shapes that could be at these locations. This suggests that the problem is only weakly identifiable.

To deal with weak identifiability at the frame level, we introduce a regularization constraint that assumes that the time-sequence of shape coefficients we identify should follow the assumed evolution in Eq. (1), that is, d and p should be constant in time, while the amplitude of each Bessel mode should behave as an exponentially decaying sinusoid in time. In the ideal scenario, out of the multiple synthetic frames that resemble a lab image at each time step, there will be at least one whose shape coefficients are given by the aforementioned time evolution. In reality, we do not observe this, so we resort to identifying a sequence in time of synthetic images that will render a time-sequence of shape coefficients as close as possible to an ideal case.

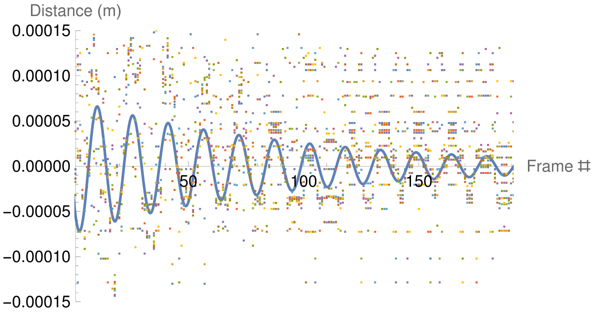

Figure 9. *Inference of decaying sinusoidal oscillation. Vertical slices of 10 closest neighbors to video frame image from dataset of 20k synthesized examples. In blue, the identified function $f(t)$.*

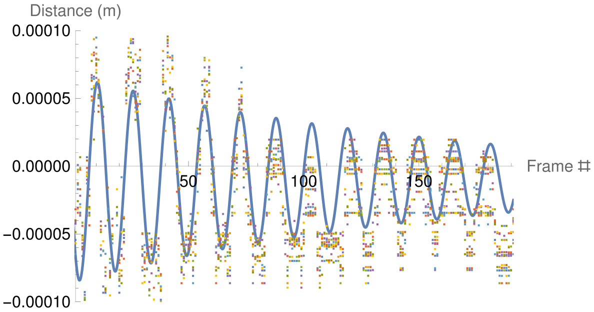

Figure 10. *Inference of decaying sinusoidal oscillation. Vertical slices of 10 closest neighbors to video frame image from dataset of 100k synthesized examples. In blue, the identified function $f(t)$.*

Formally, let $t = t_i,\ldots,t_f$ be the number of time instances. At each time step, a Euclidean norm $\|\cdot\|_2$ is used between lab images and synthetic images to find the 10 best-matching synthetic frames. We denote the set of nearest neighbor membrane shapes at each time t as $\left\{D_j(r,t)\right\}_{j=1,\ t=t_i}^{10,\ t_f}$. To identify the shape coefficients in time, we follow the motion of the center of the membrane, $r = 0$. According to Eq. (1), its vertical displacement should be given by a function of the form

$$f(t) = \alpha e^{-\beta t}\sin(\delta t - \varepsilon) + \zeta,$$

where we only aim at identifying the longest-lasting of the two decaying sinusoids. In this formula, we introduced unknown parameters $\alpha, \beta, \delta, \varepsilon, \zeta$. Notice that we added a phase ε to reflect that the instant $t = 0$ is not well-defined. These coefficients can be mapped to the shape coefficients a, b and $d + pR$ at each time frame.

As discussed earlier, in the ideal case, at least one of the ten runner up synthetic images over every time instance t_n will have shape coefficients for which the vertical displacement at the center will be $f(t_n)$. Since this is seldom the case, we find $f(t)$ that best

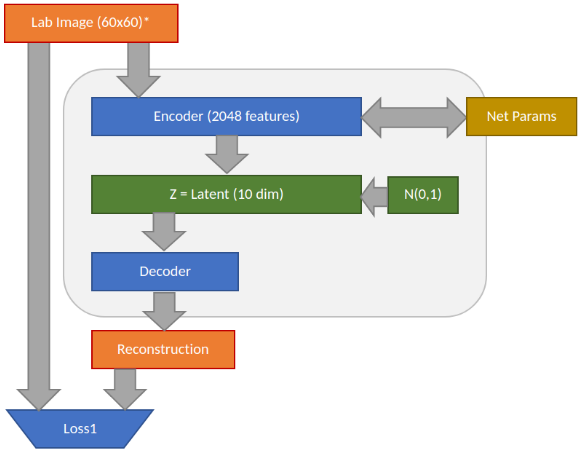

Figure 11. *We train a variational autoencoder (VAE) on lab video images to learn features that pickup relevant details in realistic images from real life.*

matches this condition. Considering this, the problem is to find optimal values of parameters $\alpha,\beta,\delta,\varepsilon,\zeta$ over the synthetic data set $\{D_j(r,t)\}_{j=1,\ t=t_i}^{10,\ t_f}$ by minimizing

$$\left\|\left(\min_j\{|D_j(0,t_i)-f(t_i)|\},\dots,\min_j\{|D_j(0,t_f)-f(t_f)|\}\right)\right\|_2.$$

In our calculations we used a Random Seed variant of a Differential Evolution optimization scheme [8].

We found that it was important to generate a sufficiently large number of samples to cover the lab video images. We saw a dramatic improvement in inference quality when going from a database of 20k samples (Figure 9) to 100k samples (Figure 10).

The regularized nearest neighbor inference provided a smooth stable inference of a decaying sinusoidal process. When we compare the zero crossings of the inferred process with events hand marked on video frames, the alignment is excellent. We were able to infer both frequency of oscillation and exponential decay rate from the inferred process. We observed that the frequency was very stable and precise whereas the decay rate was more variable and unstable.

The nearest neighbor method was computationally intensive and took hours to converge even when taking advantage of parallel computational schemes, making it impractical in its current form for real-time use by engineers or operators.

Deep Neural Network Approach

In this approach, we train a convolutional neural network (CNN) to predict shape coefficients of the membrane that corresponds to an image taken by the lab camera. The obvious approach here is to train on supervised pairs of ray-traced simulation images and their corresponding shape coefficients. Unfortunately, a high-capacity, low-bias neural network has the potential to dramatically overfit to simulation artifacts in order to get high prediction accuracy on the coefficients, leading to poor performance on actual video images. We developed a novel training procedure to minimize overfitting (Figure 11). We trained a variational autoencoder (VAE) [4] on natural video images to create features that are independent of simulations. The autoencoder simply tries to reconstruct its input so it does not need supervised labels. The autoencoder has an encoder stage that projects a 2D image onto a low-dimensional feature vector, and a decoder stage that expands the latent feature vector to a reconstruction of the image using transverse convolutional layers. We used an autoencoder with 10 latent dimensions and found that it worked well.

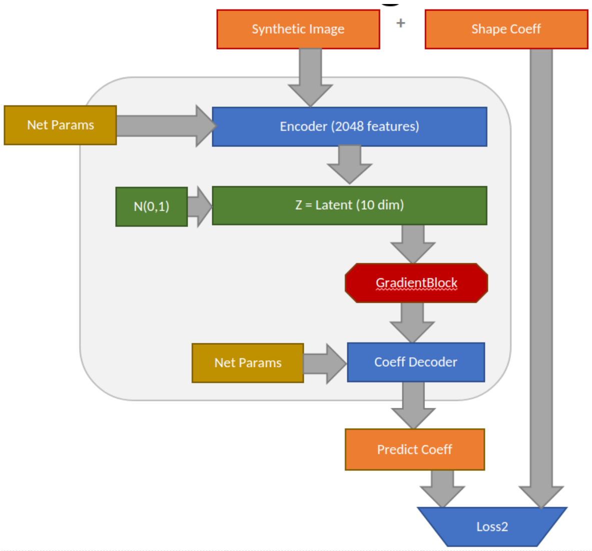

Figure 12. *We train a supervised regression model to predict shape coefficients from synthesized images generated from known coefficients using real world image features learned by VAE.*

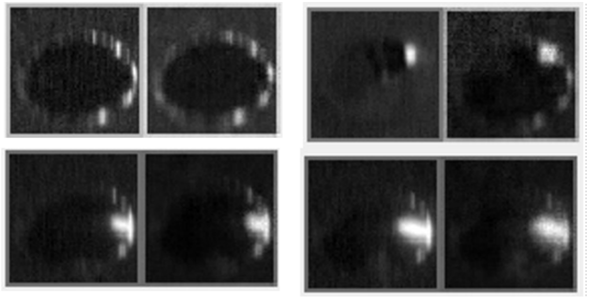

Figure 13. *Real lab video image (left) and VAE reconstructions from features (right) show features capture various membrane shapes well.*

In the second step, we reuse the encoder learned by the VAE to compute natural features from synthetic images (Figure 12). Instead of the original decoder, we substitute a new decoder that performs a regression from the natural image features to the shape coefficients, a *supervised regression model*. Because we are using the natural image features, it is harder for this supervised network to overfit to features of the simulated images.

Before applying the neural network to lab data, we validated the component functions. We verified that the autoencoder captured a wide variety of images corresponding to different possible membrane shapes by testing the reconstruction on many samples. In Figure 13 we can see a number of image pairs in which the lab video image (left) is accurately reconstructed (right).

We then tested that the supervised coefficient regression was

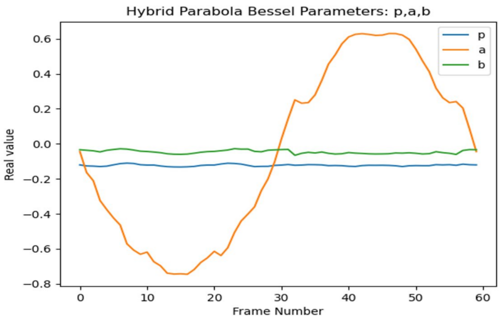

Figure 14. Neural network approach recovers sinusoidal process from individual images frames of synthesized membrane oscillation.

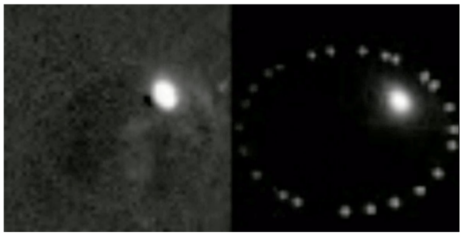

Figure 15. (Left) Lab video frame image. (Right) image synthesized by ray-tracer from shape coefficients inferred by neural network.

working at least approximately. We used the ray-tracing engine to simulate a sequence of video frames corresponding to a varying sinusoidal process with known frequency and amplitude. We then used the regression network on this sequence to extract the dominant coefficients. The network accurately inferred the sinusoidal process as can be seen in Figure 14. There are some small errors in amplitude that correspond to locations where images are not very informative due to the structure of highlights in these images.

We then applied the validated network to an extended lab video recording. As can be seen in Figure 15, the neural network can infer shape coefficients from the image and then use these coefficients with the physics model of the membrane and the Blender ray tracing engine to resynthesize an image that is often very close to the lab video image.

In this video, there are ejections at 100Hz that create high amplitude disturbances to the fluid sequence every 10 ms. As can be seen in Figure 16, the neural network is able to make use of many features included the main highlight and edge artifacts to do frame by frame inference of amplitude that tracks the oscillations well.

On an extended sequence, one can clear see the oscillation rate and decay rate of the process between ejections (Figure 17). We can see that this nozzle is settling fully between ejections.

Future Work

In the near future we plan to setup a new experimental platform to gather additional video data under various conditions to further evaluate the performance of the technique and demonstrate its robustness. To further increase the applicability of the technique, we plan to develop automatic calibration by detecting

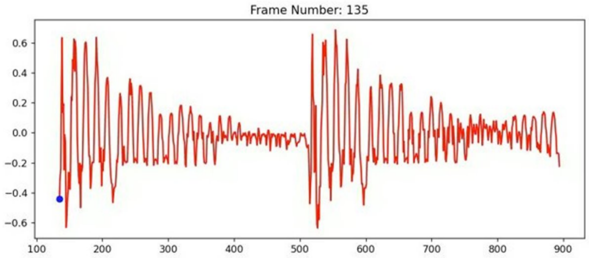

Figure 16. Neural network inference of amplitude from lab video frames.

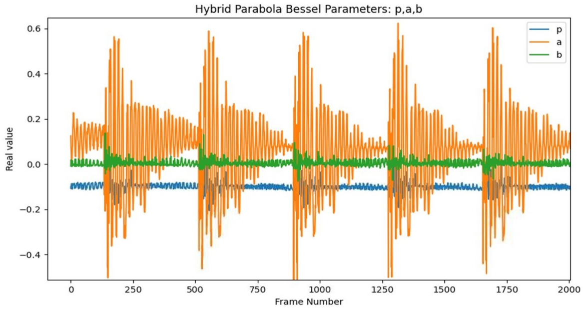

Figure 17. Neural network inference of amplitude of extended sequence can be used to compute oscillation frequency and decay rate.

camera and lighting angles automatically.

Our nearest neighbor implementation showed the promise of this approach but was too slow for practical use. The use of more efficient nearest neighbor techniques such as KD-trees or latent semantic hashing (LSH) which has been applied to high-dimensional objects might be combined with parallel GPU based implementations to increase practicality.

We would like to improve the fidelity of the simulation model and ray-tracing rendering pipeline in multiple ways. The addition of non-radial or asymmetric vibrational modes to the physics model and refining the artifact generation in the simulation around the nozzle perimeter could lead to better inference with the current framework.

In addition to improving the current framework, we have also envisioned ways to change the paradigm. Our present work uses a single white lamp from a single direction which results in very sparse ambiguous information. The use of multiple lights or a pattern projected on a curved reflector to generate incidence from multiple angles could provide many times the amount of information, greatly increasing the accuracy of inference (see Figure 18). Ray-traced simulations show multiple highlights from lights positioned at multiple angles.

The technique described here is based on oscillating membrane but other types of physics models could be used to track different kinds of motion such as Faraday waves on the surface of a liquid (see Figure 19). Similarly, the example here is based on light reflection, but the model could also be applied to different kind of light-surface interactions such as diffraction or interference patterns.

Eventually we plan to examine how some form of visual monitoring could be integrated into real-time control of the printer

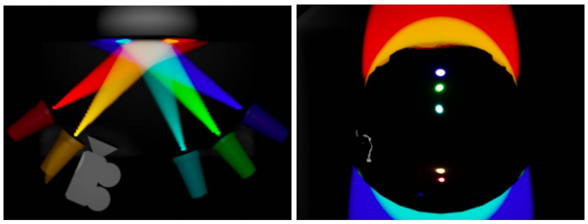

Figure 18. Multiple LED lamps using different colors are used to illuminate the membrane surface or possibly a colored reflector (left). Multiple color highlights provide more coverage than a single highlight from one lamp and colors can be used to distinguish incident angle of lamps forming the highlights (right).

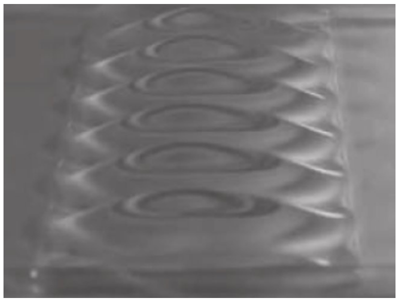

Figure 19. Faraday wave in a vibrating bath.

during operation.

Summary

The inference of shape from oscillating fluids is not easily done with previous techniques due to the unique way in which light interacts with specular fluids, extreme environments and occlusion. We show that the addition of a strong physics-prior together with machine learning inversion of an image rendering pipeline can recover high-resolution inference of fluid shape and behavior. We illustrated two solutions: (a) the regularized nearest neighbor optimization, which yielded very stable and smooth inferences; and (b) a deep learning method using natural features, which has low bias, allowing investigation of non-oscillatory behavior and significantly higher frame rates, suitable for real-time control. In addition to the setting illustrated here, the technique can be generalized to new applications and different kinds of light phenomena. The work here offers engineers and scientists a new tool to recover the shape and motion of oscillating fluids from image based monitoring that can improve both the design optimization and real0time control of industrial processes incorporating fluids.

Acknowledgments

We would like to acknowledge the generous support of the Xerox ElemX team for lab video and the funding of Xerox corporation for this research.

References

[1] George A. Articolo. *Partial Differential Equations Boundary Value Problems with MAPLE (2nd Edition).* Academic Press, 2009.

[2] Xiaoguang Han, Zhen Li, Haibin Huang, Evangelos Kalogerakis, and Yizhou Yu. High-resolution shape completion using deep neural networks for global structure and local geometry inference. In *ICCV*, pages 85–93, 2017.

[3] Berthold K. P. Horn and Michael J. Brooks. *Shape from shading*. 1989.

[4] Diederik P. Kingma and Max Welling. An introduction to variational autoencoders. *Foundations and Trends in Machine Learning*, page 1–18, 2019.

[5] Nalpantidis Lazaros, Georgios Christou Sirakoulis, and Antonios Gasteratos. Review of stereo vision algorithms: From software to hardware. *International Journal of Optomechatronics*, (2 (4)):435–462, 2008.

[6] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially varying lighting and svbrdf from a single image. In *CVPR*, pages 2475–2484, 2020.

[7] Thinal Raj, Fazida Hanim Hashim, Aqilah Baseri Huddin, Mohd Faisal Ibrahim, and Aini Hussain. Survey on lidar scanning mechanisms. *Electronics*, (9(5)):741, 2020.

[8] Rainer Storn and Kenneth Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *J. of Global Optimization*, 11(4):341–359, dec 1997.

[9] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[10] Dan Xu, Elisa Ricci, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *cvpr*, 2017.

Author Biography

***Bob Price** is a member of scientific staff at the Palo Alto Research Center where he investigates probabilistic and deep learning models with custom structures to recover spatial and relational structure of scenes and activities. He got his PhD from the University of British Columbia and did a Post Doctoral Fellowship at the University of Alberta.*

***Svyatoslav Korneev** is a member of scientific staff at the Palo Alto Research Center. He obtained his PhD from the Russian Academy of Sciences in Theoretical and Mathematical Physics.*

***Adrian Jose Lew** is a professor of Mechanical Engineering at Stanford University and consultant at the Palo Alto Research Center, where he investigates technologies for future generations of ElemX. He got his PhD from the California Institute of Technology.*

***Christoforos Somarakis** is a member of scientific staff at the Palo Alto Research Center where he investigates Dynamical Systems for Learning and Control. He received his PhD from the University of Maryland, College Park and did a Post Doctoral Fellowship at Lehigh University.*

***Raja Bala** was a Principal research scientist and leader of the Collaborative Visual Computing Group at PARC at the time of this work. His interests are in computer vision, image processing and machine learning. He received his PhD from Purdue University in Electrical Engineering.*

***Jonathan (Shengtai) Ju** is a research intern at PARC. He is working on his PhD in Electrical and Computer Engineering at Purdue University. His research interests include action recognition, computer vision, video analytics, and image/video processing.*