# 3D Tomographic Reconstruction of a Dynamic Gas Bubble from Dual-Perspective Imaging

Lingzhi Zheng

**Abstract**—Shock-bubble interaction is an important physical process that strongly influences the modeling of supersonic combustion and supernova explosions. Therefore, accurately characterizing the evolution of a gas bubble during interaction with a shock wave is of high interest. In this work, 3D tomographic reconstruction of a gas bubble from two orthogonal views (side-view and end-view) was performed. To develop the reconstruction algorithm, a 3D model of a dummy bubble representative for shock-bubble interaction, assumed to be largely axially symmetric, was first constructed to provide the ground truth. The side-view image (perpendicular to the symmetry axis) was then revolved around the symmetry axis to obtain an axially symmetric approximation. The optical-flow method was utilized to transform the axially symmetric end-view (parallel to the symmetry axis) to the input end-view. The resulting transformation, with a scaling factor to account for the view angle, was used to synthesize projections at various view angles between the side view and end view. These synthesized views, along with a TV term, served as regularizers in the 3D reconstruction. The final reconstruction result obtained using an ADMM solver shows a significant improvement compared to the result obtained using the traditional filtered back projection (FBP) method.

**Index Terms**—Computational Photography

## 1 Introduction

The dynamics of gas bubbles, such as those found in flames and interstellar clouds, during interaction with shock waves is an important topic of research for better understanding gas-dynamic instabilities, which have strong implications for the modeling of supersonic combustion and supernova explosions [1]. Shock-bubble interaction in real-world scenarios often leads to the gas bubble evolving into complex 3D shapes in millisecond timescales, which makes the characterization of bubble morphology challenging. Therefore, a simpler experimental configuration involving a planar shock wave interacting with an initially spherical bubble is often used to validate computational fluid dynamic solvers initially; the validated solvers can then be utilized to simulate more complex scenarios. Nevertheless, while the geometry of the initially spherical bubble during interaction with a planar shock wave is axially symmetric in an ideal experimental setting, small gas-dynamic perturbations that exist in most facilities cause the actual bubble morphology to often deviate from the axial symmetry assumption, contributing to increased uncertainty in the quantification of bubble dynamics. Additionally, experimental investigations of shock-bubble interactions often use apparatuses with limited optical access [2], restricting the number of view perspectives that can be used to observe shock-bubble interaction. A schematic and bubble images from a typical experiment of this kind are shown in Fig. 1. In this context, accurate 3D tomographic reconstruction of dynamic gas bubbles from just two orthogonal views becomes desirable.

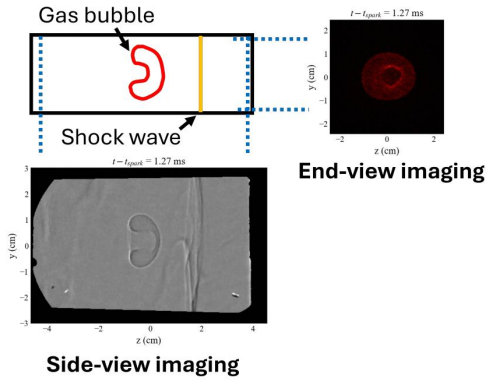


Fig. 1. Example dual-perspective imaging of shock-bubble interaction. The side view is perpendicular to the assumed symmetry axis, whereas the end view is parallel to the symmetry axis.

## 2 Related Work

Many previous works have explored tomographic reconstructions with limited viewing angles. Sparse-view computed tomography (CT) has been explored in medical imaging using techniques such as compressive sensing [3] and deep learning [4]. These studies have observed that reconstructions from sparse views, using approaches developed for abundant views (e.g., filtered back projection or FBP), often result in streaking defects, highlighting the importance of an effective image prior. Zang et al. [5] employed optical-flow-based view interpolation of two projected images as a regularizer for optimizing 3D reconstruction of fluid flow. They particularly emphasized the challenge of applying neural-network-based methods for reconstructing

• *L. Zheng is with the Department of Mechanical Engineering, Stanford University, Stanford, CA, 94305.*
*E-mail: liz147@stanford.edu*

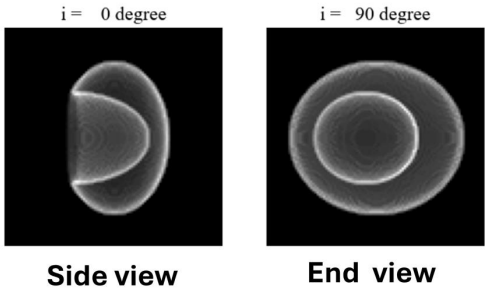


Fig. 2. The side-view and end-view projections of the dummy bubble constructed as the ground truth for 3D tomographic reconstruction.

fluid flows due to the lack of representative training data.

## 3 Proposed Method

In this work, 3D tomographic reconstruction of dynamic gas bubbles using two orthogonal projections, viewed from the side (perpendicular to the assumed symmetry axis) and end (parallel to the assumed symmetry axis) perspectives, was explored. The target largely axially symmetric bubble geometry is representative of that resulting from planar shock wave interaction with an initially spherical bubble, such as the one illustrated in Fig. 1.

### 3.1 Dummy Bubble Construction

First, a 3D model of a dummy bubble was constructed to provide the ground truth for the reconstruction algorithm. The dummy bubble was modeled as an elliptical body with an elliptical indentation from one side. To break the axial symmetry of the dummy bubble, the elliptical indentation were shifted from the symmetry axis of the main elliptical body. To mimic schlieren imaging, which visualizes the gradient of density in a flow field used in actual experiments, the gradient of the constructed dummy bubble were obtained for the computation of projections. The resultant side-view and end-view projections of the dummy bubble are shown in Fig. 2.

### 3.2 Loss Function for Data Fitting

The camera model for the present work is similar to the one adopted by Zang et al. [5] and is expressed as follows:

$$\boldsymbol{f_i} = \boldsymbol{K_i x} + \boldsymbol{n_i}, \tag{1}$$

where $\boldsymbol{f_i}$ is the projection image of the 3D object $\boldsymbol{x}$ at angle $i$, $\boldsymbol{K_i}$ is the Radon transform, and $\boldsymbol{n_i}$ represents noise of the measure. Note that in the present work, the noise term in Eqn. (1) is ignored for simplicity.

The loss function for data fitting ($L_{data}$) can then be expressed as:

$$L_{data} = \sum_{i}^{N} \|\boldsymbol{K_i x} - \boldsymbol{f_i}\|_2^2, \tag{2}$$

where $N$ is the number of views of the input data ($N$ = 2 for dual-perspective imaging). The side-view projection was set as the projection from $i = 0°$, while the end-view projection was set as the projection from $i = 90°$.

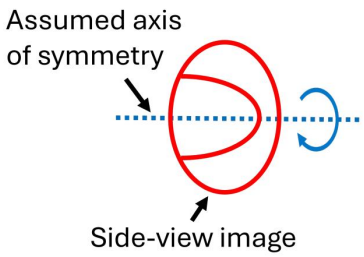


Fig. 3. Schematic for the construction of a axially symmetric approximation.

### 3.3 Optical-Flow-Based View Synthesis

To construct an effective prior for the tomographic reconstruction, the largely axially symmetric nature of the gas bubble was leveraged to first obtain an axially symmetric approximation to the actual bubble geometry, and then optical flow was introduced to compute the transformation between the input end-view image and that from the axial symmetry approximation.

To obtain the axially symmetric approximation, the input side-view image was revolved around the assumed axis of symmetry, as shown in Fig.3. The comparison between the end-view projections of the ground truth and the axially symmetric approximation is shown in Fig.4(left). The TV-L1 optical flow solver, implemented in the Python package skimage [6], was utilized to compute the optical flow. The result from the optical flow calculation is shown in Fig.4 (right).

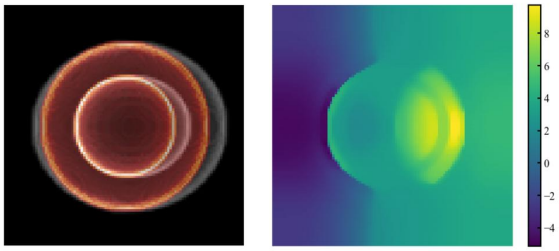


Fig. 4. (Left) comparison of the end-view projection of the ground truth (in gray) and the axially symmetric approximation (in red). (Right) computed optical flow velocity magnitude between the two end-view projections.

To apply the transformation to obtain the synthesized view at angle $j$, a scaling factor of $\sin(j)$ was multiplied to the velocity matrix $\boldsymbol{v}$ to account for the difference shifts at $j$. The view synthesis can then be expressed as:

$$\tilde{\boldsymbol{f}}_{\boldsymbol{j}} = \mathrm{warp}(\boldsymbol{f}_{\boldsymbol{j,sym}}, \boldsymbol{v} \cdot \sin(j)), \tag{3}$$

where $\tilde{\boldsymbol{f}}_{\boldsymbol{j}}$ is the synthesized view at angle $j$, $\boldsymbol{f}_{\boldsymbol{j,sym}}$ is the $j$ projection of the axially symmetric approximation, $\boldsymbol{v} \cdot \sin(j)$ is the transformation to warp the axially symmetric projection to the synthesized view. Sample result from the view synthesis and comparison to the axially symmetric projection as well as the ground truth is shown in Fig. 5.

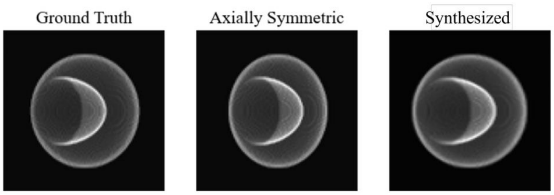


Fig. 5. View at 45° obtained from the ground truth (left), the axially symmetric approximation (center), and view synthesis (right).

### 3.4 Regularization

With the synthesized views obtained, regularizers can be constructed to help the tomographic reconstruction. The view-synthesis regularizer is expressed as follows:

$$L_{syn} = \sum_{j}^{M} \lambda_{syn,j} \|\boldsymbol{K_j x} - \tilde{\boldsymbol{f}}_{\boldsymbol{j}}\|_2^2, \tag{4}$$

where $M$ is the number of synthesized views, and $\lambda syn, j$ is the weight for the regularizer at angle $j$. Since the synthesized projections are closest to the ground truth when they are at 0° or 90°, normalized Gaussian profiles centered at 0° or 90° with a $\sigma$ of 20° were superposed to obtain $\lambda_{syn,j}$.

To improve the smoothness of the reconstruction, a TV regularizer was additionally implemented as follows:

$$L_{TV} = \lambda_{TV} \boldsymbol{D x}, \tag{5}$$

where $\lambda_{TV}$ is the weight of the TV regularizer, and $\boldsymbol{D}$ is the gradient operator.

### 3.5 ADMM Optimization

Alternating direction method of multipliers (ADMM) optimization was used to incorporate the data fitting and regularization terms discussed above for the tomographic reconstruction, and the algorithm is summarized below.

**Algorithm 1** ADMM optimization

Initialize $\lambda_{syn,j}$, $\lambda_{TV}$, $\rho$
$\boldsymbol{x} = zeros(W, H, T)$
$\boldsymbol{z} = zeros(W, H, T, 3)$
$\boldsymbol{u} = zeros(W, H, T, 3)$
**for** j=0 to 180 **do**
  $\tilde{\boldsymbol{f}}_{\boldsymbol{j}} = \text{warp}(\boldsymbol{f_{j,sym}}, \boldsymbol{v} \cdot \sin(j))$
**end for**
**for** k=0 to max_iters **do**
  $\boldsymbol{x} = \text{cg\_solve}(\sum_i^N \boldsymbol{K_i}^T \boldsymbol{K_i} + \sum_j^M \lambda_{syn,j} \boldsymbol{K_j}^T \boldsymbol{K_j} + \rho \boldsymbol{D}^T \boldsymbol{D}, \sum_i^N \boldsymbol{K_i}^T \boldsymbol{f_i} + \sum_j^M \lambda_{syn,j} \boldsymbol{K_j}^T \tilde{\boldsymbol{f}}_{\boldsymbol{j}} + \rho \boldsymbol{D}^T \boldsymbol{z})$
  $\boldsymbol{z} = S_{\lambda_{TV}/\rho}(\boldsymbol{Dx} + u)$
  $\boldsymbol{u} = \boldsymbol{u} + \boldsymbol{Dx} - \boldsymbol{z}$
**end for**

$W$, $H$, and $T$ are the width, height, and thickness of the 3D object; $\rho$ is the factor for the penalty term; $S_{\lambda_{TV}/\rho}$ is the soft thresholding operator. cg_solve represents the conjugate gradient solver (the scipy [7] implementation was adopted), and max_iters is the maximum number of iterations used in the ADMM solver.

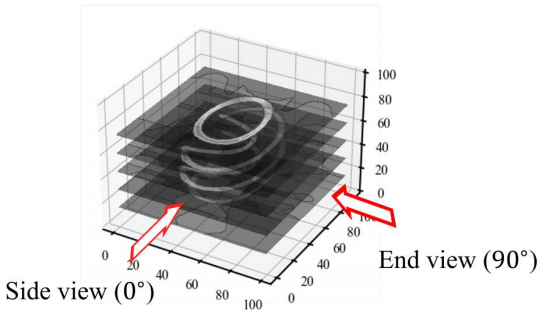


Fig. 6. Schematics of horizontal slices sampled from the 3D object reconstructed using the ADMM+TV solver proposed in the present work. The axes are in pixels.

## 4 Experimental Results

Figure 6 shows the configuration of horizontal cross-section slices sampled from the 3D objected reconstructed using the ADMM+TV solver with view-synthesis regularization. The side-view and end-view angles are also illustrated for reference.

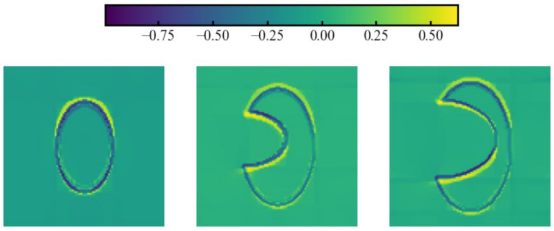


Fig. 7. Normalized error computed for the horizontal slices from the reconstructed 3D object at height = 25 pixel (left), 35 pixel (center), and 50 pixel (right).

Figure 7 shows the normalized error, $\epsilon$, of the reconstruction results from the proposed method. The normalized error is defined as $\epsilon = (S_{recon} - S_{GT})/\max(S_{GT})$ where $S_{recon}$ and $S_{GT}$ are horizontal cross-section slices of the reconstructed 3D object and the ground truth, respectively. The cross-section contours of the reconstructed 3D object can be seen in Fig. 7 to largely close to that of the ground truth, but notable deviations can also be observed. The reconstructed contours, visible in Fig. 7 as areas with $\epsilon > 0$, slightly mismatch those of the ground truth, and particularly, the elliptical indentation can be seen to have shifted down in the reconstructed contours. This is likely a result of the fact that the synthesize views do not perfectly return the projection of the ground truth. To improve the accuracy of the reconstruction, future works could investigate a more optimal approach for view synthesis.

Figure 8 shows the horizontal cross-section slices at different height extracted from reconstruction results. To compare the present approach with other reconstruction strategies, the results obtained using the traditional FPB method (implemented as iradon in skimage) is also shown in Fig. 8; to show the effect of the ADMM solver with TV regularization, the results obtained from a gradient descent (implemented in the Adam solver in the python package

pytorch [8]) solver considering only the view-synthesis regularizer are also shown for comparison.

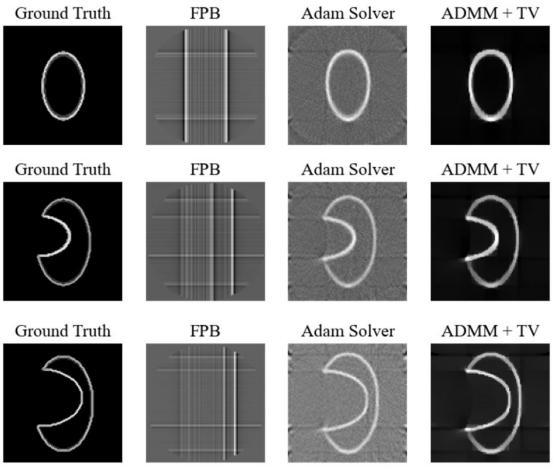


Fig. 8. Horizontal slices of the 3D tomographic reconstruction results at different heights obtained from the FBP, Adam solver, and ADMM solver with TV regularization.

From Fig. 8, the superior performance of the ADMM+TV solver with view-synthesis regularization can be clearly observed. As expected, the results from FPB, essentially a direct inversion of the Radon transform, show only streak defects due to the highly ill-posed nature of the reconstruction. With view-synthesis regularization implemented, the results from the gradient descent solver (Adam solver) without TV regularizer largely return the outline of the ground truth, but large levels of noise were observed.

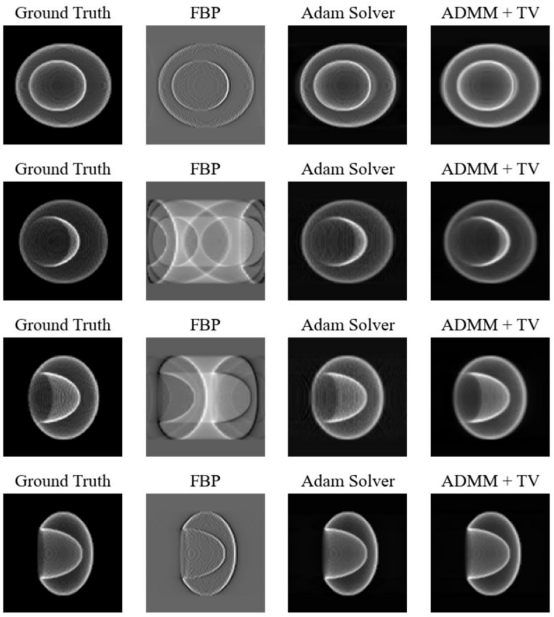


Fig. 9. Projection results at 90° (first row), 60° (second row), 30° (third row), and 0° (forth row) obtained using the FBP, Adam solver, and ADMM+TV solver approaches.

Figure 9 shows the resultant projections at various angles obtained from different reconstruction methods. The peak signal to noise ratios (PSNR) for each image shown in Fig. 9 computed relative to the projections of the ground truth are tabulated in Table 1.

TABLE 1
PSNR relative to the ground truth for projections at various angles of the 3D object reconstructed using the FPB, Adam solver, and ADMM+TV methods.

| | FBP | Adam Solver | ADMM+TV |
|---|---|---|---|
| 0° | 8.83 | 19.80 | 23.12 |
| 30° | 8.42 | 15.15 | 16.04 |
| 60° | 8.22 | 15.95 | 16.04 |
| 90° | 5.47 | 18.15 | 16.82 |

The FPB results can be seen in Fig 9 to only produce reasonable projections at 0° and 90°, where the projections are given in the input, while the Adam solver and ADMM+TV solver show very little qualitative difference. From Table 1, the ADMM+TV solver shows higher PSNR for most projection, particularly compared to results obtained using FBP, whereas, the Adam solver's PSNR at 30° and 60° are only lower than that of ADMM+TV solver by less than 1 dB. At 0°, the ADMM+TV solver produces significantly higher PSNR relative to the Adam solver (23.12 verses 19.80), while at 90°, the Adam solver result is higher than the ADMM+TV result (18.15 verses 16.82). The lower ADMM+TV PSNR at the end-view projection is also qualitatively visible in Fig. 9, as blurrier edges, likely caused by TV regularization, can be seen in the ADMM+TV result.

## 5 Conclusion

In this work, 3D tomographic reconstruction of gas bubbles using projections from two orthogonal views was attempted using optical-flow-based view synthesis regularization. The mostly axially symmetric nature of the gas bubble, modeled after the geometry found in shock-bubble interaction experiments, was taken to construct a axially symmetric approximation to the ground truth; optical flow was then used to compute a geometry transformation to warp the approximations to synthesized views. An ADMM solver with additional TV regularization was used to perform the reconstruction.

The reconstruction result from the proposed method largely captures geometric features of the ground truth, but slight mismatches in the horizontal cross-section contours were observed. The results from the ADMM+TV solver with view synthesis were compared with those of the traditional FBP method and a gradient descent solver (Adam solver) with view synthesis. The ADMM+TV result was largely able to qualitatively recover the ground truth free of streaks or noise defects, and the resultant projections of the reconstructed 3D object show higher PSNR at view angles where view synthesis was applied. However, at 90°, the ADMM+TV projection results show lower PSNR than those of the Adam solver, which is likely due to a limitation of the TV regularization.

## ACKNOWLEDGMENTS

The author would like to thank Alex Levy for serving as the mentor for this project.

## REFERENCES

[1] D. Ranjan, J. Oakley, and R. Bonazza, "Shock-bubble interactions," *Annual Review of Fluid Mechanics*, vol. 43, pp. 117–140, 2011.

[2] Z. Zhai, L. Zou, Q. Wu, and X. Luo, "Review of experimental richtmyer–meshkov instability in shock tube: from simple to complex," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 232, no. 16, pp. 2830–2849, 2018.

[3] G.-H. Chen, J. Tang, and S. Leng, "Prior image constrained compressed sensing (piccs): a method to accurately reconstruct dynamic ct images from highly undersampled projection data sets," *Medical physics*, vol. 35, no. 2, pp. 660–663, 2008.

[4] Y. Han and J. C. Ye, "Framing u-net via deep convolutional framelets: Application to sparse-view ct," *IEEE transactions on medical imaging*, vol. 37, no. 6, pp. 1418–1429, 2018.

[5] G. Zang, R. Idoughi, C. Wang, A. Bennett, J. Du, S. Skeen, W. L. Roberts, P. Wonka, and W. Heidrich, "Tomofluid: Reconstructing dynamic fluid from sparse view videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1870–1879.

[6] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in python," *PeerJ*, vol. 2, p. e453, 2014.

[7] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[8] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.