Computer Animation and Virtual Worlds

WILEY

SPECIAL ISSUE PAPER

Decoupling Density Dynamics: A Neural Operator Framework for Adaptive Multi-Fluid Interactions

Yalan Zhang[1,2] | Yuhang Xu[1,2] | Xiaokun Wang[1] | Angelos Chatzimparmpas[3] | Xiaojuan Ban[1]

[1]School of Intelligence Science and Technology, University of Science and Technology Beijing, Beijing, China | [2]Shunde Innovation School, University of Science and Technology Beijing, Guangdong, China | [3]Department of Information and Computing Sciences, Utrecht University, Utrecht, the Netherlands

Correspondence: Yalan Zhang (zhangyl@ustb.edu.cn)

Received: 25 April 2025 | **Accepted:** 8 May 2025

Funding: This research was supported by the National Natural Science Foundation of China (No. 62306032), Guangdong Basic and Applied Basic Research Foundation (No. 2022A1515110350), Interdisciplinary Research Project for Young Teachers of USTB (FRF-IDRY-GD24-003). The computing work is partly supported by the MAGICOM Platform of Beijing Advanced Innovation Center for Materials Genome Engineering.

Keywords: density-conditioned convolution | Lagrangian simulation | multi-phase fluid dynamics | neural particle methods

ABSTRACT

The dynamic interface prediction of multi-density fluids presents a fundamental challenge across computational fluid dynamics and graphics, rooted in nonlinear momentum transfer. We present Density-Conditioned Dynamic Convolution, a novel neural operator framework that establishes differentiable density-dynamics mapping through decoupled operator response. The core theoretical advancement lies in continuously adaptive neighborhood kernels that transform local density distributions into tunable filters, enabling unified representation from homogeneous media to multi-phase fluid. Experiments demonstrate autonomous evolution of physically consistent interface separation patterns in density contrast scenarios, including cocktail and bidirectional hourglass flow. Quantitative evaluation shows improved computational efficiency compared to a SPH method and qualitatively plausible interface dynamics, with a larger time step size.

1 | Introduction

Fluid simulation with multi-density interactions remains a critical challenge in computer graphics and computational physics. While position-based neural solvers [1, 2] have demonstrated remarkable success in homogeneous fluid simulation, their extension to multi-phase scenarios with varying density ratios (e.g., oil-water separation, magma-water interaction) introduces fundamental limitations. Traditional neural architectures process density as passive input features, failing to explicitly model the dynamic coupling between fluids with contrasting material properties.

Systematic evaluation identifies several key challenges in existing learning-based approaches when applied to multi-density fluids: (1) Static convolution kernels cannot adapt to density-dependent material responses, causing error accumulation at fluid interfaces; (2) Fixed-dimensional feature encoding limits generalization to unseen phase combinations.

To address these limitations, we propose Density-Conditioned Dynamic Convolution, a novel neural operator framework. Its core innovation lies in establishing explicit mappings between local density distributions and material responses. Specifically, we contribute:

- A novel Density-Conditioned Dynamic Convolution module that uses continuously adaptive neighborhood kernels to model density-dependent interactions.
- A relative density encoding scheme that decouples feature learning from absolute density values, enabling generalization to arbitrary phase combinations.

These contributions collectively advance the state-of-the-art in neural particle methods for multi-density fluid simulation by explicitly addressing the challenges posed by varying material properties.

2 | Related Work

2.1 | Traditional SPH-Based Multi-Phase Simulation

Multi-phase fluid systems present unique challenges in handling density discontinuities at fluid interfaces. Erroneous density estimation at phase boundaries propagates to velocity and pressure miscalculations, necessitating specialized interface treatment. Colagrossi et al. [3] pioneered the weakly compressible model using particle volume instead of density for pressure gradient computation, effectively bypassing density discontinuities at gas-liquid interfaces through surface tension terms. However, their method required prohibitively small time steps for stability. Hu et al. [4] introduced interfacial pressure averaging between phases, employing distinct approximation functions for cross-phase interactions. While innovative, this approach violated mass conservation principles, leading to simulation divergence in long-term scenarios.

Building on these foundations, Hu and Adams [5] formulated an incompressible multi-phase SPH method using phase-aware gradient/divergence operators. Their Poisson-based projection method achieved enhanced interface stability at the cost of increased computational complexity. Grenier and Colagrossi [6] derived pressure terms through Lagrangian variational principles, but required solving ill-conditioned matrices for particle volume distribution—a critical bottleneck in large-scale simulations. Monaghan et al. [7] proposed a pragmatic solution using artificial repulsion forces at interfaces, sacrificing physical accuracy for computational efficiency. This approach proved inadequate for turbulent flows due to unphysical constraints on interfacial particle motion.

Recent advances focus on density ratio robustness. Chen et al. [8] developed a novel density smoothing scheme by treating cross-phase neighbors as intra-phase particles, coupled with innovative density initialization and pressure correction terms, but did not include an artificial repulsion term, resulting in a rougher simulated two-phase interface. Krimi et al. [9] enhanced interface sharpness through continuous pressure surface equations, yet faced challenges in parameter selection without quantitative criteria. The most significant breakthrough comes from Xu et al. [10], who replaced the traditional mass-density model with a volume-flux divergence-free formulation. This paradigm shift eliminated interface instability by decoupling density ratios from numerical approximations, setting new benchmarks for high-density contrast simulations.

These methodological evolutions demonstrate three persistent challenges in SPH-based methods for multi-phase fluids: (1) Momentum conservation at dynamic interfaces, (2) Prevention of phase particle aggregation without resorting to non-physical forces, and (3) Computational efficiency for interactive or real-time applications. Current SPH-based approaches must balance between physical accuracy and numerical stability through either enhanced mathematical formulations or empirical corrections—a trade-off our neural approach aims to resolve through learned dynamics.

2.2 | Machine Learning for Fluid Simulation

Recent advances in machine learning (ML) have opened new frontiers in fluid simulation by leveraging data-driven pattern recognition. In contrast to Eulerian grid-based methods, particle-based approaches deal with unstructured data, which poses unique challenges for neural network modeling. Stanton et al. [11] introduced a precomputed state graph method for real-time fluid simulation, where each graph vertex represents the position and velocity of particles in a frame, and each edge denotes transient connections between states. Similarly, Sanchez-Gonzalez et al. [12] proposed Graph Network-based Simulators (GNS), which use particles to represent the state of a physical system. By learning message-passing mechanisms between edges, GNS can simulate fluids, rigid solids, and deformable materials in a unified framework.

The similarity between unstructured Lagrangian particle sets and point cloud models has inspired the adoption of point cloud processing techniques in Lagrangian systems across interdisciplinary research. Qi et al. [13] pioneered PointNet, a network specifically designed for point cloud processing. PointNet employs multi-layer perceptrons (MLPs) to extract features and uses max pooling to ensure permutation invariance, a crucial property for point cloud data. The following year, Qi et al. [14] introduced PointNet++, which addressed the limitation of PointNet in capturing only global features. PointNet++ incorporates sampling and grouping strategies to hierarchically extract local features, enhancing its ability to model fine-grained structures.

In the context of fluid simulation, Kashefi et al. [15] were among the first to apply a PointNet-based deep learning strategy to predict incompressible velocity and pressure fields around cylinders with varying cross-sectional shapes. In industrial applications, Jiang et al. [16] utilized a PointNet-based model to reconstruct high-resolution vascular flow models. Here, the vascular geometry was represented as a point cloud, and the PointNet model was used to extract spatiotemporal features of the time-varying flow field, which were then decoded into high-fidelity flow reconstructions. For sequential point cloud data, PointRNN [17] was proposed, leveraging recurrent neural networks (RNNs) to predict the motion trajectories of 3D point clouds.

Schenck et al. [18] leveraged fully convolutional neural networks to detect and track liquid dynamics in pouring sequences. They

proposed SPNets, a novel network architecture featuring a custom ConvSP layer to compute fluid-particle interaction forces and a ConvSDF layer to model particle-static object interactions. Meanwhile, Miyanawala et al. [19] combined convolutional neural networks (CNNs) with stochastic gradient descent to simplify the modeling of unsteady Navier-Stokes equations. Their approach uses nonlinear corrected discrete convolutions to approximate the mapping between bluff body shapes and fluid forces, offering a computationally efficient alternative to traditional methods.

Continuous Convolution (CConv) methods introduced SPHerical filter-based convolution operations for point clouds. Ummenhofer et al. [2] utilized convolutional neural networks to process groups of moving particles representing fluids in space and time. Unlike previous methods, CConv does not construct explicit graph structures to connect particles. Instead, it employs spatial convolution as a differentiable operation to associate particles with their neighbors. This approach excels in using separate networks to extract static and dynamic particle features, enabling the learning of fluid-solid interaction phenomena. In terms of simulation quality, CConv demonstrates advantages over the state-of-the-art DFSPH [20] method in certain scenarios. Hybrid methods like Wessels et al. [21] combined Physics-Informed Neural Networks (PINNs) with updated Lagrangian formulations for incompressible free-surface flows. Recent studies have further extended the theoretical boundaries of CConv, such as the dynamic multi-scale grid method [22], which significantly reduces computational complexity by analyzing multi-scale recursive characteristics of particle motion patterns in space-time domains. Notably, the generalized CConv framework based on separable basis functions [23] establishes a more universal mathematical formulation, and experimentally demonstrated that even and odd function components contained in the basis functions are key aspects of stability and accuracy. Prantl et al. [1] further enhanced momentum conservation through specialized network architectures. Related work on kernel-based methods for probability density estimation, such as Kernel Conditional Density Operators [24], also explores adaptive kernel design based on local data properties, providing a broader context for our density-conditioned approach.

3 | Methodology

3.1 | Neural Solver Framework

Our neural solver builds upon the Lagrangian formulation of fluid dynamics, where particle interactions are modeled through continuous convolution operators. Let $\mathbf{X}^t = \{\mathbf{x}_i^t\}_{i=1}^N$ denote particle positions at time t and $\mathbf{V}^t$ their velocity field. The temporal evolution follows:

$$\mathbf{X}^{t+\Delta t} = \mathcal{F}(\mathbf{X}^t, \mathbf{V}^t, \mathbf{A}^t) + \epsilon(\mathbf{X}^t) \tag{1}$$

where $\mathcal{F}$ encapsulates inertial dynamics (e.g., gravity, external forces) and $\epsilon(\cdot)$ represents the neural correction term. The inertial component adheres to Verlet integration:

$$\mathbf{X}^{t,\prime} = \mathbf{X}^t + \Delta t \mathbf{V}^t + \frac{1}{2}\Delta t^2 \mathbf{A}^t \tag{2}$$

The neural correction $\epsilon(\mathbf{X}^t)$ is then applied to the inertially predicted positions $\mathbf{X}^{t,\prime}$ to obtain the final updated positions $\mathbf{X}^{t+1} = \mathbf{X}^{t,\prime} + \epsilon(\mathbf{X}^t)$, addressing errors and incorporating complex learned dynamics.

The neural correction $\epsilon(\mathbf{X}^t)$ employs continuous convolutions with density-conditioned modulation:

$$\epsilon(\mathbf{x}_i^t) = \sum_{j \in \mathcal{N}_R(\mathbf{x}_i^t)} \Gamma_{ij}^\rho \cdot g(\Lambda(\mathbf{x}_i^t - \mathbf{x}_j^t)) \cdot w(\|\mathbf{x}_i^t - \mathbf{x}_j^t\|) \tag{3}$$

where $\Gamma_{ij}^\rho = \gamma_{ij}^\rho \odot f_j + \beta_{ij}^\rho$ denotes the density-dependent modulation operator, with $\gamma_{ij}^\rho, \beta_{ij}^\rho \in \mathbb{R}^d$ generated through density-ratio transformations. Here, $\mathbf{x}_i^t$ is the position of particle i at time t, $\mathcal{N}_R(\mathbf{x}_i^t)$ is the set of neighbors of particle i within radius R, g is a learnable kernel, Λ is a coordinate mapping function for a unit ball to a unit cube to implement SPHerical filters, $w(\cdot)$ is a smooth kernel function (e.g., cubic spline) that goes to zero outside the radius R, f_j are features associated with particle j, and d is the feature dimension.

3.2 | Density-Conditioned Convolution

The core innovation lies in the density-aware feature transformation. For particle i with density ρ_i, we construct density embeddings through nonlinear projection:

$$\mathbf{e}_i = \phi\left(\log\left(1 + \frac{\rho_i}{\rho_{\text{ref}}}\right)\right), \quad \phi : \mathbb{R} \to \mathbb{R}^{d_e} \tag{4}$$

where ρ_{ref} is a reference density and d_e the embedding dimension. Particle-pair interactions are then modulated by:

$$\begin{aligned} \gamma_{ij}^\rho &= \sigma\left(\mathbf{W}_\gamma[\mathbf{e}_i \oplus \mathbf{e}_j]\right) \\ \beta_{ij}^\rho &= \mathbf{W}_\beta[\mathbf{e}_i \oplus \mathbf{e}_j] \end{aligned} \tag{5}$$

where $\oplus$ denotes concatenation of the density embeddings $\mathbf{e}_i$ and $\mathbf{e}_j$, $\mathbf{W}_\gamma$ and $\mathbf{W}_\beta$ are learnable weight matrices, and σ the sigmoid activation.

3.3 | Theoretical Analysis

The density-conditioned operator maintains continuity with respect to density variations. Let $\rho_j = \alpha\rho_{\text{ref}}$ and $\rho_i = \beta\rho_{\text{ref}}$, the modulation parameters satisfy:

$$\begin{aligned} \frac{\partial \gamma_{ij}}{\partial \alpha} &= \sigma'(\cdot)\mathbf{W}_\gamma^{(1)}[\phi'(\log(1+\alpha))/\alpha + 1] \\ \frac{\partial \beta_{ij}}{\partial \beta} &= \mathbf{W}_\beta^{(1)}[\phi'(\log(1+\beta))/\beta + 1] \end{aligned} \tag{6}$$

where $\sigma'(\cdot)$ is the derivative of the sigmoid function, and $\mathbf{W}_\gamma^{(1)}, \mathbf{W}_\beta^{(1)}$ represent the first layer transformations within the MLPs generating γ_{ij}^ρ and β_{ij}^ρ respectively. This smooth dependence ensures stable gradient propagation during optimization. Moreover, the operator preserves physical consistency through:

$$\lim_{(\alpha,\beta)\to(1,1)} \Gamma_{ij}^\rho = \mathbf{I}_d \Rightarrow \epsilon(\mathbf{x}_i^t) \to \epsilon_{\text{base}}(\mathbf{x}_i^t) \tag{7}$$

ALGORITHM 1 | Density-Conditioned Fluid Simulation.

Initialize particle positions $\mathbf{X}^0$, densities ρ^0, neighborhood radius R
for $t = 0$ to $T - 1$ **do**
  Calculate inertial motion:

$$\mathbf{X}^{t,\prime} \leftarrow \mathcal{F}(\mathbf{X}^t, \mathbf{V}^t, \mathbf{A}^t)$$

  Build neighborhood graph $\mathcal{G}^t$ based on $\mathbf{X}^t$
  Density embeddings $\boldsymbol{\epsilon}_i^t$ via Equation (4) for all particles i
  Generate $\gamma_{ij}^t, \beta_{ij}^t$ via Equation (5) for all particle pairs (i, j) in $\mathcal{G}^t$
  Calculate neural correction:

$$\epsilon(\mathbf{X}^t) \leftarrow \sum_{j \in \mathcal{N}_R(\mathbf{x}_i^t)} \Gamma_{ij}^{\rho} \cdot g(\Lambda(\mathbf{x}_i^t - \mathbf{x}_j^t)) \cdot w(\|\mathbf{x}_i^t - \mathbf{x}_j^t\|)$$

  Update positions: $\mathbf{X}^{t+1} \leftarrow \mathbf{X}^{t,\prime} + \epsilon(\mathbf{X}^t)$
  Update velocities:

$$\mathbf{V}^{t+1} \leftarrow (\mathbf{X}^{t+1} - \mathbf{X}^t)/\Delta t$$

end for

recovering the baseline single-density model when material densities converge.

The architecture's generalization capability stems from its nonlinear factorization of density effects. Through tensor decomposition analysis, we observe:

$$\Gamma_{ij}^{\rho} \approx \sum_{k=1}^{K} \lambda_k(\rho_i)\mu_k(\rho_j)\mathbf{U}_k \tag{8}$$

where $\{\lambda_k, \mu_k\}$ are learned basis functions and $\mathbf{U}_k$ orthogonal interaction modes. This separable structure enables extrapolation to unseen density combinations through basis recombination.

The complete simulation procedure, as formalized in Algorithm 1, is summarized below.

3.4 | Implementation Details

The network architecture consists of a series of density-conditioned convolution layers. For the density embedding function ϕ, we use a small MLP with ReLU activation. The modulation generation functions $(\mathbf{W}_\gamma[\cdot], \mathbf{W}_\beta[\cdot])$ are also implemented as MLPs. The kernel function $w(\cdot)$ is a standard smooth kernel, e.g., the cubic spline kernel commonly used in SPH. Training is performed using a physics-aware learning strategy, starting with single-phase simulations and gradually introducing multi-density scenarios with increasing density ratios. The loss function minimizes the difference between the predicted particle positions and ground truth positions derived from a high-fidelity SPH solver (IMM Mixture [10]) on a set of training trajectories. Specific hyperparameters, such as network layer sizes, learning rates, were tuned based on validation performance and simulation stability.

4 | Results

Our experimental framework rigorously evaluates the proposed density-aware convolutional architecture through several emblematic scenarios: A multi-layered fluid cocktail simulation and a bidirectional hourglass flow, and a triphasic canyon flow. These configurations leverage the core innovation of our model—processing sequential particle states (positions, velocities, and densities) to recursively predict future dynamics while preserving intrinsic material properties. Figures 1–3 qualitatively illustrate the simulation outcomes. The sub-figures within Figures 1–3 represent different time points during the simulation, progressing from left to right or top to bottom.

4.1 | Four-Phase Fluid Cocktail Dynamics

A cocktail glass containing four immiscible fluids with various densities was simulated under gravitational forcing. Initialized in a vertically layered configuration, the system was perturbed through wall-boundary collisions, inducing transient phase mixing that challenged the model's ability to recover intrinsic stratification dynamics.

The computational framework demonstrated robust density discrimination, autonomously resolving chaotic interfacial patterns into stable horizontal stratification through progressive phase separation. As shown in Figure 1, light-density fluids exhibited upward migration against gravitational forces, forming distinct surface-adherent layers, while higher-density phases underwent controlled sedimentation trajectories. Interfacial boundaries evolved from diffuse mixing states to sharply defined stratification planes, with momentum conservation maintained throughout the transient reorganization process. Crucially, the architecture autonomously reconciled competing adhesion and

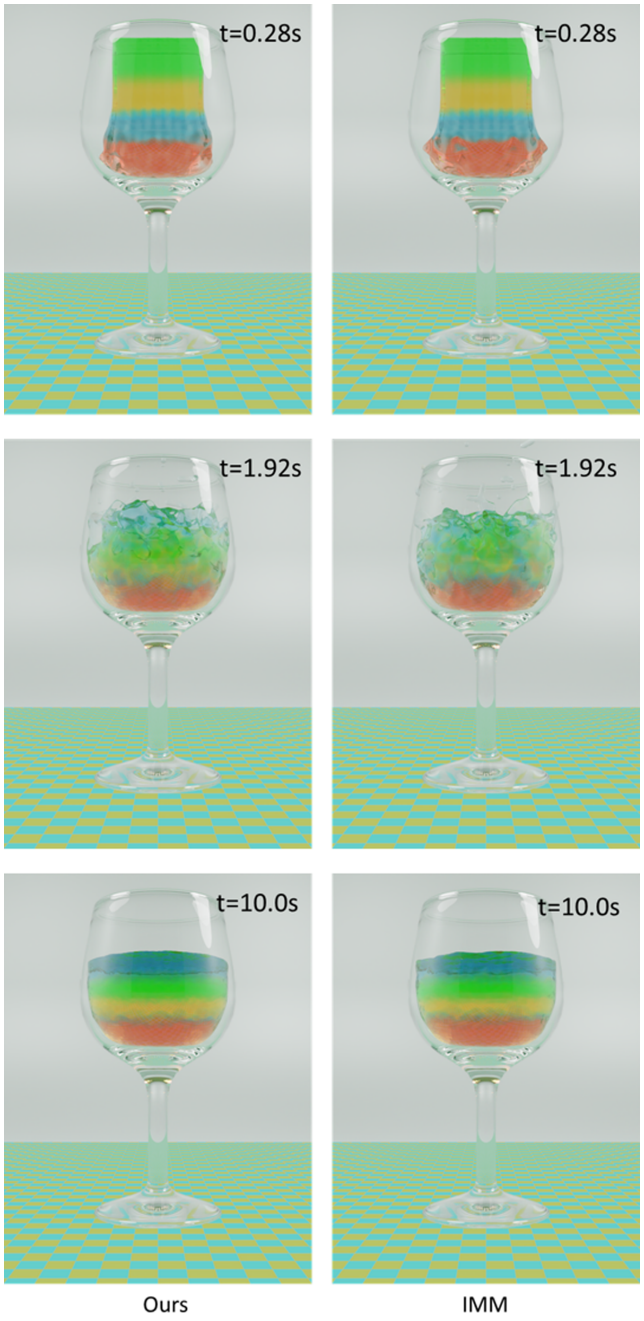

FIGURE 1 | Four-phase fluid cocktail.

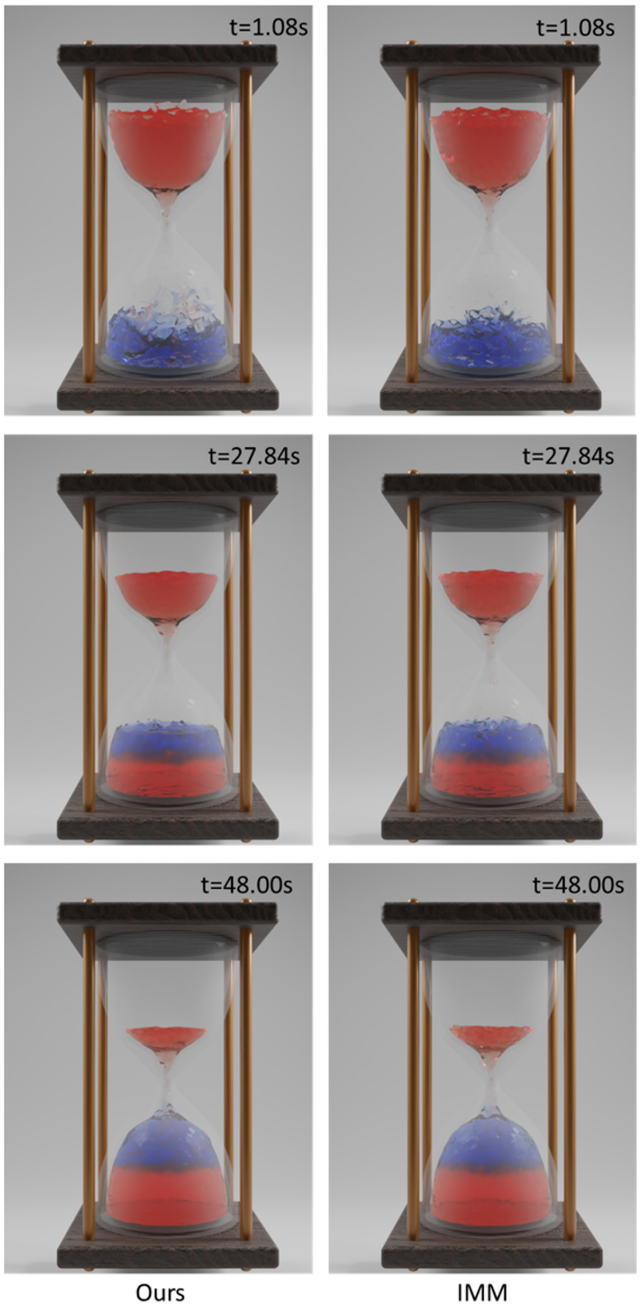

FIGURE 2 | Bidirectional hourglass fluid.

buoyancy forces without manual parameter tuning, achieving equilibrium configurations that aligned with continuum mechanical expectations for multi-layered immiscible systems.

4.2 | Bidirectional Hourglass Flow

A dual-density hourglass configuration was implemented to examine dynamic phase inversion phenomena. A high-density fluid released from the upper chamber penetrated the lower low-density medium, generating characteristic surface wave propagation and transient vortex structures. The model successfully captured nonlinear interfacial dynamics, including droplet coalescence during gravitational descent and shear-induced entrainment at the constriction neck.

Over extended simulation horizons, the system exhibited spontaneous phase redistribution driven solely by learned density contrasts. In Figure 2, the primary high-density phase accumulated in the lower reservoir while secondary counterflow patterns emerged, with low-density particles migrating upward through viscous coupling effects. This bidirectional transport mechanism culminated in stable heterogeneous equilibrium states, demonstrating the architecture's capacity to resolve competing advection-diffusion processes inherent to density-mismatched fluid pairs.

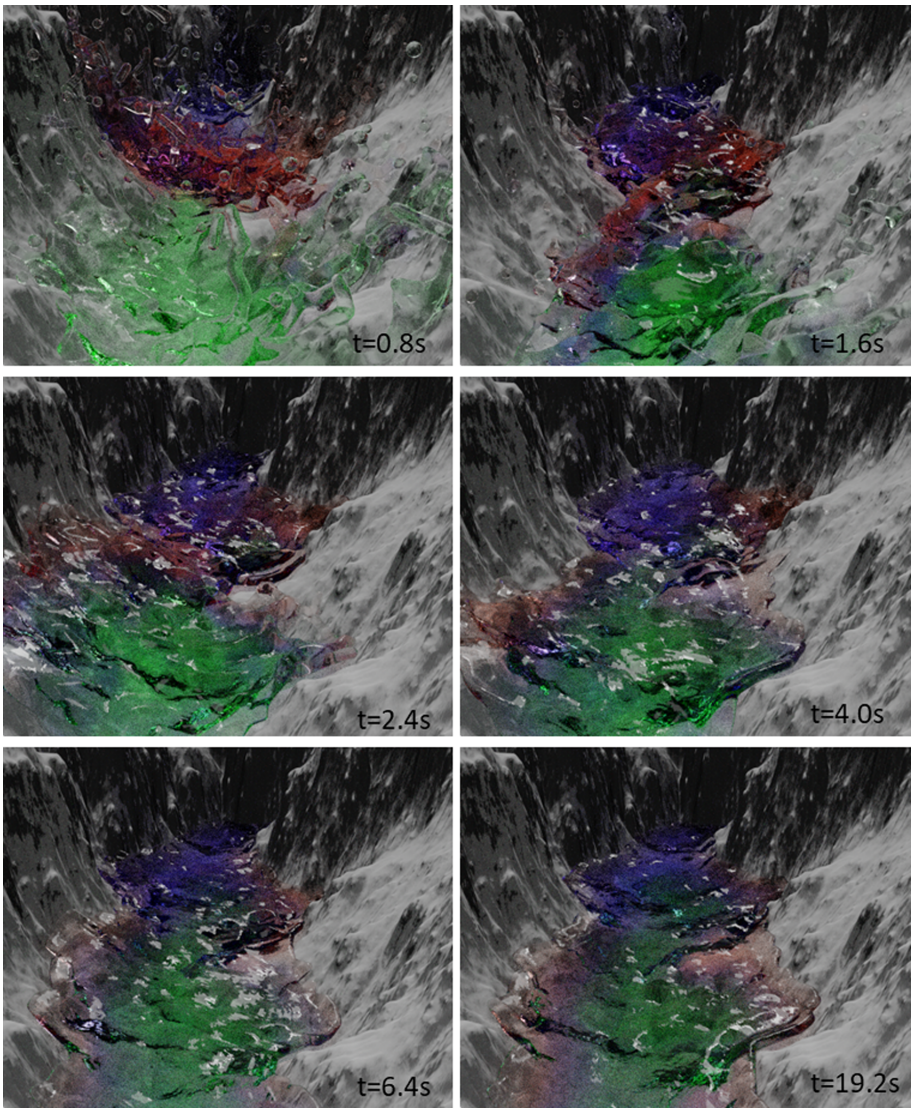

FIGURE 3 | Canyon triphasic flow dynamics.

4.3 | Canyon Triphasic Flow Dynamics

The canyon configuration demonstrates density-mediated flow reorganization across gently sloped synthetic terrain. As depicted in Figure 3, three density-stratified phases (red: ρ_{low}, green: ρ_{high}, blue: ρ_{mid}) evolve under mild gravitational forcing, where terrain geometry rather than steep gradients dominates phase segregation patterns.

The high-density green phase maintains persistent accumulation in basal topographic depressions, forming continuous density currents along primary drainage axes. The mid-density blue fluid exhibits dual transport regimes—achieving partial stratification along intermediate slopes while developing fractal mixing interfaces at elevation transitions. Low-density red phase predominantly invades downwards under the slight force of gravity.

Notably, the mild slope permits prolonged density competition at mesoscale topographic features (slope breaks, micro-depressions). Figure 3 panels reveal cyclic reorganization: (a) Initial gravitational sorting establishes vertical stratification; (b–d) Slope-driven advection induces shear-layer instabilities at phase boundaries; (e,f) Final configuration achieves terrain-adapted quasi-equilibrium with green phase locked in drainage channels, blue forming slope-parallel laminae, and red occupying residual high-elevation pockets.

4.4 | Quantitative Evaluation

Quantitative results in Table 1 demonstrate our method's computational efficiency per simulation step. Our density-conditioned dynamic convolution achieves comparable or slightly better step times compared to the IMM mixture solver baseline [10] across the tested scenarios.

More significantly, beyond per-step efficiency, our approach exhibits substantially improved numerical stability. Baseline methods like IMM Mixture often require very small time steps (e.g., on the order of 1×10^{-4} s to 1×10^{-3} s) to maintain stability in complex multi-phase scenarios, especially those with high density contrasts or intricate interface dynamics. In contrast, our

TABLE 1 | Time consumption per simulation step (seconds).

Scenario	Ours	IMM mixture [10]
Hourglass fluid	0.0026	0.0031
Cocktail	0.0058	0.0064
Canyon	0.0099	0.0112

method consistently runs stably across all presented scenarios (Cocktail, Hourglass, Canyon) using a significantly larger time step of $\Delta t = 0.02$ s.

The ability to use a time step that is potentially one to two orders of magnitude larger translates directly into a substantial reduction in the total number of steps required to simulate a given physical duration, leading to significantly faster overall simulation times. This highlights the robustness inherent in our density-conditioned framework. While direct comparison of accuracy metrics like interface sharpness at identical (small) time steps was not performed in this study, the demonstrated capacity to maintain stability and produce physically plausible visual results (as shown in Figures 1–3) at a large time step of 0.02 s represents a crucial practical advantage for complex multi-phase fluid simulations.

The experimental outcomes collectively verify the framework's novel capacity to intrinsically encode density-dependent material responses within its convolutional operators. By embedding density as a first-class feature in the convolutional operators — rather than treating it as a passive input channel — the system acquires an almost thermodynamic "intuition," distinguishing it from black-box predictors that often conflate inertial and buoyant effects. This is particularly evidenced by the autonomous emergence of counterintuitive transport phenomena in the hourglass configuration and the precise recovery of equilibrium stratification states in multi-phase systems — capabilities that transcend conventional data-driven fluid prediction paradigms through explicit integration of physical priors.

5 | Conclusion

This research exposes inherent limitations of conventional neural operators in density dimension representation, proposing a paradigm shift in multi-phase particle interactions through dynamic kernel functions. The density-conditioned mechanism enables autonomous density gradient resolution, spontaneously generating transport modes obeying vorticity-density coupling laws in non-equilibrium processes like heavy fluid permeation. By transcending the dimensional constraints of fixed feature spaces, this method pioneers continuous prediction across density discontinuities within neural architectures, establishing a new paradigm for first-principle simulation of complex fluid systems.

Despite the promising results, the current method faces limitations, particularly in simulating highly turbulent flows, scenarios with extremely large density ratios beyond those tested, and generalizing efficiently to significantly larger simulation scales without further optimizations.

Future research will extend to miscible multi-phase fluid prediction, focusing on resolving interface ambiguities arising from density-viscosity coupling during phase dissolution. The generalization capability of our density-conditioned mechanism offers new possibilities for such exploration, where establishing implicit correlations between mixing entropy and kernel attenuation rates may enable adaptive prediction from sharp interfaces to gradient mixing states. The fundamental challenge lies in resolving cross-scale coupling between dissolution-diffusion processes and inertial transport, necessitating a paradigm shift in neural operators' spatiotemporal modulation of non-equilibrium flow fields. Further exploration could also include incorporating additional physical effects like surface tension or exploring alternative network architectures for improved scalability and robustness in complex turbulent regimes.

Author Contributions

Yalan Zhang was responsible for conceptualization and writing the main manuscript text. Yuhang Xu handled methodology, formal analysis, and validation. Xiaokun Wang focused on visualization and investigation. Angelos Chatzimparmpas contributed to supervision, manuscript review, and editing. Xiaojuan Ban was involved in funding acquisition, manuscript review, and editing. All authors have read and agreed to the published version of the manuscript.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (No. 62306032), Guangdong Basic and Applied Basic Research Foundation (No. 2022A1515110350), Interdisciplinary Research Project for Young Teachers of USTB (FRF-IDRY-GD24-003). The computing work is partly supported by the MAGICOM Platform of Beijing Advanced Innovation Center for Materials Genome Engineering.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Research data are not shared.

References

1. L. Prantl, B. Ummenhofer, V. Koltun, and N. Thuerey, "Guaranteed Conservation of Momentum for Learning Particle-Based Fluid Dynamics," *Advances in Neural Information Processing Systems* 35 (2022): 6901–6913.

2. B. Ummenhofer, L. Prantl, N. Thuerey, and V. Koltun, "Lagrangian Fluid Simulation With Continuous Convolutions," in *International Conference on Learning Representations* (International Conference on Learning Representations, 2019).

3. A. Colagrossi and M. Landrini, "Numerical Simulation of Interfacial Flows by Smoothed Particle Hydrodynamics," *Journal of Computational Physics* 191, no. 2 (2003): 448–475.

4. X. Y. Hu and N. A. Adams, "A Multi-Phase SPH Method for Macroscopic and Mesoscopic Flows," *Journal of Computational Physics* 213, no. 2 (2006): 844–861.

5. X. Hu and N. A. Adams, "An Incompressible Multi-Phase SPH Method," *Journal of Computational Physics* 227, no. 1 (2007): 264–278.

6. N. Grenier, M. Antuono, A. Colagrossi, D. Le Touzé, and B. Alessandrini, "An Hamiltonian Interface SPH Formulation for

Multi-Fluid and Free Surface Flows," *Journal of Computational Physics* 228, no. 22 (2009): 8380–8393.

7. J. J. Monaghan and A. Rafiee, "A Simple SPH Algorithm for Multi-Fluid Flow With High Density Ratios," *International Journal for Numerical Methods in Fluids* 71, no. 5 (2013): 537–561.

8. Z. Chen, Z. Zong, M. Liu, L. Zou, H. Li, and C. Shu, "An SPH Model for Multiphase Flows With Complex Interfaces and Large Density Differences," *Journal of Computational Physics* 283 (2015): 169–188.

9. A. Krimi, M. Rezoug, S. Khelladi, X. Nogueira, M. Deligant, and L. Ramírez, "Smoothed Particle Hydrodynamics: A Consistent Model for Interfacial Multiphase Fluid Flow Simulations," *Journal of Computational Physics* 358 (2018): 53–87.

10. Y. Xu, X. Wang, J. Wang, et al., "An Implicitly Stable Mixture Model for Dynamic Multi-Fluid Simulations," in *SIGGRAPH Asia 2023 Conference Papers* (Association for Computing Machinery, 2023), 1–11.

11. M. Stanton, B. Humberston, B. Kase, J. F. O'Brien, K. Fatahalian, and A. Treuille, "Self-Refining Games Using Player Analytics," *ACM Transactions on Graphics* 33, no. 4 (2014): 1–9.

12. A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, and P. Battaglia, "Learning to Simulate Complex Physics With Graph Networks," in *International Conference on Machine Learning* (PMLR, 2020), 8459–8468.

13. C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep Learning on Point Sets for 3d Classification and Segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Institute of Electrical and Electronics Engineers, 2017), 652–660.

14. C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," *Advances in Neural Information Processing Systems* 30 (2017): 30.

15. A. Kashefi, D. Rempe, and L. J. Guibas, "A Point-Cloud Deep Learning Framework for Prediction of Fluid Flow Fields on Irregular Geometries," *Physics of Fluids* 33, no. 2 (2021): 027103.

16. Z. Jiang, F. Gao, R. Gu, J. Xu, G. Xu, and T. Rabczuk, "Resistance-Time Co-Modulated Pointnet for Temporal Super-Resolution Simulation of Blood Vessel Flows," arXiv preprint, arXiv:2111.10372 (2021).

17. H. Fan and Y. Yang, "Pointrnn: Point Recurrent Neural Network for Moving Point Cloud Processing," arXiv preprint, arXiv:1910.08287 (2019).

18. C. Schenck and D. Fox, "Perceiving and Reasoning About Liquids Using Fully Convolutional Networks," *International Journal of Robotics Research* 37, no. 4–5 (2018): 452–471.

19. T. P. Miyanawala and R. K. Jaiman, "An Efficient Deep Learning Technique for the Navier-Stokes Equations: Application to Unsteady Wake Flow Dynamics," arXiv preprint, arXiv:1710.09099 (2017).

20. J. Bender and D. Koschier, "Divergence-Free Smoothed Particle Hydrodynamics," in *Proceedings of the 14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Association for Computing Machinery, 2015), 147–155.

21. H. Wessels, C. Weißenfels, and P. Wriggers, "The Neural Particle Method–an Updated Lagrangian Physics Informed Neural Network for Computational Fluid Dynamics," *Computer Methods in Applied Mechanics and Engineering* 368 (2020): 113127.

22. J. Liu, Y. Chen, B. Ni, W. Ren, Z. Yu, and X. Huang, "Fast Fluid Simulation via Dynamic Multi-Scale Gridding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37 (Association for the Advancement of Artificial Intelligence Press, 2023), 1675–1682.

23. R. Winchenbach and N. Thuerey, "Symmetric Basis Convolutions for Learning Lagrangian Fluid Mechanics," arXiv preprint, arXiv:2403.16680 (2024).

24. I. Schuster, M. Mollenhauer, S. Klus, and K. Muandet, "Kernel Conditional Density Operators," in *International Conference on Artificial Intelligence and Statistics* (PMLR, 2020), 993–1004.

Supporting Information

Additional supporting information can be found online in the Supporting Information section. **Data S1.** Supplementary Information.