Coupled Reconstruction of 2D Blood Flow and Vessel Geometry from Noisy Images via Physics-Informed Neural Networks and Quasi-Conformal Mapping*

Han Zhang †, Xue-Cheng Tai ‡, Jean-Michel Morel §, and Raymond H. Chan ¶

Abstract. Blood flow imaging provides important information for hemodynamic behavior within the vascular system and plays an essential role in medical diagnosis and treatment planning. However, obtaining high-quality flow images remains a significant challenge. In this work, we address the problem of denoising flow images that may suffer from artifacts due to short acquisition times or device-induced errors. We formulate this task as an optimization problem, where the objective is to minimize the discrepancy between the modeled velocity field, constrained to satisfy the Navier-Stokes equations, and the observed noisy velocity data. To solve this problem, we decompose it into two subproblems: a fluid subproblem and a geometry subproblem. The fluid subproblem leverages a Physics-Informed Neural Network to reconstruct the velocity field from noisy observations, assuming a fixed domain. The geometry subproblem aims to infer the underlying flow region by optimizing a quasi-conformal mapping that deforms a reference domain. These two subproblems are solved in an alternating Gauss-Seidel fashion, iteratively refining both the velocity field and the domain. Upon convergence, the framework yields a high-quality reconstruction of the flow image. We validate the proposed method through experiments on synthetic flow data in a converging channel geometry under varying levels of Gaussian noise, and on real-like flow data in an aortic geometry with signal-dependent noise. The results demonstrate the effectiveness and robustness of the approach. Additionally, ablation studies are conducted to assess the influence of key hyperparameters.

Key word. Flow Image Reconstruction, Blood Flow Simulation, Physics-Informed Neural Network, Quasi-Conformal Geometry, Image Denoise, Image Segmentation

AMS subject classifications. 65N21, 65D18, 76D05, 92C50

1. Introduction.

Blood flow imaging plays a fundamental role in characterizing cardiovascular dynamics, offering critical insights into hemodynamic behavior within the vascular system. This technique is indispensable for medical monitoring, cardiovascular disease diagnosis, and surgical planning. As a non-invasive approach, it enables safe assessment of patients with severe cardiovascular conditions while avoiding the flow disturbances caused by intravascular instrumentation [41].

However, achieving accurate and high-resolution blood flow imaging remains a significant challenge. Existing modalities, such as flow MRI, often require long acquisition times [26] to produce clear and high-quality flow images. This not only causes discomfort for patients but

*The work of Xue-Cheng Tai is partially supported by NORCE Kompetanseoppbygging program. The work of Jean-Michel Morel is partially supported by RGC-GRF project 11309925, Mathematical Formalization of GIS. The work of Raymond H. Chan is partially supported by HKRGC Grants CityU11309922, LU13300125, ITF Grant No. MHP/054/22, and LU BGR105824. Thanks to the InnoHK project at Hong Kong Centre for Cerebro-cardiovascular Health Engineering (COCHE).

†Department of Mathematics, City University of Hong Kong and Hong Kong Centre for Cerebro-Cardiovascular Health Engineering, Hong Kong, China. (hzhang863-c@my.cityu.edu.hk)

‡Norwegian Research Centre (NORCE), Nygardsgaten 112, 5008 Bergen, Norway. (xtai@norceresearch.no)

§Department of Mathematics, City University of Hong Kong, Hong Kong, China. (jeamorel@cityu.edu.hk)

¶Department of Operations and Risk Management and the School of Data Science, Lingnan University and Hong Kong Centre for Cerebro-Cardiovascular Health Engineering, Hong Kong, China. (raymond.chan@ln.edu.hk)

also makes the imaging process highly susceptible to motion artifacts, such as those caused by respiration. On the other hand, fast acquisition approaches such as low-dose flow MRI or Doppler ultrasound offer faster imaging but typically suffer from high noise levels and signal loss, resulting in data that may be unreliable for precise diagnostic purposes.

In this work, we focus on the problem of reconstructing flow images that are often corrupted by artifacts resulting from short acquisition times or device-induced noise. This task has been modeled by [21, 22] as an optimization problem, by minimizing the discrepancy between a modeled velocity field, constrained by the incompressible Navier-Stokes equations, and the noisy observations.

To more effectively tackle this problem, we propose a reformulation that decomposes into two coupled subproblems: a fluid subproblem and a geometry subproblem. In the fluid subproblem, the flow dynamics are reconstructed within a fixed domain by minimizing the discrepancy between the modeled and measured velocity fields, while ensuring physical consistency through the incompressible Navier-Stokes constraints. In the geometry subproblem, a correction mapping, interpretable as a domain gradient, is estimated to register the reconstructed velocity to the observed data, thereby capturing domain misalignments and approximating the gradient with respect to the spatial domain. By alternating between the two subproblems in a Gauss-Seidel fashion, the overall optimization proceeds iteratively until convergence, which then solves the original problem with theoretical support.

To efficiently solve this coupled optimization problem while leveraging the advantages of differentiable architectures, we design a new algorithm by making the entire framework to be network-based. For the fluid subproblem, a novel Physics-Informed Neural Network (PINN) is used to reconstruct the velocity field, allowing the Navier-Stokes constraints to be enforced directly through backpropagation. For the geometry subproblem, a U-Net architecture is adopted to predict the quasi-conformal mapping, which iteratively corrects geometric misalignments by minimizing the discrepancy between the measured and reconstructed velocity fields through a correction mapping.

We validate our method through comprehensive experiments. First, we evaluate performance on synthetic flow images in a converging channel geometry under both low and high Gaussian noise levels. Next, we test the approach on more anatomically realistic aorta geometries using a signal-dependent noise model. In all cases, our method produces more accurate and higher-quality reconstructions compared to conventional baselines. We also performed ablation studies to assess the impact of different weighting parameters in the optimization of the fluid and geometry subproblems, identifying effective hyperparameter choices. Note that our work is presented and implemented in the 2D setting; however, it can be potentially extended to 3D by replacing the Beltrami coefficient with 3D conformality measures in [43].

Overall, the contribution of this task could be summarized as follows.

- We propose a novel framework for jointly reconstructing both the blood flow field and the flow domain geometry from noisy velocity images, by integrating the Navier-Stokes model with quasi-conformal mapping theory.
- We reformulate the inverse Navier-Stokes problem into two decoupled subproblems: a fluid subproblem and a geometry subproblem. Theoretical analysis is provided to justify the validity of this decomposition.
- We develop a method that utilizes Physics-Informed Neural Networks to reconstruct

the velocity field by fitting to noisy measurement data and enforcing the Navier-Stokes equations as soft physical constraints.

- We introduce a mapping approach for segmenting the flow region using quasi-conformal geometry, which enables globally guided domain evolvation and leveraging morphological priors from the template.

2. Related Work.

2.1. Blood Simulation. Accurate blood flow measurement is vital for cardiovascular healthcare, informing diagnosis, surgical planning, and treatment. Although invasive methods such as Fractional Flow Reserve (FFR) [31] are widely used, they pose risks such as complications and can disrupt the dynamics of natural flow [41], leading to measurement inaccuracies. Numerical simulations offer a noninvasive alternative by modeling blood as an incompressible Newtonian fluid governed by the Navier-Stokes equations [30]. To account for vessel deformation, these models are extended to fluid-structure interaction (FSI) frameworks [32].

Methods have been proposed to solve FSI problems. These include dimensional coupling [13], parallel preconditioning [11], and splitting schemes with stability guarantees [7]. Other approaches involve space-time formulations [6], discontinuous Galerkin methods [39], coupled momentum methods [12], and learning-based approach [44, 48].

2.2. Image Denoise. Image restoration is challenged by noise types such as speckle, Poisson, and Rician noise, which degrade quality and complicate analysis. To address these, a variety of denoising methods have been developed. Lou et al. [25] proposed a weighted difference between anisotropic and isotropic total variation (TV) regularization. Li et al. [24] introduced a variational model for multiplicative noise removal based on the difference of convex functions [35] combined with primal-dual algorithms. Xiao et al. [40] further improved impulsive noise removal by incorporating adaptive proximal parameters into a nonconvex TV-log model. To reduce dependence on initialization, Getreuer et al. [16] developed the convex Getreuer-Tong-Vese (GTV) model for MAP-Rician noise, efficiently solved using Bregman splitting [17]. Chen et al. [9] proposed the strictly convex CZ model, which adds a quadratic regularization term for stability.

Beyond these convex approaches, more advanced techniques employing higher-order and nonconvex regularization have also been explored. Kang et al. [18] proposed a spatially adaptive nonconvex prior to better preserve fine structures, and Martin et al. [27] demonstrated effective Rician noise removal using nonconvex fidelity terms in conjunction with proximal point algorithms. These developments reflect the ongoing effort to improve robustness, structural preservation, and reconstruction accuracy in challenging noise environments.

2.3. Physics-Based Reconstruction. The reconstruction of flow images [10], often corrupted by artifacts from short acquisition times or device noise, has been addressed using physics-based approaches. Kontogiannis *et al.* [21] formulated the inverse Navier-Stokes problem, which optimizes over the fluid dynamics and the computation domain by minimizing the discrepancy between modeled velocity fields and observations under incompressible Navier-Stokes constraints. This was extended to 3D flows in [22], introducing a stabilized Nitsche cut-cell FEM for high-Reynolds-number flows, an implicit geometry representation

with a viscous Eikonal constraint.

In parallel, Aguayo *et al.* [3, 4, 2] developed methods for modeling obstacles in incompressible flows using fictitious domains, penalization potentials, and Brinkman-type permeability terms for distributed-resistance treatment. For problems with known geometries, physics-informed neural networks have been applied to reconstruct flow fields: Sun *et al.* [34] proposed a physics-constrained neural network to reconstruct flow fields from sparse and noisy measurements, combining probabilistic modeling of physical constraints with data uncertainty to guide the network toward physically consistent solutions. Gao *et al.* [15] developed a CNN-based framework for super-resolution and denoising of fluid flows, using physics-informed losses to reconstruct high-resolution fields from coarse or noisy observations without high-resolution labels. Wang *et al.* [37] employed physics-informed neural networks regularized by the Navier-Stokes equations for dense flow reconstruction, but their method does not account for boundary conditions or domain estimation during the reconstruction process.

The problem investigated in this study is inspired by the work of [21], but our approach introduces several key differences. First, we reformulate the original problem into two alternating solved coupled subproblems explicitly. Such a reformulation differs from previous approaches and is supported by a dedicated theoretical result (Theorem 3.3), ensuring that it is mathematically well founded. Second, we solve the fluid subproblem in a new way by using a physics-informed neural network (PINN), which removes the need for explicit numerical discretization, making the approach meshless and allowing the integration of modern deep learning techniques. Third, the geometry subproblem is newly formulated as a registration task, where the domain evolution is represented by a learned deformation map. Fourth, we incorporate quasi-conformal (QC) theory into the geometric component to ensure topology preservation and naturally embed morphological priors in the reconstruction process.

Notation	Description	Notation	Description
$\boldsymbol{x}$	Spatial point	c	Correction mapping
$\boldsymbol{u}$	Modeled flow velocity	f	Deformation mapping
p	Kinetic pressure	Ω_0	Reference flow domain
ν	Kinematic viscosity	Ω	Flow domain
$\boldsymbol{n}$	Outward normal vector	Γ^{i}	Inlet boundary
$\boldsymbol{g}$	Inlet boundary condition	Γ^{o}	Outlet boundary
$\tilde{\boldsymbol{u}}$	Measured flow velocity	Γ^{w}	Wall boundary

Table 1
Notation table for problem formulation.

3. Inverse Navier-Stokes problem.

The main objective of this work is to reconstruct the velocity field over the vessel lumen region, given only a measured blood flow image. To do so, the model should not only try to extract information from the measured data but also try to satisfy its fluid physics, which is generally described by the incompressible Navier-Stokes equation [32]. Before presenting the model, we first describe the notation that is needed for the discussion. Let D denote the full image domain, and let $\tilde{\boldsymbol{u}}$ represent the noisy image obtained from measurements. We describe the lumen domain of the blood vessel as $\Omega \subset D$.

Within this domain Ω. The boundary of the vessel domain $\partial\Omega$ is divided into three parts: the inlet Γ^{i}, the outlet Γ^{o}, and the wall of the vessel Γ^{w}, satisfying $\partial\Omega = \Gamma^{\mathrm{i}} \cup \Gamma^{\mathrm{o}} \cup \Gamma^{\mathrm{w}}$.

In this inverse problem setting, only the noisy velocity field $\tilde{\boldsymbol{u}}$ is available from the imaging data, while the underlying flow domain Ω, the inlet boundary condition $\boldsymbol{g} \in C(\Gamma^{\mathrm{i}})$, and the true velocity field $\boldsymbol{u}$ are unknown. The goal is to reconstruct Ω, $\boldsymbol{g}$, and $\boldsymbol{u}$ such that the modeled velocity field is consistent with the observed data. To quantify the data misfit, we define a discrepancy functional that measures the L^2 distance between the measured velocity and the simulated one as

$$\mathbf{L}_{\mathrm{data}}(\boldsymbol{u}) = \|\tilde{\boldsymbol{u}} - \boldsymbol{u}\|^2_{L^2(D)}, \tag{3.1}$$

The modeled velocity field is required to satisfy the steady incompressible Navier–Stokes equations in the domain Ω. Specifically, the velocity $\boldsymbol{u}$ associated with a pressure p satisfies the system

$$\begin{cases} \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \nabla p + \nu \Delta \boldsymbol{u} = \boldsymbol{0} & \text{in } \Omega, \\ \nabla \cdot \boldsymbol{u} = 0 & \text{in } \Omega, \\ \boldsymbol{u} = \boldsymbol{g} & \text{on } \Gamma^{\mathrm{i}}, \\ (-p + \nu \nabla \boldsymbol{u}) \cdot \boldsymbol{n} = \boldsymbol{0} & \text{on } \Gamma^{\mathrm{o}}, \\ \boldsymbol{u} = \boldsymbol{0} & \text{on } \Gamma^{\mathrm{w}}. \end{cases} \tag{3.2}$$

for a domain $\Omega \subset D$, where ν denotes the kinematic viscosity and $\boldsymbol{n}$ is the outward normal vector on the boundary. With this forward model (3.2), we could define a solution operator $\mathcal{U}_\Omega$ that maps the inlet condition $\boldsymbol{g}$ and the flow domain Ω to the corresponding velocity field $\boldsymbol{u}$, that is, $\mathcal{U}_\Omega : (\boldsymbol{g}, \Omega) \mapsto \boldsymbol{u}$ [32].

With these settings, we pose the inverse reconstruction problem as a PDE-constrained optimization problem. Specifically, our objective is to recover the domain Ω and the inlet boundary condition $\boldsymbol{g}$ by minimizing the discrepancy functional subject to the Navier–Stokes equations, that is, we solve

$$\min_{\boldsymbol{g}, \Omega} \quad \mathbf{L}_{\mathrm{data}}(\boldsymbol{u} = \mathcal{U}_\Omega[\boldsymbol{g}, \Omega]). \tag{3.3}$$

However, optimizing the flow domain Ω is not straightforward and a specific representation is required. Previous works have employed level-set methods [21] and phase-field approaches [1]. In our approach, we model the domain using a quasi-conformal mapping, allowing for both topological control of the evolving process and seamless integration into a network-based optimization framework.

For the quasi-conformal mapping that will be introduced later, we need to use a reference domain Ω_0 that has the same topology as the target domain Ω. The actual flow domain Ω is then expressed by a quasi-conformal mapping f as

$$\Omega = f(\Omega_0). \tag{3.4}$$

As a result, with $\partial\Omega_0 = \Gamma_0^{\mathrm{i}} \cup \Gamma_0^{\mathrm{o}} \cup \Gamma_0^{\mathrm{w}}$, the boundaries of the deformed domain are expressed in terms of the reference boundaries as

$$\Gamma^{\mathrm{i}} = f(\Gamma_0^{\mathrm{i}}), \quad \Gamma^{\mathrm{o}} = f(\Gamma_0^{\mathrm{o}}), \quad \Gamma^{\mathrm{w}} = f(\Gamma_0^{\mathrm{w}}), \tag{3.5}$$

Consequently, we redefine the solution operator to account for the deformation, denoted by $\mathcal{U}_f : (\boldsymbol{g}, f) \mapsto \boldsymbol{u}$, mapping a boundary condition $\boldsymbol{g}$ and a quasi-conformal mapping function f to a velocity field $\boldsymbol{u}$ satisfying (3.2), (3.4) and (3.5). The data-fidelity-driven optimization problem is then reformulated as

$$\min_{\boldsymbol{g},f} \quad \mathbf{L}_{\text{data}}(\boldsymbol{u} = \mathcal{U}_f\,[\boldsymbol{g}, f]), \tag{3.6}$$

subject to the Navier–Stokes equations

$$\begin{cases} \boldsymbol{u}\cdot\nabla\boldsymbol{u} - \nabla p + \nu\Delta\boldsymbol{u} = \mathbf{0} & \text{in } f(\Omega_0), \\ \nabla\cdot\boldsymbol{u} = 0 & \text{in } f(\Omega_0), \\ \boldsymbol{u} = \boldsymbol{g} & \text{on } f(\Gamma_0^{\text{i}}), \\ (-p + \nu\nabla\boldsymbol{u})\cdot\boldsymbol{n} = \mathbf{0} & \text{on } f(\Gamma_0^{\text{o}}), \\ \boldsymbol{u} = \mathbf{0} & \text{on } f(\Gamma_0^{\text{w}}). \end{cases} \tag{3.7}$$

Despite this reformulation, the joint optimization over multiple coupled variables remains challenging, especially given the dependency on a deforming spatial domain. To mitigate this complexity, we adopt a Gauss–Seidel-style alternating optimization strategy. The optimization is split into two subproblems. The first, termed the fluid subproblem, fixes the current domain and solves for the velocity field $\boldsymbol{u}$, pressure p, and the inlet condition $\boldsymbol{g}$. The second, referred to as the geometry subproblem, holds $\boldsymbol{u}$ and $\boldsymbol{g}$ fixed while optimizing the deformation mapping f, thus updating the flow domain.

3.1. Fluid Problem. For the fluid subproblem, we assume that the domain Ω is fixed by a known mapping f, i.e., $\Omega = f(\Omega_0)$. With the geometry held constant, it becomes more tractable to minimize the discrepancy between the observed velocity field $\tilde{\boldsymbol{u}}$ and the modeled solution $\boldsymbol{u}$, which satisfies the Navier–Stokes equations on the fixed domain. In this setting, we redefine the solution operator $\mathcal{U}_g\colon \boldsymbol{g} \mapsto \boldsymbol{u}$ that maps a given inlet boundary condition $\boldsymbol{g}$ to the corresponding velocity field $\boldsymbol{u}$, while implicitly incorporating the fixed mapping f. The resulting optimization problem becomes a partial minimization over $\boldsymbol{g}$ with $\boldsymbol{u} = \mathcal{U}_g\,[\boldsymbol{g}]$, formulated as

$$\min_{\boldsymbol{g}} \quad \mathbf{L}_{\text{data}}(\boldsymbol{u} = \mathcal{U}_g\,[\boldsymbol{g}]). \tag{3.8}$$

Here, the deformation map f is fixed for this minimization problem, reducing the minimization to a problem that only solves for the fluid dynamics.

3.2. Geometry Problem. To model the geometry subproblem, which aims to improve the accuracy of domain estimation, it is essential to derive information that guides the optimization of the deformation mapping. Note that although the observed velocity field is corrupted by noise, it still reflects the flow in the true physical domain. Specifically, the reconstructed flow field that satisfies (3.7) represents the template domain, as it is defined over the region $f(\Omega_0)$. Aligning this reconstructed field with the measured one thus drives the domain evolution by deforming the template domain Ω_0. Conceptually, such a registration map serves as an approximation of the domain gradient, thereby guiding the deformation toward a more accurate domain representation.

To this end, we seek a correction mapping c such that the composed domain $c^{-1} \circ f(\Omega_0)$ more closely approximates the true flow domain. By extending the modeled velocity field $\boldsymbol{u}$ with zero values in the region $D \setminus \Omega$, the refinement is achieved through a registration process that aligns $\boldsymbol{u}$ with the observed data $\tilde{\boldsymbol{u}}$ through a registration procedure by minimizing

$$\mathbf{L}_{\text{reg}} = \|\tilde{\boldsymbol{u}} - \boldsymbol{u} \circ c\|^2_{L^2(D)}\,, \tag{3.9}$$

where we observe that when the correction mapping c is an identity mapping, the optimization functional $\|\tilde{\boldsymbol{u}} - \boldsymbol{u} \circ c\|_{L^2(D)}$ reduces to the original data fidelity functional in $\|\tilde{\boldsymbol{u}} - \boldsymbol{u}\|_{L^2(D)}$. This reflects the ideal scenario in which no further geometric correction is necessary, indicating that the current domain accurately captures the true vascular region where the flow occurs. The theoretical validation of this domain evolution formulation, that it admits a solution minimizing the original objective (3.6) subject to the Navier–Stokes constraints in (3.7), will be presented later in this section.

Both the correction mapping and the deformation mapping are modeled as quasi-conformal mappings that are diffeomorphic. This design is essential to preserve the topology of the reference domain, ensuring that $\Omega = f(\Omega_0)$ retains the same topological structure as Ω_0. This property allows the model to incorporate morphological priors effectively. Furthermore, this formulation is central to the theoretical analysis presented later. We give some detailed explanations of the Beltrami mapping we shall use for the deformation in the following.

Definition 3.1 (Quasi-conformal map). *A quasi-conformal map is a map $f : \mathbb{C} \to \mathbb{C}$ that satisfies the Beltrami equation*

$$\frac{\partial f}{\partial \bar{z}} = \mu(z)\frac{\partial f}{\partial z} \tag{3.10}$$

for some complex-valued function called Beltrami coefficient μ satisfying $\|\mu\|_\infty < 1$ and such that $\frac{\partial f}{\partial z}$ is non-vanishing almost everywhere. The complex partial derivatives are given by

$$\frac{\partial f}{\partial z} := \frac{1}{2}\left(\frac{\partial f}{\partial x} - i\frac{\partial f}{\partial y}\right) \quad \text{and} \quad \frac{\partial f}{\partial \bar{z}} := \frac{1}{2}\left(\frac{\partial f}{\partial x} + i\frac{\partial f}{\partial y}\right).$$

The Beltrami coefficient μ quantifies the deviation from conformality. If $\mu = 0$ at a point p, the map is conformal in a neighborhood around p, reducing (3.10) to the Cauchy–Riemann equations. In this case, f near p can be written as

$$f(z) = f(p) + f_z(p)(z + \mu(p)\bar{z}),$$

where $f(p)$ is a translation and $f_z(p)$ a conformal scaling. Thus, μ captures the non-conformal part of f, and f is conformal if and only if $\mu \equiv 0$.

Using a quasi-conformal mapping to represent the deformations of the domain, the geometry subproblem can be finally formulated with respect to a correction mapping c as

$$\min_{\boldsymbol{c}} \quad \mathbf{L}_{\text{reg}} = \|\tilde{\boldsymbol{u}} - \boldsymbol{u} \circ c\|^2_{L^2(D)}, \quad \text{s.t.} \quad \mu(c) < 1. \tag{3.11}$$

In our formulation, the quasi-conformal map c effectively serves as a domain gradient, producing a smooth topology-preserving deformation that directly drives boundary evolution. This ensures that the reconstructed geometry keeps the same topology as the template — an important advantage in cardiovascular applications, where vessel networks have consistent anatomical structures and the template provides a meaningful morphological prior. Even in rare cases with atypical anatomy, a coarse template can still be extracted (e.g., from a rough segmentation) and subsequently refined via flow-driven registration. Such topology preservation cannot be achieved by phase-based approaches, and level-set methods lack a theoretical foundation for controlling it. Moreover, because the deformation is obtained by minimizing a global registration energy over the entire flow field, the resulting geometry evolution is smooth, stable, and globally coherent. This stands in contrast to the level-set approach, whose evolution is fundamentally local, driven by contour curvature and neighborhood features, making it difficult to enforce global structure and long-range consistency during optimization.

3.3. Subproblem Formulations Verification. After decomposing the original problem into the fluid and geometry subproblems, we now describe the alternating procedure used to iteratively solve these subproblems in order to approximate the solution of the original problem.

Let n denote the iteration number. Given a f_n, we first solve for iteration n by minimizing the data discrepancy objective (3.8) subject to the Navier–Stokes constraints (3.7) over the domain $f_n(\Omega_0)$, where the first mapping f_1 is initialized (identity mapping in this work). With the obtained u_n, p_n, g_n from the fluid subproblem (3.8), we then solve the geometry subproblem 3.9 for the correction map to get its minimizer c_n. This map updates the domain for the next iteration via $f_{n+1} = c_n^{-1} \circ f_n$. The process of alternatively solving u_n, p_n, g_n and c_n then repeats until convergence is met. We also refer to Figure 1 for the pipeline flow.

This alternating scheme captures the interplay between fluid physics and geometric deformation: The reconstructed velocity field informs the domain correction, which in turn influences the next fluid solution. As the iterations progress, both the flow field u_n and the mapping f_n are expected to converge to some limits that align with the measured data, the governing physics, and anatomical constraints.

To support this methodology, we establish a theory that, under proper assumptions, the converging sequence generated by this alternating procedure admits a partial and local minimizer of the original problem. The full proof is provided in the appendix for completeness.

Definition 3.2. *Let $F(x, y)$ be a function defined on a product space $\mathcal{X} \times \mathcal{Y}$. The partial minimization of F with respect to the variable y, for a fixed $x \in \mathcal{X}$, is defined as*

$$\min_{y \in \mathcal{Y}} F(x, y).$$

Theorem 3.3. *Let $\mathcal{H}(D)$ be the group of diffeomorphisms on a compact domain D. Let Ω_0 be a domain with smooth boundary and $\tilde{u}$ measured data in D. Let $\{(u_n, p_n, f_n, g_n, c_n)\}_{n=1}^{\infty}$ be a sequence of solutions generated by an iterative inverse Navier–Stokes reconstruction algorithm, which means*

- *$f_1 \in \mathcal{H}(D)$ is given,*
- *(u_n, p_n, g_n) minimize $\|\tilde{u} - u\|^2_{L^2(D)}$ with fixed $f_n \in \mathcal{H}(D)$ for (u_n, p_n, g_n, f_n) satisfying (3.7),*

- $f_{n+1} = c_n^{-1} \circ f_n$, *where* $c_n = \arg\min_{c \in \mathcal{H}(D)} \|\tilde{u} - u_n \circ c\|_{L^2(D)}$.

Assume the following:

(1) *The velocity fields converge:* $u_n \to u_*$ *in* $L^\infty(D)$,

(2) *The domain mappings converge:* $f_n \to f_*$ *in* $L^\infty(D)$,

(3) *In a neighborhood of* f_* *in* $L^\infty(D)$, *minimizing* $\|\tilde{u} - u\|^2_{L^2(D)}$ *under* (3.7) *admits a unique solution* (u_n, p_n, g_n) *and* $u \in W^{1,\infty}(D)$. *The solution operator* $\mathcal{U} : f \mapsto u$ *is continuous for the* $L^\infty(D)$ *norm.*

(4) *For large* n, *the solution* u_n *is uniformly Lipschitz.*

Then, the following conclusions hold:

(*a*) *The correction mappings converge:* $c_n \to \boldsymbol{id}$ *in* $L^\infty(D)$,

(*b*) *The limit* f_* *is the partial and local minimizer under* (3.7) *for* $f \in \mathcal{H}(D)$,

(*c*) *The limit* u_* *is the partial and local minimizer under* (3.7) *for* $f \in \mathcal{H}(D)$,

Note that assumptions (1)-(2) stem from empirical observations on our numerical scheme. For all cases presented in this paper, the iterative scheme converges in a manner consistent with these assumptions. The uniqueness and the regularity of the solution to the Navier-Stokes equations (i.e., equation (3.7) with an appropriate choice of boundary data $\boldsymbol{g}$) are established in [33, Theorem 3.7.3] and [19, Theorem 7.1]. Therefore, given proper smoothness of Ω_0 and f_n, assuming the uniqueness of the associated minimization problem and the uniform Lipschitz continuity of the solution sequence is reasonable and consistent with the well-posedness of the PDE constraint.

4. Numerical Method.

In the following sections, we provide specific implementation details, including the point sampling strategy within the region and along the boundaries defined by the template mask, the formulation of the fluid subproblem, and the geometry modeling. Finally, we present the complete algorithm.

4.1. Mask Sampling for Flow Domain.

Since a binary mask is used to represent the flow region where blood is expected to circulate, special consideration is required when sampling points for PINN training. We now describe our sampling strategy for the fluid domain Ω and its boundary $\partial\Omega$ at iteration n. Denoting the deformation mapping at this iteration by f, the domain is given by $\Omega = f(\Omega_0)$. Similarly, as in (3.5), the boundary components satisfy

$$\Gamma^{\mathrm{i}} = f(\Gamma_0^{\mathrm{i}}), \quad \Gamma^{\mathrm{o}} = f(\Gamma_0^{\mathrm{o}}), \quad \Gamma^{\mathrm{w}} = f(\Gamma_0^{\mathrm{w}}), \tag{4.1}$$

The input data consists of a velocity image $\tilde{\boldsymbol{u}}$ that is defined on $[0, \varepsilon H] \times [0, \varepsilon K]$, where ε is determined by the image resolution. Each pixel point is positioned at coordinates

$$\mathcal{D} := \{(\varepsilon/2 + h \cdot \varepsilon,\ \varepsilon/2 + k \cdot \varepsilon) \ : \text{ where } h = 0, \ldots, H-1 \text{ and } k = 0, \ldots, K-1\}. \tag{4.2}$$

Note that although the value for $\tilde{\boldsymbol{u}}$ is only explicitly defined on the grid points in $\mathcal{D}$, its values at other locations can be obtained through interpolation methods; in this work, bilinear interpolation is used for this purpose.

Let us also denote the mask that is used to represent the domain Ω as M_Ω defined on pixel points $\mathcal{D}$ as

$$\boldsymbol{M}_\Omega = \begin{cases} 1 & \text{for } \boldsymbol{x} \in \mathcal{W}, \\ 0 & \text{for } \boldsymbol{x} \in \mathcal{D} \setminus \mathcal{W}. \end{cases} \tag{4.3}$$

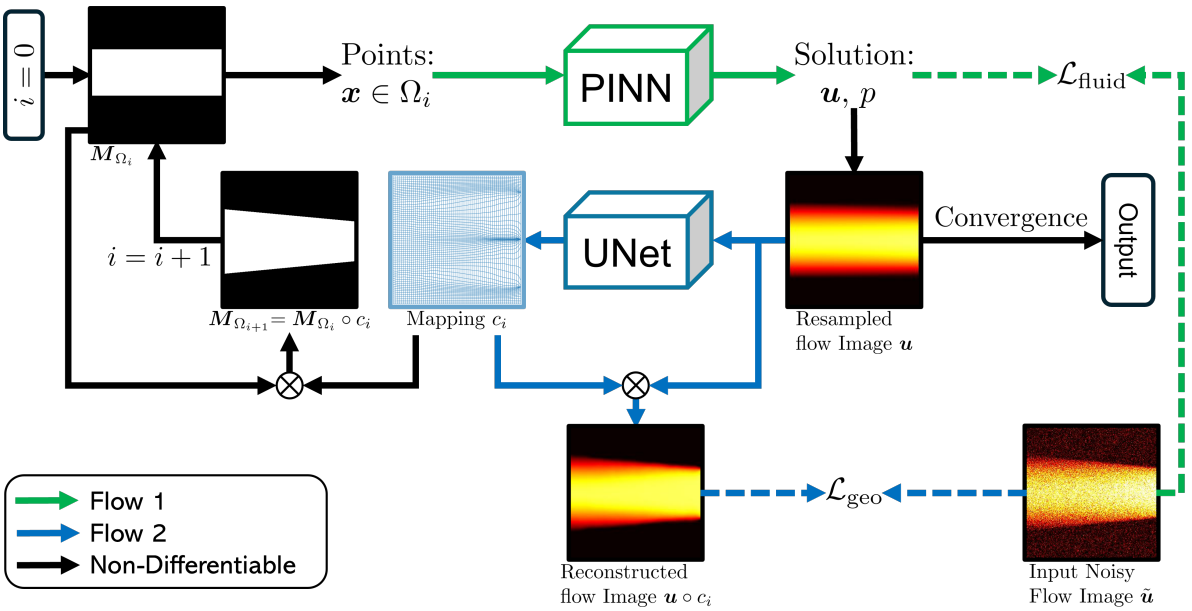

Figure 1. *Overview of the iterative pipeline: starting from an initial mask defining the flow domain, points are sampled and input to a PINN for flow prediction. The predicted flow is resampled over the image domain, followed by quasi-conformal mapping to align with the noisy image. The mask is updated via this mapping, and the process repeats until convergence.*

where $\mathcal{W} = \mathcal{D} \cap \Omega$ are the pixel points within the domain Ω. Using an edge detector [29] on $\boldsymbol{M}_\Omega$, which is enforced to be 4-connected, we could then obtain the edge image as $\boldsymbol{M}_\Gamma$ illustrated in Figure 2.(a). The point set that was used to approximately represent the boundary of Ω is as

$$\mathcal{E} = \{\boldsymbol{x} \in \mathcal{D} : \boldsymbol{M}_\Gamma(\boldsymbol{x}) = 1\} \in \mathcal{G}.$$

The inlet and outlet boundaries are constrained to lie on the edges of the image grid, with coordinates either at $\frac{1}{2}\varepsilon$ or $H\varepsilon - \frac{1}{2}\varepsilon$. In contrast, the vessel wall boundary can be represented by a set of points that are not on the grid boundary as follows

$$\mathcal{E}^{\text{w}} = \{(x_1, x_2) \in \mathcal{E} \mid x_1 \notin \{\tfrac{1}{2}\varepsilon, H\varepsilon - \tfrac{1}{2}\varepsilon\};\ x_2 \notin \{\tfrac{1}{2}\varepsilon, K\varepsilon - \tfrac{1}{2}\varepsilon\}\}, \tag{4.4}$$

and divide $\mathcal{E}^{\text{w}}$ into Z_{w} disjoint points collections as

$$\mathcal{E}^{\text{w}} = \mathcal{E}^{\text{w}}_1 \cup \mathcal{E}^{\text{w}}_2 \cup \ldots \mathcal{E}^{\text{w}}_{Z_{\text{w}}},$$

where the points of each $\mathcal{E}^{\text{w}}_z$ represent a curve in the image that is 4-connected, and each set are mutually disjoint.

Furthermore, we also divide $\mathcal{E} \setminus \mathcal{E}^{\text{w}}$ into multiple disjoint points collection as

$$\begin{aligned} \mathcal{E}^{\text{i}} &= \mathcal{E}^{\text{i}}_1 \cup \mathcal{E}^{\text{i}}_2 \cup \cdots \cup \mathcal{E}^{\text{i}}_{Z_{\text{i}}}, \\ \mathcal{E}^{\text{o}} &= \mathcal{E}^{\text{o}}_1 \cup \mathcal{E}^{\text{o}}_2 \cup \cdots \cup \mathcal{E}^{\text{o}}_{Z_{\text{o}}}, \end{aligned}$$

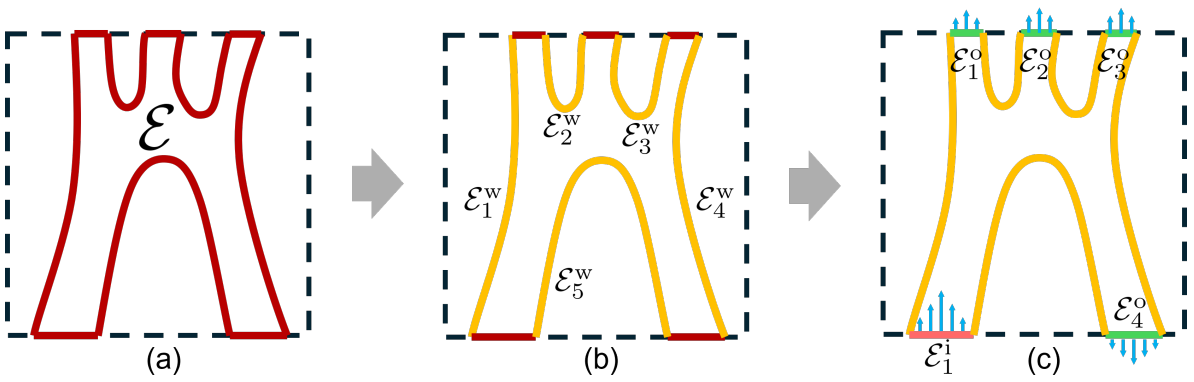

Figure 2. *Illustration of the steps used to divide the entire boundary into wall, outlet, and inlet boundaries. Red indicates the full boundary or the unclassified segments; yellow denotes wall boundaries; green represents outlets; and pink indicates inlets.*

where each of the point sets is 4-connected, and the sets are mutually disjoint. To determine whether a given point collection $\mathcal{E}_z^*$ (where $* \in \{\mathrm{i}, \mathrm{o}\}$ corresponds to an inlet or outlet), we first compute the average velocity over the points in the set as follows:

$$\bar{\boldsymbol{u}}^* = \frac{1}{|\mathcal{E}^*|} \sum_{\boldsymbol{x} \in \mathcal{E}^*} \boldsymbol{u}(\boldsymbol{x}). \tag{4.5}$$

where $|\mathcal{E}_z^*|$ represent the number of points in $\mathcal{E}_z^*$. Then, as in Figure 2.(c), if $\bar{\boldsymbol{u}}^*$ points inward to the image domain, which indicates an inflow pattern, $\mathcal{E}^*$ should be an inlet boundary. Otherwise, the average velocity points outward from the image domain, indicating an outflow pattern, and thus the collection corresponds to an outlet.

Thus, the pixel points for the interior region of Ω is as

$$\mathcal{G} = \mathcal{W} - \mathcal{E}.$$

and with $\mathcal{G}$ and $\mathcal{E}$ in hand, we can then do sampling for the interior domain $\Omega \setminus \partial\Omega$ and the boundary Γ.

To sample a new point in the interior domain $\Omega \setminus \partial\Omega$, we could first randomly choose one point $\boldsymbol{p}$ from $\mathcal{G}$. Then, the sampled point is as

$$\boldsymbol{x}^{\mathrm{g}} = \boldsymbol{p} + [l_1, l_2]', \tag{4.6}$$

where l_1 and l_2 are randomly sampled from $(-\epsilon, \epsilon)$.

To sample points along a segment of $\mathcal{E}_z^{\mathrm{i}}$, $\mathcal{E}_z^{\mathrm{o}}$, and $\mathcal{E}_z^{\mathrm{w}}$, we apply piecewise linear interpolation. Specifically, let $\mathcal{E}^*$ denote one such segment consisting of an ordered set of points $\{\boldsymbol{p}_0, \boldsymbol{p}_1, \ldots, \boldsymbol{p}_{N_*}\}$, where each consecutive pair $(\boldsymbol{p}_j, \boldsymbol{p}_{j+1})$ defines a straight-line segment. Then, we first generate a random value $l \in [0, L]$, where L is the total length of the boundary. We then determine which segment the sampled point will lie in by accumulating in each segment's

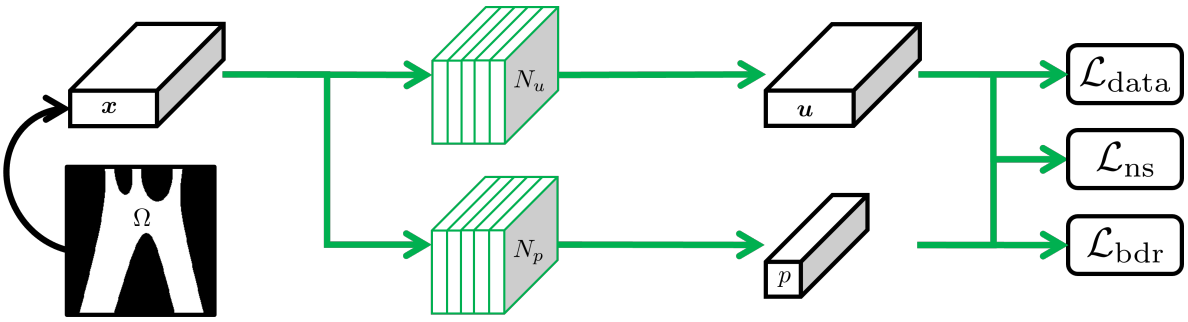

Figure 3. *Architecture of the PINN for fluid flow prediction, featuring two separate networks dedicated to velocity and pressure estimation.*

length S_r until $L_j = \sum_{r=1}^{j} S_r \geq l$ is met, which means the sampled points within segment j as (p_{j-1}, p_j). We finally have the sampled point on the boundary as

$$\boldsymbol{x}^{\mathrm{e}} = \frac{L_j - l}{L_j - L_{j-1}} \boldsymbol{p}_{j-1} + \frac{l - L_{j-1}}{L_j - L_{j-1}} \boldsymbol{p}_j. \tag{4.7}$$

4.2. Fluid Subproblem. We describe some details for optimizing the fluid subproblems (3.8) for each iteration n through Physics-Informed Neural Networks. For convenience, we denote the current spatial domain as Ω, omitting the iteration index. This represents Ω_n at iteration n. We first randomly sample M^{g} points from the inner domain as $\boldsymbol{x}_m^{\mathrm{g}} \in \Omega \setminus \partial\Omega$ according to (4.6) for $m = 1, \ldots, M^{\mathrm{g}}$. Regarding the boundaries, we randomly sample the coordinates M^{i}, M^{o} and M^{w} at the inlet, outlet and wall boundary as $\boldsymbol{x}_m^{\mathrm{i}} \in \Gamma^{\mathrm{i}}$, $\boldsymbol{x}_m^{\mathrm{o}} \in \Gamma^{\mathrm{o}}$ and $\boldsymbol{x}_m^{\mathrm{w}} \in \Gamma^{\mathrm{w}}$ according to (4.7), respectively. We then introduce two networks as given in Figure 3: one for the velocity computation, named $\mathcal{N}_u$, and another for the pressure computation, named $\mathcal{N}_p$. We use $\boldsymbol{\theta}_u$ and $\boldsymbol{\theta}_p$ the trainable parameters for these two neural networks. The two networks map any point $\boldsymbol{x} \in \Omega$ to the values of the velocity and pressure at this point, respectively. Omitting thesubscripts of the sampled points, the flow velocity and pressure for the point $\boldsymbol{x}^{\mathrm{g}}$ can be calculated as

$$\begin{aligned} \boldsymbol{u}^{\mathrm{g}}(\boldsymbol{x}^{\mathrm{g}}; \boldsymbol{\theta}_u) &= \mathcal{N}_u(\boldsymbol{x}^{\mathrm{g}}; \boldsymbol{\theta}_u), \\ p^{\mathrm{g}}(\boldsymbol{x}^{\mathrm{g}}; \boldsymbol{\theta}_p) &= \mathcal{N}_p(\boldsymbol{x}^{\mathrm{g}}; \boldsymbol{\theta}_p), \end{aligned} \tag{4.8}$$

where for the boundaries, the notations are changed correspondingly as $\boldsymbol{u}^{\mathrm{i}}$, $\boldsymbol{u}^{\mathrm{o}}$, $\boldsymbol{u}^{\mathrm{w}}$ and p^{o}. Also, let's define the discretized norm used throughout. Let U denote one of the relevant domains: Ω, Γ^{i}, Γ^{o}, or Γ^{w}. For each domain U, let $\{\boldsymbol{x}_m\}_{m=1}^{M}$ represent the corresponding set of sampled points (indexed appropriately). Given a function $F : U \to \mathbb{R}$, we define the discrete L^2 norm as

$$\|F\|_{L^2(U;M)} := \left(\frac{1}{M} \sum_{m=1}^{M} |F(\boldsymbol{x}_m)|^2 \right)^{1/2} \qquad \text{where } x_m \sim U. \tag{4.9}$$

To train the networks, in addition to the measured data, the governing partial differential equations should also be incorporated into the loss function through their residuals to make the neural networks physics-informed. Since the networks take spatial coordinates as input and predict physical quantities such as velocity and pressure, automatic differentiation (e.g., in PyTorch or TensorFlow) can be used to compute the derivatives required to evaluate these residuals. We then present the details of these residuals, which together constitute the loss function.

As the modeled velocity is extended by zero value over $D \setminus \Omega$, the data fitting term is thus only defined in the domain Ω as

$$\mathcal{L}_{\text{data}}(\boldsymbol{\theta}_u, \boldsymbol{\theta}_p) = \|\tilde{\boldsymbol{u}} - \boldsymbol{u}^{\text{g}}\|^2_{L^2(\Omega; M^{\text{g}})}. \tag{4.10}$$

To make the process physics-informed, we rewrite the Navier-Stokes equation into residual forms

$$\begin{aligned}\mathcal{L}_{\text{ns}}(\boldsymbol{\theta}_u, \boldsymbol{\theta}_p) = &\|\boldsymbol{u}^{\text{g}} \cdot \nabla \boldsymbol{u}^{\text{g}} - \nabla p^{\text{g}} + \nu \Delta \boldsymbol{u}^{\text{g}}\|^2_{L^2(\Omega; M^{\text{g}})} \\ &+ \|\nabla \cdot \boldsymbol{u}^{\text{g}}\|^2_{L^2(\Omega; M^{\text{g}})},\end{aligned} \tag{4.11}$$

and its associated boundary conditions into residual form as

$$\mathcal{L}_{\text{bdr}}(\boldsymbol{\theta}_u, \boldsymbol{\theta}_p) = \|\boldsymbol{u}^{\text{i}} - \boldsymbol{g}^{\text{i}}\|^2_{L^2(\Gamma^{\text{i}}; M^{\text{i}})} + \|(-p^{\text{o}} + \nu \nabla \boldsymbol{u}^{\text{o}}) \cdot \boldsymbol{n}\|^2_{L^2(\Gamma^{\text{o}}; M^{\text{o}})} + \|\boldsymbol{u}^{\text{w}}\|^2_{L^2(\Gamma^{\text{w}}; M^{\text{w}})}. \tag{4.12}$$

We also assume that the inlet velocity boundary follows a parabolic profile. Therefore, the inlet velocity prediction should be regularized to be a parabolic velocity profile whose average velocity matches the average noisy velocity over Γ^{i}, given by

$$\boldsymbol{g}^{\text{r}}(\boldsymbol{x}) = \frac{3}{2} \frac{\int_{\Gamma^{\text{i}}} \tilde{\boldsymbol{u}}}{|\Gamma^{\text{i}}|} \left(1 - \frac{|\boldsymbol{q} - \boldsymbol{x}|^2}{(|\Gamma^{\text{i}}|/2)^2}\right),$$

where $\boldsymbol{q}$ is the centroid of Γ^{i} and $|\Gamma^{\text{i}}|$ represents the length of it. This leads to an additional regularization for the inlet velocity as

$$\mathcal{L}_{\text{in}} = \|\boldsymbol{u}^{\text{i}} - \boldsymbol{g}^{\text{r}}\|^2_{L^2(\Gamma^{\text{i}}; M^{\text{i}})}, \tag{4.13}$$

Integrating the losses from both the flow and deformation problems with proper boundary losses, we obtain

$$\mathcal{L}_{\text{fluid}}(\boldsymbol{\theta}_u, \boldsymbol{\theta}_p) = \alpha_{\text{data}} \mathcal{L}_{\text{data}} + \alpha_{\text{ns}} \mathcal{L}_{\text{ns}} + \alpha_{\text{bdr}} \mathcal{L}_{\text{bdr}} + \alpha_{\text{in}} \mathcal{L}_{\text{in}}, \tag{4.14}$$

where α_{data}, α_{ns}, α_{bdr} and α_{in} are the weights for reconstruction error, the Navier-Stokes equations, the boundary condition and inlet velocity regularization respectively. Then, the fluid optimization problem is modeled as follows,

$$\min_{\boldsymbol{\theta}_u, \boldsymbol{\theta}_p} \mathcal{L}_{\text{fluid}}(\boldsymbol{\theta}_u, \boldsymbol{\theta}_p). \tag{4.15}$$

To reduce the loss values, gradient descent methods are used to update $\boldsymbol{\theta}_u$ and $\boldsymbol{\theta}_p$ with the help of two distinct optimizers by

$$\begin{aligned}\boldsymbol{\theta}_u &\leftarrow \boldsymbol{\theta}_u - \tau_u \nabla_{\boldsymbol{\theta}_u} \mathcal{L}_{\text{fluid}}(\boldsymbol{\theta}_u, \boldsymbol{\theta}_p), \\ \boldsymbol{\theta}_p &\leftarrow \boldsymbol{\theta}_p - \tau_p \nabla_{\boldsymbol{\theta}_p} \mathcal{L}_{\text{fluid}}(\boldsymbol{\theta}_u, \boldsymbol{\theta}_p).\end{aligned} \tag{4.16}$$

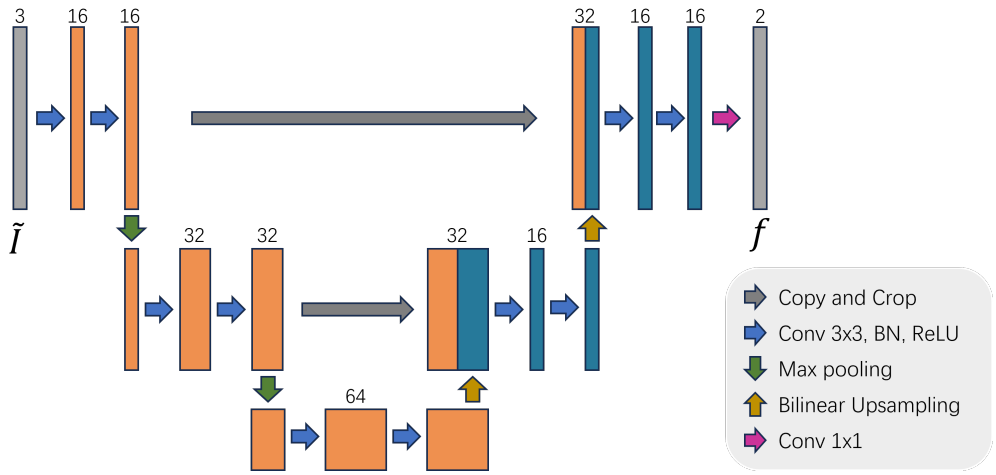

Figure 4. *Architecture of the UNet used for quasi-conformal mapping estimation.*

where $\nabla_{\boldsymbol{\theta}_u}$ and $\nabla_{\boldsymbol{\theta}_p}$ are the gradients with respect to $\boldsymbol{\theta}_u$ and $\boldsymbol{\theta}_p$, which are computed by autogradient within *PyTorch*. Here, τ_u and τ_p represent adaptive step sizes determined by various methods [5, 20], with step sizes computed according to the approach outlined in [20].

After sufficiently training the neural networks $\mathcal{N}_u$ and $\mathcal{N}_p$ in iteration n by minimizing (4.14) with $\Omega = \Omega_n$ in (4.11)–(4.13), we sample the reconstructed flow image over the entire image grid $\mathcal{D}$ (defined in (4.2) for use in the geometry subproblem to be detailed in the next subsection. The sampled velocity field $\hat{\boldsymbol{u}}$ is defined as:

$$\hat{\boldsymbol{u}}(\boldsymbol{x}) = \begin{cases} \boldsymbol{u}^{\mathrm{g}}(\boldsymbol{x}), & \text{if } \boldsymbol{x} \in \mathcal{G}, \\ \boldsymbol{u}^{\mathrm{i}}(\boldsymbol{x}), & \text{if } \boldsymbol{x} \in \mathcal{E}^{\mathrm{i}}, \\ \boldsymbol{u}^{\mathrm{o}}(\boldsymbol{x}), & \text{if } \boldsymbol{x} \in \mathcal{E}^{\mathrm{o}}, \\ \boldsymbol{u}^{\mathrm{w}}(\boldsymbol{x}), & \text{if } \boldsymbol{x} \in \mathcal{E}^{\mathrm{w}}, \\ 0, & \text{otherwise.} \end{cases} \tag{4.17}$$

Here, $\mathcal{G}$, $\mathcal{E}^{\mathrm{i}}$, $\mathcal{E}^{\mathrm{o}}$, and $\mathcal{E}^{\mathrm{w}}$ denote the sets of points in the discretized image grid corresponding to Ω, Γ^{i}, Γ^{o}, and Γ^{w} as described in Section 4.1, respectively.

4.3. Geometry Subproblem. We now address the geometry subproblem for each iteration n. We first obtain the value of solved $\hat{\boldsymbol{u}}$ from the fluid subproblem. Then, we also define the norm over the discretized grid $\mathcal{D}$ for a function $F : \mathcal{D} \to \mathbb{R}$ as

$$\|F\|_{L^2(\mathcal{D})} = \left(\frac{1}{HK} \sum_{\boldsymbol{x} \in \mathcal{D}} |F(\boldsymbol{x})|^2 \right)^{1/2}, \tag{4.18}$$

As discussed in the previous section, we need to register the image $\hat{\boldsymbol{u}}$ obtained from the solution of the fluid subproblem (4.15) to the captured noisy image $\tilde{\boldsymbol{u}}$ to produce a mapping c, which serves as an approximation of the domain gradient. This mapping is generated by a

neural network

$$c = \mathcal{N}_c(\hat{\boldsymbol{u}}; \boldsymbol{\theta}_c) \tag{4.19}$$

where $\boldsymbol{\theta}_c$ is the trainable weights of the network. Following the approach of [46, 47], we use a U-Net architecture for $\mathcal{N}_c$, as illustrated in Figure 4, to estimate the quasi-conformal mapping. To ensure accurate registration, the network is trained by minimizing the following registration loss.

$$\mathcal{L}_{\text{reg}}(\boldsymbol{\theta}_c) = \|\tilde{\boldsymbol{u}} - \hat{\boldsymbol{u}} \circ c\|^2_{L^2(\mathcal{D})}, \tag{4.20}$$

For c to be a quasi-conformal mapping, its associated Beltrami coefficient must be strictly less than 1 according to Definition 3.1. In addition, to promote the mapping c to be diffeomorphic, we apply Laplacian regularization to encourage smoothness and control geometric distortion. Thereby, the regularization terms are introduced as

$$\begin{aligned} \mathcal{L}_{\text{bc}}(\boldsymbol{\theta}_c) &= \|e^{\mu(c)-1}\|^2_{L^2(\mathcal{D})}, \\ \mathcal{L}_{\text{lap}}(\boldsymbol{\theta}_c) &= \|\Delta c\|^2_{L^2(\mathcal{D})}, \end{aligned} \tag{4.21}$$

where $\mu(c)$ represents the Beltrami coefficient of the mapping c, computed through a finite difference implementation of Equation (3.10). The term Δc denotes the Laplacian of the mapping c. Together, these regularizations ensure that c adheres to quasi-conformal constraints while maintaining desirable geometric properties, such as continuity and smoothness.

The overall objective function for the geometry subproblems (3.9) is then replaced by

$$\mathcal{L}_{\text{geo}}(\boldsymbol{\theta}_c) = \alpha_{\text{reg}}\mathcal{L}_{\text{reg}} + \alpha_{\text{bc}}\mathcal{L}_{\text{bc}} + \alpha_{\text{lap}}\mathcal{L}_{\text{lap}}, \tag{4.22}$$

where α_{reg}, α_{bc}, and α_{lap} are the weighting parameters for the registration error, Beltrami regularization, and Laplacian regularization, respectively. Then, the geometry subproblem is reformulated as

$$\min_{\boldsymbol{\theta}_c} \mathcal{L}_{\text{geo}}(\boldsymbol{\theta}_c). \tag{4.23}$$

To reduce the loss values, gradient descent methods are used to update $\boldsymbol{\theta}_c$ by

$$\boldsymbol{\theta}_c \leftarrow \boldsymbol{\theta}_c - \tau_c \nabla_{\boldsymbol{\theta}_c} \mathcal{L}_{\text{geo}}(\boldsymbol{\theta}_c), \tag{4.24}$$

where $\nabla_{\boldsymbol{\theta}_c}$ is the gradient with respect to $\boldsymbol{\theta}_c$, τ_c represents adaptive step sizes as in [5]. After sufficient training, the produced mapping is used to warp the current mask, thereby refining the vessel region. This updated domain is then used in the next iteration of fluid subproblem training.

4.4. Overall Algorithm. The proposed algorithm iteratively alternates between solving the flow subproblem (4.15) and the geometry subproblem (4.23) to achieve accurate blood flow imaging. As a result, the overall optimization problem (3.6), subject to the constraint in (3.7), is approximately solved through this alternating scheme between the fluid subproblem (4.15) and the geometry subproblem (4.23).

We begin with the fluid subproblem using the domain mask M_{Ω_n}, which represents the mask at iteration n, as defined in (4.3). The initial mask M_{Ω_1}, provided at the beginning, serves as both a topological prior and a coarse approximation of the spatial domain in which blood flow is expected to occur. Based on the current mask M_{Ω_n}, spatial points are sampled and used as input to the PINN models $\mathcal{N}_u$ and $\mathcal{N}_p$ in (4.8) used for (4.15). These two networks are then trained on the sampled points by minimizing the fluid loss function defined in (4.14), using the gradient descent described in (4.16). Once sufficiently trained, the models produce a velocity field that adheres to the governing physical laws (3.7) while fitting the noisy observational data $\tilde{u}$.

After training, we evaluate the predicted velocity at the points of the image grid $\mathcal{D}$ (as defined in (4.2)), which produces a reconstructed flow image $\hat{\boldsymbol{u}}_n$, which corresponds to the indexed sampled velocity image at iteration n, denoted by $\hat{\boldsymbol{u}}$ in (4.17). To correct for potential misalignments between the solved domain of this iteration and the actual vessel geometry, we estimate a quasi-conformal registration mapping that aligns $\hat{\boldsymbol{u}}_n$ with the observed flow $\tilde{\boldsymbol{u}}$. This is achieved by training the mapping network $\mathcal{N}_c$ to minimize the registration loss (4.22), using the gradient descent scheme in (4.24). After sufficient training, the network outputs a registration mapping c_n, which is used to update the vessel mask by warping the previous domain: $M_{\Omega_{n+1}} = M_{\Omega_n} \circ c_n$. As shown in Remark 1, this is equivalent to $M_{\Omega_{n+1}} = M_{c_n^{-1}(\Omega_n)}$. The updated mask $M_{\Omega_{n+1}}$ then replaces the previous one and serves as the domain mask for the next iteration, as illustrated in Figure 1, enabling a new round of point sampling for the fluid subproblem training.

This alternation between the PINN-based flow reconstruction problem (4.15) and domain refinement through the quasi-conformal registration problem (4.23) continues iteratively. The algorithm terminates when either a convergence criterion is satisfied or the maximum number of iterations is reached. The overall algorithm is summarized in Algorithm 4.1 for a clear presentation.

Remark 1. *The two formulations*

$$M_{\Omega_{n+1}} = M_{c_n^{-1}(\Omega_n)} \quad \textit{and} \quad M_{\Omega_{n+1}} = M_{\Omega_n} \circ c_n \tag{4.25}$$

are equivalent under the interpretation of M_Ω as the characteristic function of the domain Ω. This equivalence is seen by evaluating the composition:

$$M_{\Omega_n} \circ c_n(x) = 1 \iff c_n(x) \in \Omega_n \iff x \in c_n^{-1}(\Omega_n). \tag{4.26}$$

Therefore, $M_{\Omega_n} \circ c_n$ is the characteristic function of $c_n^{-1}(\Omega_n)$, which justifies the update $M_{\Omega_{n+1}} = M_{\Omega_n} \circ c_n$ as a computationally convenient form to generate the warped mask.

5. Experiment.

5.1. Implementation Details.

Data Synthesis Synthetic data were generated using MATLAB FEATool with the inlet velocity profile

$$\boldsymbol{g}(\boldsymbol{x}) = v\left(1 - \frac{|\boldsymbol{q}-\boldsymbol{x}|^2}{(|\Gamma^{\mathrm{i}}|/2)^2}\right), \tag{5.1}$$

Algorithm 4.1 The main algorithm

Require: Measured flow image $\tilde{\boldsymbol{u}}$, reference mask M_{Ω_0};
Ensure: Reconstructed flow image $\boldsymbol{u}$, reconstructed mask M_Ω;

$M_{\Omega_1} \leftarrow M_{\Omega_0}$;
$\boldsymbol{\theta}_u, \boldsymbol{\theta}_p, \boldsymbol{\theta}_c \leftarrow$ random initialization;
for $n = 1 : N_{\text{MAX}}$ **do**
    Sample points using M_{Ω_n} as described in Section 4.1;
    $\boldsymbol{\theta}_u, \boldsymbol{\theta}_p \leftarrow$ gradient descent as (4.16) with the sampled points;
    for each $\boldsymbol{x} \in \mathcal{D}$ (defined in (4.2)) **do**
        $\hat{\boldsymbol{u}}(\boldsymbol{x}) \leftarrow \mathcal{N}_u(\boldsymbol{x}; \boldsymbol{\theta}_u)$;
    end for
    $\hat{\boldsymbol{u}}_n \leftarrow \hat{\boldsymbol{u}}$;
    if $\|\tilde{\boldsymbol{u}} - \hat{\boldsymbol{u}}_n\|^2_{L^2(\mathcal{D})}$ converge **then**
        $\boldsymbol{u} \leftarrow \hat{\boldsymbol{u}}_n$, $M_\Omega \leftarrow M_{\Omega_n}$;
        Finish and output;
    end if
    $\boldsymbol{\theta}_c \leftarrow$ gradient descent as (4.24) with $\hat{\boldsymbol{u}}$;
    $c_n \leftarrow \mathcal{N}_c(\hat{\boldsymbol{u}}_n; \boldsymbol{\theta}_c)$;
    $M_{\Omega_{n+1}} \leftarrow M_{\Omega_n} \circ c_n$;
end for

where v is a velocity parameter chosen for a specific experiment, $\boldsymbol{q}$ denotes the centroid of the inlet boundary Γ^{i}, and $|\Gamma^{\text{i}}|$ represents its length (in 2D). All synthesized images were of size 256×256. The spatial resolution for each experiment is reported individually.

Noise Model To simulate the measurement noise in MRI flow imaging, we adopted a signal-dependent noise model combining Gaussian components

$$\tilde{\boldsymbol{u}} = \boldsymbol{u} + \sqrt{|\boldsymbol{u}|} \cdot \boldsymbol{n}_1 + \boldsymbol{n}_2, \tag{5.2}$$

where $\boldsymbol{n}_1 \sim \mathcal{N}(0, \sigma_1^2)$ and $\boldsymbol{n}_2 \sim \mathcal{N}(0, \sigma_2^2)$ are two independent Gaussian noise terms.

Network Architecture The PINN model consists of two separate MLPs for velocity and pressure. The velocity network has 8 layers with 40 neurons per layer, while the pressure network also has 8 layers with 20 neurons per layer. Increasing the number of layers can improve expressiveness, but can lead to a higher computational cost without significant performance gains [44]. The U-Net architecture used to generate the correction mapping is shown in Figure 4 and has been demonstrated to be effective for producing high-quality quasi-conformal mappings in previous works [43, 45].

Error Metric To evaluate the performance of our method, we use several metrics, including Mean Squared Error (MSE), Relative Error (RE), Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM), Dice coefficient, and 95th percentile Hausdorff distance (HD95). Detailed definitions of these metrics are provided in Appendix B.

Computational Platform Our methods were implemented in all experiments using the *Python* programming language in combination with the *PyTorch* framework. The experiments were conducted on a Windows 11 PC equipped with an Nvidia 4060 Ti GPU and a 13th Gen

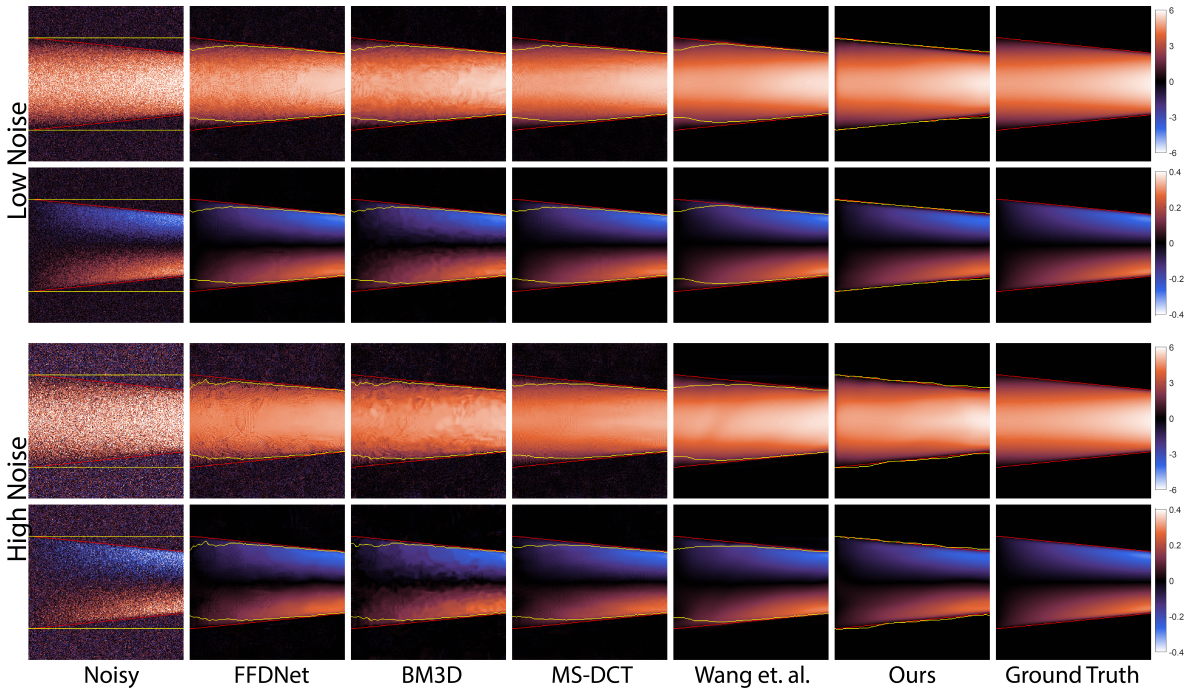

Figure 5. *Qualitative comparison of flow and region reconstruction. Red contours show ground truth; yellow contours indicate predicted domains. Top two rows: low noise group; bottom two: high noise group. Each pair shows x- (top) and y-direction (bottom) velocity. Yellow contour in "Noisy" column marks the initialization.*

Intel(R) Core(TM) i7-13700KF 3.40 GHz CPU.

Parameters We set the weights of the composite loss function as follows if not specified: for fluid subproblem, $\alpha_{\text{data}} = 1.0$, $\alpha_{\text{pde}} = 1.0 \times 10^{-3}$, $\alpha_{\text{bd}} = 1.0$, $\alpha_{\text{in}} = 0.1$; for geometry subproblem, $\alpha_{\text{reg}} = 1.0$, $\alpha_{\text{bc}} = 1.0 \times 10^{1}$, $\alpha_{\text{lap}} = 1$.

5.2. Flow Image over Converging Channel. To validate the proposed method, we conducted experiments on synthetic flow images within a converging channel. The data consists of a two-channel image of size 256×256, representing the vertical and horizontal components of the velocity field over a physical domain measuring 40 cm $\times$ 40 cm. In the simulation setup, the inlet boundary velocity follows a parabolic profile with a maximum velocity of 5 cm/s. We assume a Newtonian fluid with dynamic viscosity $\mu = 0.035\ \text{g} \cdot \text{cm}^{-1} \cdot \text{s}^{-1}$ and density $\rho = 1.05\ \text{g} \cdot \text{cm}^{-3}$, corresponding to a Reynolds number of 150 based on a characteristic length of 5 cm and a characteristic velocity of 1 cm/s. Either low or high levels of Gaussian noise were added to the synthesized images. For low noise, the x-direction flow had zero-mean noise with standard deviation 1, and the y-direction flow had zero-mean noise with standard deviation 0.01. For high noise, the x-direction flow had a standard deviation of 2, and the y-direction flow had a standard deviation of 0.02. The training was conducted for 240 iterations, alternating between 40 iterations of the fluid subproblem and 40 iterations of the geometry subproblem. Each fluid subproblem involves 50,000 collocation points. The entire

		Flow Image Reconstruction				Fluid Domain Segmentation		
		RE	MSE	PSNR	SSIM	MSE	Dice	HD95
Low	Noise	0.4974	0.4742	23.119	0.1962	0.0961	0.9119	24.739
	FFDNet	0.0827	0.0973	32.935	0.7452	0.0454	0.9522	16.454
	BM3D	0.0348	0.0483	31.276	0.8870	0.0457	0.9518	16.342
	MC-DCT	0.0411	**0.0383**	34.523	0.9172	0.0476	0.9498	16.616
	Wang *et al.*	0.0648	0.0486	34.921	**0.9855**	0.0407	0.9573	16.000
	Ours	**0.0124**	0.0414	**35.298**	0.9847	**0.0067**	**0.9933**	**4.000**
High	Noise	1.9621	1.7588	17.304	0.0645	0.0961	0.9119	24.739
	FFDNet	0.1656	0.3241	29.386	0.6564	0.0521	0.9358	19.417
	BM3D	0.1279	0.1012	32.130	0.7733	0.0529	0.9350	20.105
	MC-DCT	0.0891	0.0727	31.283	0.8224	0.0681	0.9292	19.647
	Wang *et al.*	0.0577	0.0495	33.816	0.9637	0.0439	0.9539	21.000
	Ours	**0.0150**	**0.0410**	**35.296**	**0.9841**	**0.0114**	**0.9887**	**4.000**

Table 2

Quantitative results for flow images in the converging channel geometry, comparing FFDNet [49], BM3D [23], MC-DCT [42], Wang et al. [37], and the proposed method.

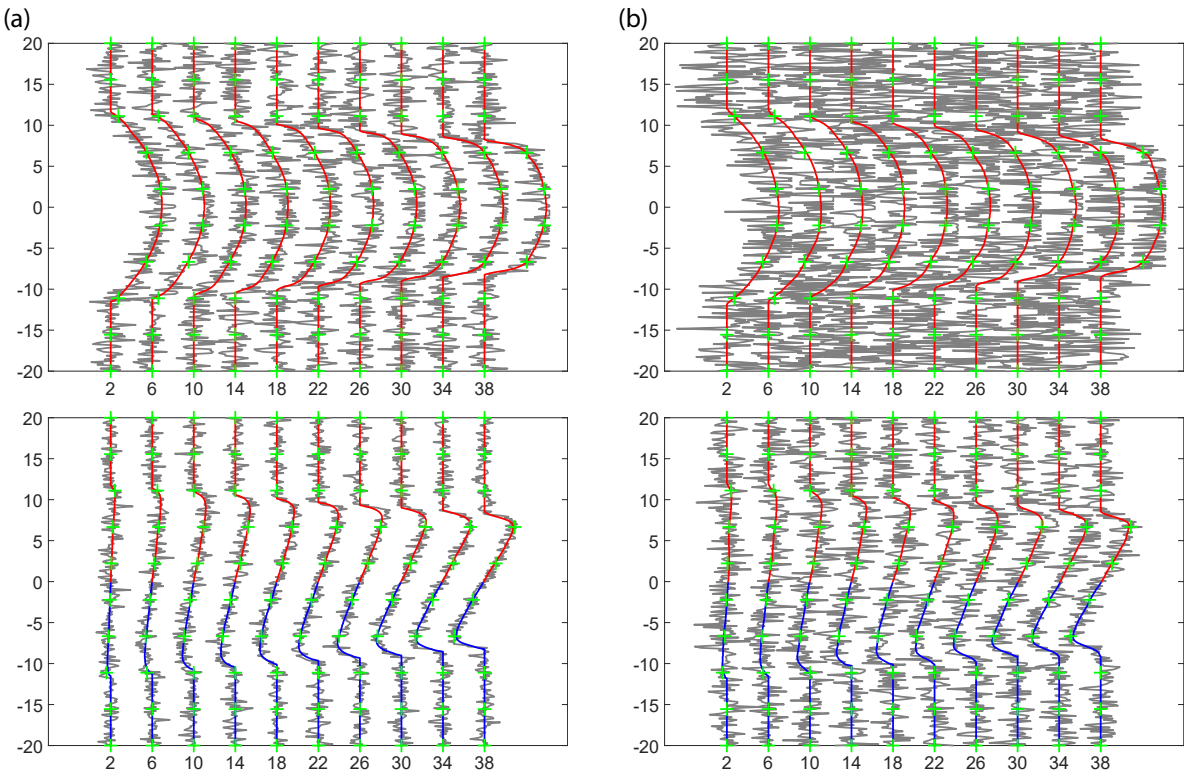

Figure 6. *Discrepancies in velocity fields among the noisy input (gray), reconstructed results (red for positive, blue for negative), and ground truth (green "+"). The upper indicates the velocity in x-axis while the lower for y-axis. (a) Low noise; (b) High noise.*

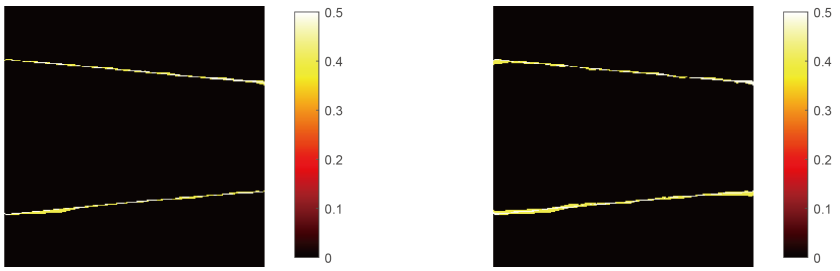

Figure 7. *Illustration of segmentation uncertainty for the converging channel geometry with Gaussian noise. The maps show the standard deviation computed from five noisy images with randomly generated noise. The left corresponds to the low-noise case, while the right represents the high-noise case.*

training process takes approximately 5 minutes for the converging channel geometry case.

We compared our method with learning-based approaches like FFDNet [49] and traditional image processing techniques such as BM3D [23], multiscale DCT [42], and a physics-based method [37]. The evaluation included both quantitative metrics and qualitative visualizations for flow image reconstruction and region segmentation. Since competing methods do not provide segmentations directly, we applied the geodesic active contour model [8] to extract flow regions from their denoised outputs.

Figure 5 shows qualitative comparisons. The red contours indicate the ground-truth flow regions, and the yellow contours show those inferred by each method. The first two rows correspond to low-noise inputs, and the last two correspond to high-noise conditions. For each noise level, the velocity components in the x- and y-directions are shown in the upper and lower rows, respectively. The yellow contour in the "Noisy" column reflects the initial domain estimate used to initialize the applicable methods.

As shown in Figure 5 and Table 2, our method consistently outperforms the others. Regarding flow image quality, conventional image denoising models are unable to fully eliminate artifacts, often resulting in unnatural or physically implausible flow patterns. While the physics-based approach by Wang *et al.* [37] achieves improved performance, its accuracy remains limited in both mechanical reconstruction and domain segmentation, which arises from the lack of coupled geometry evolution and the absence of domain boundary information during the optimization process. In contrast, by incorporating the physics of fluid flow through the Navier–Stokes equations and the associated domain evolution, our approach produces more realistic and accurate reconstructions in both fluid mechanics and domain. The PINN model with boundary information used for velocity estimation further ensures robust and efficient optimization. Regarding region segmentation, it is observed that the active contour model, when applied to denoised images, often yields inaccurate boundaries, particularly near wall regions and close to the inlet, due to its reliance solely on image gradients. In contrast, our method benefits from the underlying velocity field information through a registration-based manner, enabling it to recover flow regions that closely match the ground truth and extract a more accurate flow region.

To closely examine the noisy measurements, the reconstructed velocity field, and the ground truth, we present Figure 6. The upper row shows the x-velocity component, and the bottom row shows the y-velocity component. In each plot, the curves represent velocity

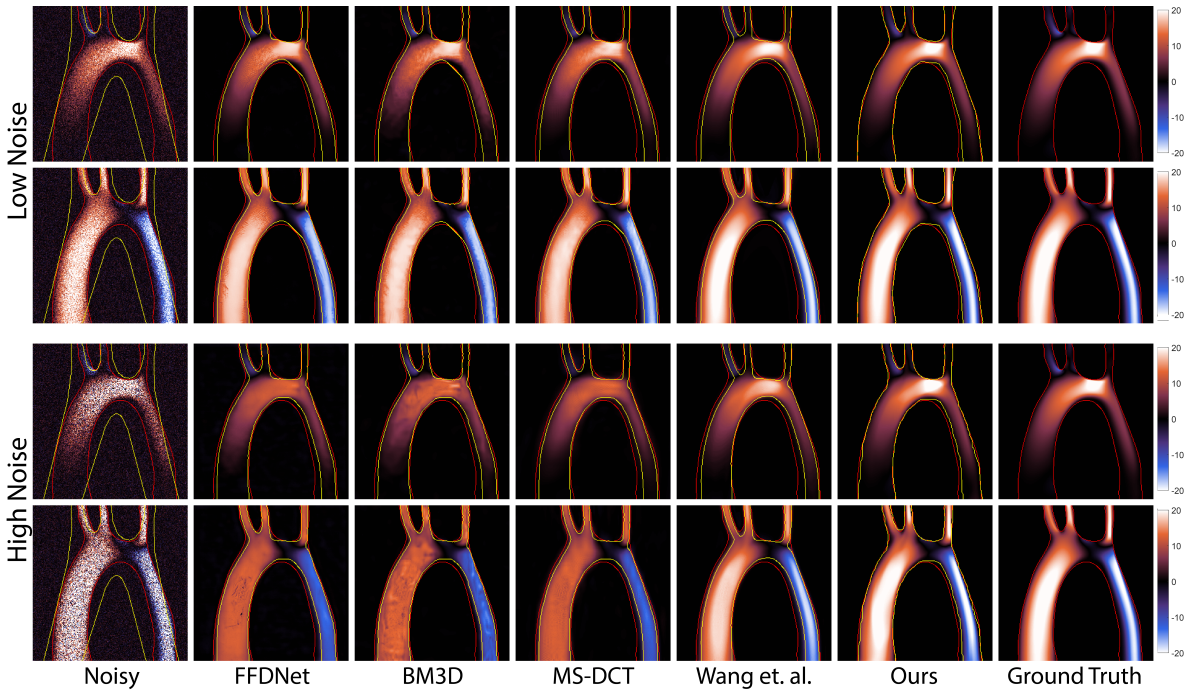

Figure 8. *Qualitative comparison of flow and region reconstruction. Red contours show ground truth; yellow contours indicate predicted domains. Top two rows: low noise group; bottom two: high noise group. Each pair shows x- (top) and y-direction (bottom) velocity. Yellow contour in "Noisy" column marks the initialization.*

magnitudes sampled along a horizontal line at a fixed x-location. The horizontal displacement from the vertical indicates speed — larger deviations correspond to higher velocities. Reconstructed velocities are colored red for positive (rightward/upward) and blue for negative (leftward/downward) values; gray curves represent noisy observations, and green "+" markers indicate ground truth. From the profile observations, the reconstructed velocities closely match the ground truth across inflow, outflow, and intermediate regions. They accurately capture the fluid dynamics and effectively suppress noise, as indicated by the gray curves, while preserving physical consistency.

We also evaluate the uncertainty of our model under different levels of Gaussian noise. Specifically, we apply our method to five independently generated noisy images at both low and high noise levels, and compute the standard deviation of the resulting segmented regions. The corresponding uncertainty map, shown in Figure 7, indicates that the segmentation uncertainty remains relatively low even under high noise. This robustness can be attributed to the global guidance provided by the registration-based geometry modeling.

5.3. Flow Image over Aorta Vessel. To evaluate our method in a more realistic setting, we conducted experiments on synthetic flow images within an aorta geometry. The 256×256 images represent an 8 cm $\times$ 8 cm domain, with the radius of the aorta around 2 cm. Following the simulation setup in [36, 14], a parabolic inlet velocity profile with a peak of 20 cm/s

		Flow Image Reconstruction				Fluid Domain Segmentation		
		RE	MSE	PSNR	SSIM	MSE	Dice	HD95
Low	Noise	2.1716	5.8275	24.375	0.2967	0.1627	0.7904	26.476
	FFDNet	0.1456	0.8605	32.514	0.9251	0.0517	0.9194	15.454
	BM3D	0.1371	0.7917	32.276	0.9176	0.0529	0.9119	14.342
	MC-DCT	0.1670	0.7214	34.675	0.9353	0.0613	0.8922	14.000
	Wang *et al.*	0.1651	0.4825	35.642	0.9788	0.0587	0.9024	13.928
	Ours	**0.0847**	**0.4133**	**38.447**	**0.9869**	**0.0281**	**0.9574**	**8.000**
High	Noise	3.1875	24.0538	18.680	0.2487	0.1627	0.7904	26.476
	FFDNet	0.1566	3.4682	26.672	0.8915	0.0637	0.9019	16.170
	BM3D	0.2654	3.7285	24.287	0.8835	0.0607	0.8881	16.854
	MC-DCT	0.2753	4.1753	27.176	0.9117	0.0648	0.8767	15.000
	Wang *et al.*	0.2328	0.8734	33.374	0.9667	0.0761	0.8937	15.000
	Ours	**0.0964**	**0.5973**	**36.136**	**0.9804**	**0.0296**	**0.9540**	**8.000**

Table 3

Quantitative results for flow images in the aorta geometry, comparing FFDNet [49], BM3D [23], MC-DCT [42], Wang et al. [37], and the proposed method.

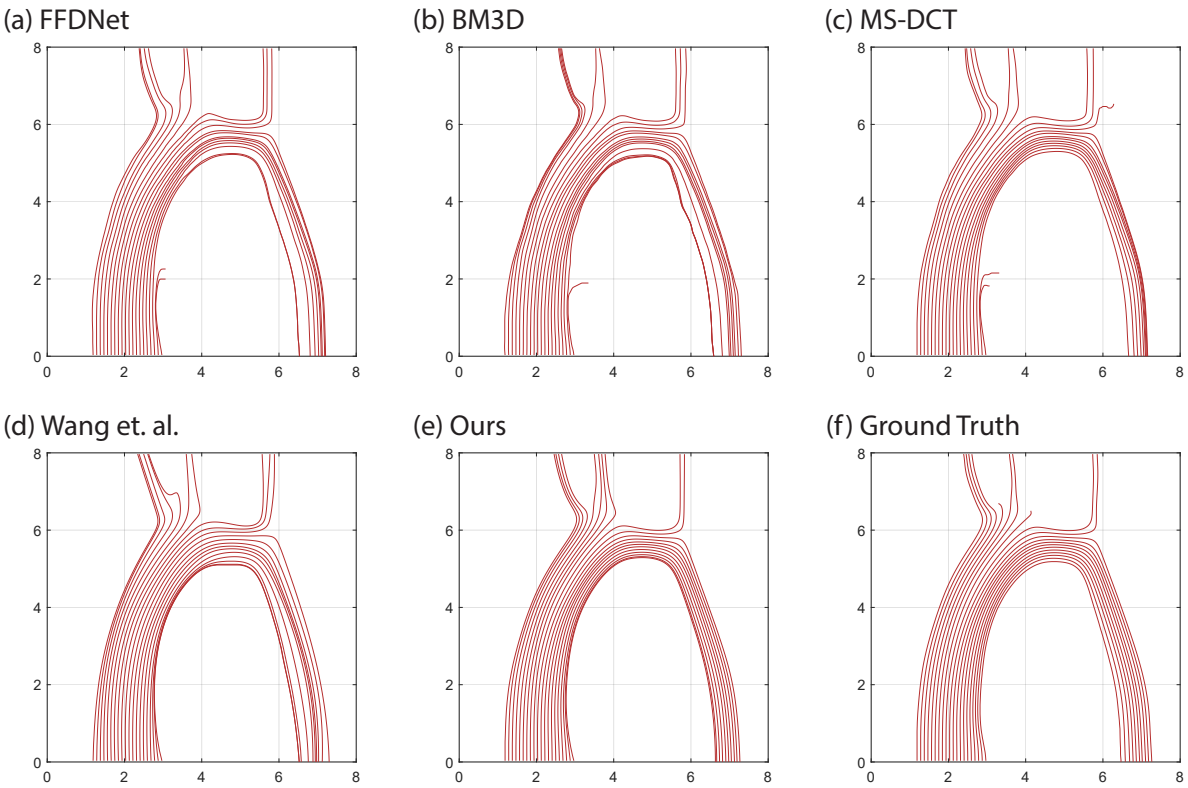

Figure 9. *Streamline visualization reconstructed from a low-noise flow image under identical settings.*

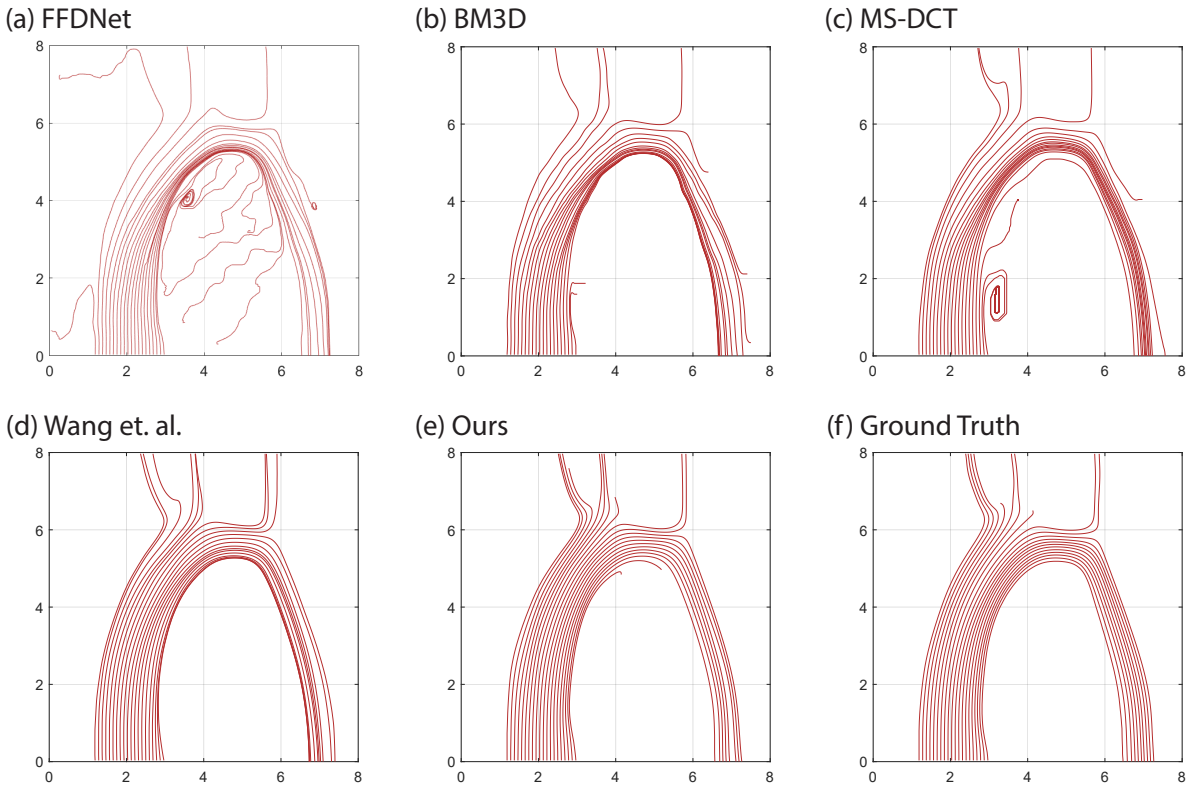

Figure 10. *Streamline visualization reconstructed from a high-noise flow image under identical settings.*

was used. The dynamic viscosity is set to $\mu = 0.035\ \mathrm{g \cdot cm^{-1} \cdot s^{-1}}$ and the density to $\rho = 1.05\ \mathrm{g \cdot cm^{-3}}$. The Reynolds number for the simulation is 600, based on a characteristic length of 1 cm and a characteristic velocity of 20 cm/s. Signal-dependent noise was added following the model in (5.2). For low noise, the standard deviations were set to $\sigma_1 = 1$ and $\sigma_2 = 2$ for the two directions, while for high noise, both directions used $\sigma_1 = 2$ and $\sigma_2 = 2$. The training was performed for 400 iterations, alternating between 40 iterations of the fluid subproblem and 40 iterations of the geometry subproblem. Each fluid subproblem uses 50,000 collocation points. The entire training process takes approximately 9 minutes for the aorta geometry case.

As in the previous case, we compared our method with FFDNet [49], BM3D [23], multiscale DCT [42], and a physics-based method [37], evaluating both flow reconstruction and segmentation. For methods without native segmentation, the geodesic active contour model [8] was applied. Figure 8 presents qualitative results, following the same layout as Figure 5, with red contours indicating ground truth and yellow contours showing inferred domains under both low and high noise conditions.

As shown in the results in Figure 8 and Table 3, our method consistently outperforms the others in all evaluation metrics. Conventional denoising methods such as FFDNet, BM3D, and multiscale DCT struggle to effectively suppress noise, particularly under high-noise conditions.

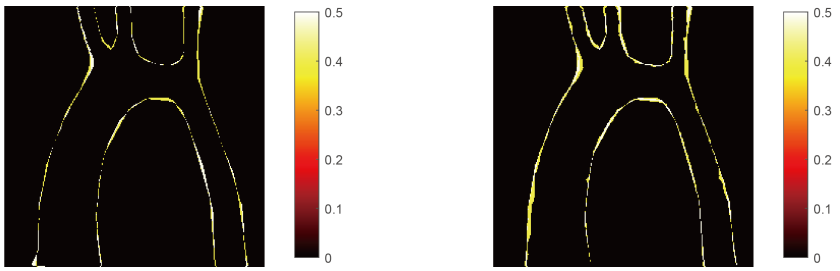

Figure 11. *Illustration of segmentation uncertainty for the aorta geometry with signal-dependent noise. The maps show the standard deviation computed from five noisy images with randomly generated noise. The left corresponds to the low-noise case, while the right represents the high-noise case.*

This limitation is especially evident in the last two columns of the figure, where these methods produce flow fields with pronounced artifacts or physically implausible patterns. In such cases, the reconstructed images fail to preserve the underlying flow structure and offer limited utility for meaningful fluid dynamics analysis. While the physics-based approach by Wang *et al.* [37] achieves improved quantitative performance compared to traditional denoising methods, its accuracy remains insufficient, primarily due to the absence of coupled geometry evolution and of the associated domain boundary information during the optimization process.

In contrast, our method consistently produces clean and physically realistic reconstructions, as demonstrated in Figures 9 and 10, which show reconstructed streamlines from low- and high-noise images, respectively. This improvement is primarily attributed to the incorporation of physical constraints via the Navier–Stokes equations and the robust optimization enabled by the differentiable PINN model, as well as the coupled domain evolution during the process. By enforcing consistency with the governing physics of fluid flow and the domain, our differentiable approach effectively suppresses noise while preserving essential flow features embedded in noisy measurements.

Regarding flow region segmentation, intensity-based methods, such as geodesic active contours, often result in inaccurate boundary detection, particularly near vessel walls, where velocity naturally decreases. This misidentification leads to segmentations that deviate from the true physiological flow region.

Our method, on the contrary, performs flow reconstruction and region segmentation in a coupled manner, allowing each task to inform and refine the other. This mutual reinforcement improves the accuracy of both the reconstructed flow fields and the segmented domains. The synergy between physics-informed reconstruction and segmentation contributes to the overall performance advantage of our approach, as confirmed by both visual results and quantitative evaluations. For the high-noise aorta case, the segmented domain occasionally shows jagged boundaries. This was addressed in [21] by adding a viscous term that implies motion of the domain's boundary by its mean curvature [28]. Instead, we apply morphological opening followed by closing with a disk of radius $r = 6$, which effectively bounds the absolute value of the boundary curvature by $1/r$.

To assess the segmentation uncertainty in the aorta geometry case with signal-dependent noise, we follow a procedure similar to the converging channel experiment. Our method is applied to five independently generated noisy images at both low and high noise levels, and the

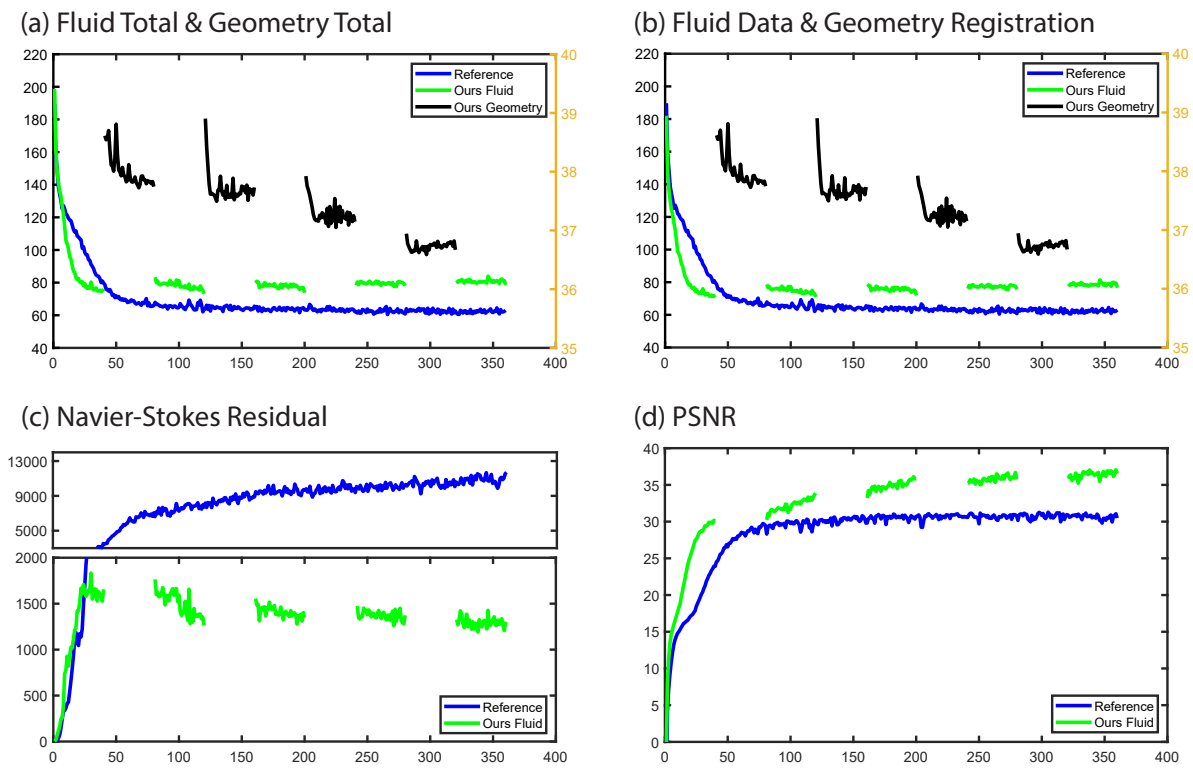

Figure 12. *Training dynamics and convergence behavior for the aorta geometry case by our method and the reference. The reference method is done by removing the Navier–Stokes residual terms and boundary conditions, training the model solely with the data-fidelity term. The green curve represents the total loss of the fluid subproblem, the black curve corresponds to the geometry subproblem, and the blue curve denotes the result obtained by the reference model trained with only the data-fidelity term. (a) Total fluid and geometry losses; (b) data loss in the fluid subproblem and registration loss in the geometry subproblem; (c) Navier–Stokes residuals; (d) PSNR evolution during training.*

standard deviation of the resulting segmented regions is calculated. The uncertainty map in Figure 11 shows that the segmentation remains stable, with controllable variations even under high noise conditions. This consistency highlights the robustness of the proposed framework, benefiting from the globally guided deformation in the registration-based geometry modeling.

We further examine the error plots in Figure 12 to better understand the training dynamics and the role of the physics-informed component, by comparing our method with a reference approach. The reference method is obtained by removing the Navier–Stokes residual terms and boundary conditions, and training the model solely using the data-fidelity term. In these plots, for our method, the green curve represents the total loss of the fluid subproblem, while the black curve corresponds to the geometry subproblem. The blue curve shows the result obtained using the reference approach. As illustrated in Figure 12(a), the alternating behavior between the total fluid loss (y-axis on the left) and the total geometry loss (y-axis on the right) clearly demonstrates the effect of our alternating optimization strategy. Because our method

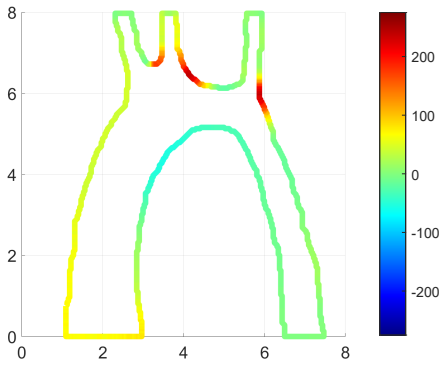

Figure 13. *Wall shear stress (WSS) distribution on the aortic wall obtained from the reconstructed solution in the high-noise case. The color codes the spatial variation of shear stress magnitude* ($\mathrm{dyn/cm^2}$).

incorporates physical constraints, the data error decreases more slowly than in the reference model. However, as shown in Figure 12(c), the minimization of the Navier–Stokes residual leads to significantly improved reconstruction accuracy, which is reflected by the better PSNR in Figure 12(d).

Wall shear stress (WSS) is a clinically significant hemodynamic quantity. Therefore, we further estimate this variable and visualize its spatial distribution in Figure 13. The results are obtained from the high-noise aorta geometry case. Using the reconstructed velocity networks, which allow direct computation of velocity gradients, together with the pressure network, we evaluate the WSS distribution on the vessel boundary. A color-coded plot with an accompanying color bar is provided to illustrate its spatial variation. Combined with the velocity visualization shown in Figure 8, we observe that elevated shear stress occurs in regions where the flow direction changes markedly due to geometric constraints of the vessel, while lower WSS values appear in areas where the velocity field propagates more smoothly.

5.4. Self Ablation. Here, we perform ablation studies to assess the impact of different weighting parameters in the optimization of the fluid and geometry subproblems. The high-niose flow image with Aorta geometry is used.

Test on super-resolution Our method is built upon a meshless PINN solver constrained by the Navier–Stokes equations. Thanks to the meshless formulation, the reconstructed velocity field can be evaluated at arbitrary spatial resolution while maintaining solution accuracy. To demonstrate this advantage, we include a super-resolution experiment using aorta flow data of sizes 64×64, 128×128, and the originally 256×256, For each case, we apply the same reconstruction algorithm, and then evaluate the resulting velocity field on a 256×256 grid. As shown in Table 4, all reconstructed high-resolution velocity fields closely match the 256×256 ground truth, as reflected by the quantitative metrics. This experiment confirms that our method can reliably recover high-quality flow fields even from low-resolution measurements.

Test on $\alpha_{\mathbf{data}}$: We conducted an ablation study to evaluate the effect of the data fitting weight parameter α_{data}, which plays a critical role in aligning the predicted velocity field with the observed data. The best overall performance was achieved at $\alpha_{\mathrm{data}} = 10^0$, striking a strong yet balanced emphasis on data fidelity while maintaining physical regularization. At

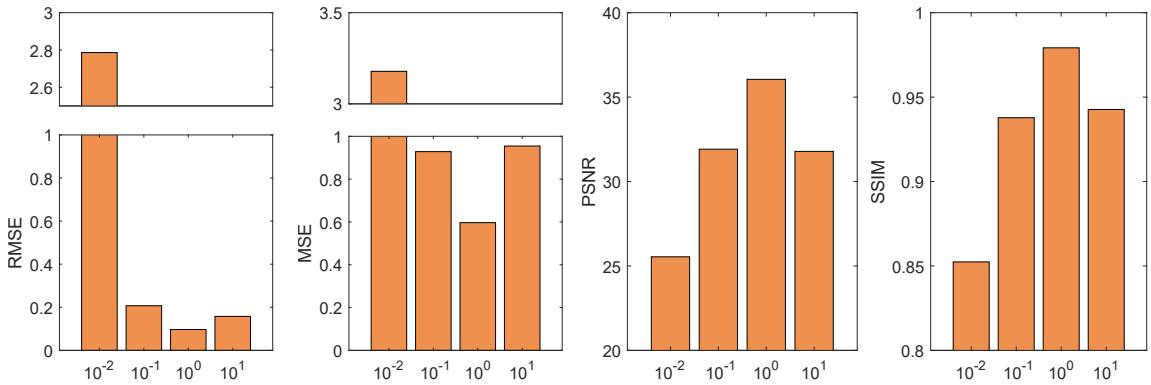

Figure 14. *Ablation study evaluating the impact of the data fitting weight α_{data} on training performance.*

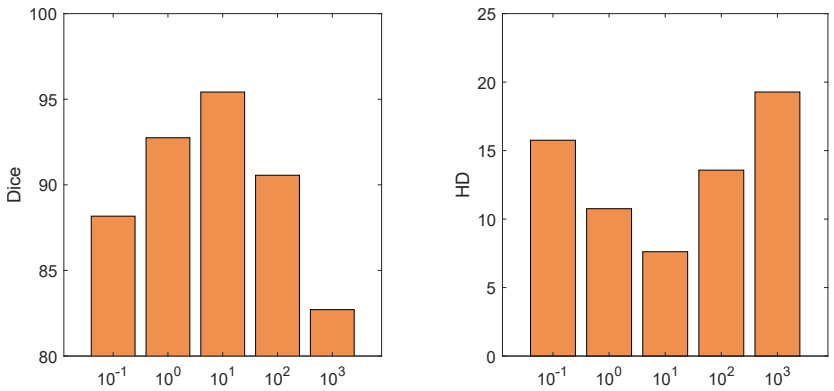

Figure 15. *Ablation study evaluating the impact of Beltrami regularization weight α_{data} on training performance.*

$\alpha_{\text{data}} = 10^1$, the model still performed well, although signs of overfitting appeared as the network prioritized fitting noisy data, which reduced generalization. Lowering the weight to 10^{-1} caused a slight performance drop due to weaker data alignment, particularly in regions with fine flow features. When α_{data} was reduced further to 10^{-2}, the performance suffered significantly: the data constraint was too weak to effectively guide the model, causing the optimization to rely primarily on regularization and resulting in poor recovery of the flow field and domain structure.

Test on α_{bc} : We conducted a self-ablation study on the sensitivity to the Beltrami coefficient regularization weight, α_{bc}. The best performance occurred at $\alpha_{\text{bc}} = 10^1$, balancing mapping regularization and registration. At $\alpha_{\text{bc}} = 10^0$, results were acceptable, although minor deterioration appeared, probably due to increased deformation causing localized shrinkage. Lowering to $\alpha_{\text{bc}} = 10^{-1}$ significantly degraded performance, as weak regularization allowed excessive domain deviation, topological errors, and outlier regions that harmed precision. Conversely, at $\alpha_{\text{bc}} = 10^2$, over-regularization constrained the model, limiting domain propagation and adaptation to subtle variations, leading to incomplete or inaccurate reconstructions. This over-constraint effect intensified at $\alpha_{\text{bc}} = 10^3$.

Resol.	RE	MSE	PSNR	SSIM
64^2	0.1381	0.6891	33.648	0.9498
128^2	0.1207	0.6343	35.187	0.9679
256^2	0.0969	0.5966	36.053	0.9792
α_{data}	RE	MSE	PSNR	SSIM
10^{-2}	2.7861	3.1784	25.542	0.8524
10^{-1}	0.2071	0.9284	31.917	0.9378
10^{0}	0.0969	0.5966	36.053	0.9792
10^{1}	0.1578	0.9546	31.786	0.9427
α_{bc}	$\|\mu\|_2$	Dice	HD	$\|\|\mu\|\|_\infty < 1$
10^{-1}	1.078	0.8817	15.757	✗
10^{0}	0.337	0.9375	10.757	✓
10^{1}	0.068	0.9542	7.616	✓
10^{2}	0.005	0.9056	13.572	✓
10^{3}	0.000	0.8271	19.278	✓

Table 4

Ablation study on the impact of flow image resolution, varying data fidelity weight α_{data} and the Beltrami regularization weight α_{bc} for flow images in an aorta-shaped domain under high-noise conditions.

6. Conclusion. In this work, we propose a novel framework that integrates deep learning with physical modeling to improve the reconstruction of 2D flow images under noisy measurements. The problem is formulated to enforce physical consistency through the Navier–Stokes equations and preserve anatomical structures using quasi-conformal mappings. It is decomposed into two subproblems: a fluid subproblem, which estimates the velocity field within a given domain using Navier–Stokes constraints, and a geometry subproblem, which refines the domain by aligning the reconstructed velocity with observed data via a quasi-conformal mapping. These subproblems are solved iteratively in a Gauss–Seidel fashion until convergence, providing a solution supported by theoretical guarantees. In practice, we employ a Physics-Informed Neural Network for the fluid subproblem and a UNet-based Mapping Estimator for the geometry subproblem.

We extensively validated our method through experiments. Initial tests on synthetic flow images in a converging channel under varying Gaussian noise levels validate the proposed method. Further evaluations on anatomically realistic aorta geometries with signal-dependent noise confirmed its effectiveness in complex real-world settings. Our method consistently outperformed the baseline approaches, delivering more accurate and higher-quality reconstructions. Ablation studies also highlighted the impact of weighting schemes and guided the selection of effective hyperparameters.

Future work will extend this framework to 3D flow reconstruction and 4D dynamic flow analysis by replacing the 2D quasi-conformal mappings with Jacobian-based constraints or hyperelastic energy models to ensure smooth and bijective deformations in higher dimensions. We also plan to adapt the framework for higher Reynolds number flows by employing more expressive architectures, such as PirateNets [38]. Additionally, we aim to tackle more chal-

lenging imaging modalities, such as ultrasound Doppler imaging, which involve higher noise levels and real-time computational constraints. Furthermore, since the current Gauss–Seidel alternating scheme lacks a formal convergence guarantee, and the validation of the split problems relies primarily on empirical observations, we plan to develop a theoretical convergence analysis by adding additional regularization.

REFERENCES

[1] J. Aguayo, C. Bertoglio, and A. Osses, *A distributed resistance inverse method for flow obstacle identification from internal velocity measurements*, Inverse Problems, 37 (2021), p. 025010.

[2] J. Aguayo, C. Bertoglio, and A. Osses, *Distributed parameter identification for the navier–stokes equations for obstacle detection*, Inverse Problems, 40 (2023), p. 015012.

[3] J. Aguayo and H. C. Lincopi, *Analysis of obstacles immersed in viscous fluids using brinkman's law for steady stokes and navier–stokes equations*, SIAM Journal on Applied Mathematics, 82 (2022), pp. 1369–1386.

[4] J. Aguayo and A. Osses, *A stability result for the identification of a permeability parameter on navier-stokes equations*, Inverse Problems, 38 (2022), p. 075001.

[5] S.-i. Amari, *Backpropagation and stochastic gradient descent method*, Neurocomputing, 5 (1993), pp. 185–196.

[6] Y. Bazilevs, V. M. Calo, T. J. Hughes, and Y. Zhang, *Isogeometric fluid-structure interaction: Theory, algorithms, and computations*, Computational Mechanics, 43 (2008), pp. 3–37.

[7] M. Bukač, S. Čanić, R. Glowinski, J. Tambača, and A. Quaini, *Fluid-structure interaction in blood flow capturing non-zero longitudinal structure displacement*, Journal of Computational Physics, 235 (2013), pp. 515–541.

[8] V. Caselles, R. Kimmel, and G. Sapiro, *Geodesic active contours*, International Journal of Computer Vision, 22 (1997), pp. 61–79.

[9] L. Chen and T. Zeng, *A convex variational model for restoring blurred images with large rician noise*, Journal of Mathematical Imaging and Vision, 53 (2015), pp. 92–111.

[10] V. Corona, M. Benning, L. F. Gladden, A. Reci, A. J. Sederman, and C.-B. Schönlieb, *Joint phase reconstruction and magnitude segmentation from velocity-encoded mri data*, in Time-dependent Problems in Imaging and Parameter Identification, Springer, 2021, pp. 1–24.

[11] P. Crosetto, S. Deparis, G. Fourestey, and A. Quarteroni, *Parallel algorithms for fluid-structure interaction problems in haemodynamics*, SIAM Journal on Scientific Computing, 33 (2011), pp. 1598–1622.

[12] C. A. Figueroa, I. E. Vignon-Clementel, K. E. Jansen, T. J. Hughes, and C. A. Taylor, *A coupled momentum method for modeling blood flow in three-dimensional deformable arteries*, Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 5685–5706.

[13] L. Formaggia, J.-F. Gerbeau, F. Nobile, and A. Quarteroni, *On the coupling of 3d and 1d navier–stokes equations for flow problems in compliant vessels*, cmame, 191 (2001), pp. 561–582.

[14] N. Freidoonimehr, M. Arjomandi, A. Zander, and R. Chin, *SimVascular clinical test cases : Aortofemoral normal - 2*, 2025. https://simvascular.github.io/clinical/aortofemoral2.html.

[15] H. Gao, L. Sun, and J.-X. Wang, *Super-resolution and denoising of fluid flow using physics-informed convolutional neural networks without high-resolution labels*, Physics of Fluids, 33 (2021).

[16] P. Getreuer, M. Tong, and L. A. Vese, *A variational model for the restoration of mr images corrupted by blur and rician noise*, in International Symposium on Visual Computing, Springer, 2011, pp. 686–698.

[17] T. Goldstein and S. Osher, *The split bregman method for l1-regularized problems*, SIAM Journal on Imaging Sciences, 2 (2009), pp. 323–343.

[18] M. Kang, M. Kang, and M. Jung, *Nonconvex higher-order regularization based rician noise removal with spatially adaptive parameters*, Journal of Visual Communication and Image Representation, 32 (2015), pp. 180–193.

[19] J. P. Kelliher, *Navier–stokes equations with navier boundary conditions for a bounded domain in the*

plane, SIAM journal on mathematical analysis, 38 (2006), pp. 210–232.
[20] D. Kingma and J. Ba, *Adam: A method for stochastic optimization*, Proc. of IEEE Int. Conf. on Learning Representation, 12 (2015).
[21] A. Kontogiannis, S. V. Elgersma, A. J. Sederman, and M. P. Juniper, *Joint reconstruction and segmentation of noisy velocity images as an inverse navier–stokes problem*, Journal of Fluid Mechanics, 944 (2022), p. A40.
[22] A. Kontogiannis, S. V. Elgersma, A. J. Sederman, and M. P. Juniper, *Bayesian inverse navier–stokes problems: joint flow field reconstruction and parameter learning*, Inverse Problems, 41 (2024), p. 015008.
[23] M. Lebrun, *An analysis and implementation of the bm3d image denoising method*, Image Processing on Line, 2 (2012), pp. 175–213.
[24] Z. Li, Y. Lou, and T. Zeng, *Variational multiplicative noise removal by dc programming*, Journal of Scientific Computing, 68 (2016), pp. 1200–1216.
[25] Y. Lou, T. Zeng, S. Osher, and J. Xin, *A weighted difference of anisotropic and isotropic total variation model for image processing*, SIAM Journal on Imaging Sciences, 8 (2015), pp. 1798–1823.
[26] M. Markl, A. Frydrychowicz, S. Kozerke, M. Hope, and O. Wieben, *4d flow mri*, Journal of Magnetic Resonance Imaging, 36 (2012), pp. 1015–1036.
[27] A. Martín, E. Schiavi, and S. Segura de León, *On 1-laplacian elliptic equations modeling magnetic resonance image rician denoising*, Journal of Mathematical Imaging and Vision, 57 (2017), pp. 202–224.
[28] B. Merriman, J. K. Bence, and S. Osher, *Diffusion generated motion by mean curvature*, Department of Mathematics, University of California, Los Angeles, 1992.
[29] J. R. Parker, *Algorithms for Image Processing and Computer Vision*, John Wiley & Sons, 2010.
[30] C. S. Peskin, *Flow patterns around heart valves: A numerical method*, Journal of Computational Physics, 10 (1972), pp. 252–271, https://doi.org/https://doi.org/10.1016/0021-9991(72)90065-4.
[31] N. H. Pijls, B. de Bruyne, K. Peels, P. H. van der Voort, H. J. Bonnier, J. Bartunek, and J. J. Koolen, *Measurement of fractional flow reserve to assess the functional severity of coronary-artery stenoses*, New England Journal of Medicine, 334 (1996), pp. 1703–1708.
[32] A. Quarteroni and L. Formaggia, *Mathematical modelling and numerical simulation of the cardiovascular system*, Handbook of numerical analysis, 12 (2004), pp. 3–127.
[33] H. Sohr, *The Navier-Stokes Equations: An elementary Functional Analytic Approach*, Springer Science & Business Media, 2012.
[34] L. Sun and J.-X. Wang, *Physics-constrained bayesian neural network for fluid flow reconstruction with sparse and noisy data*, Theoretical and Applied Mechanics Letters, 10 (2020), pp. 161–169.
[35] P. D. T. Tao and E. B. Souad, *Algorithms for solving a class of nonconvex optimization problems. methods of subgradients*, in Fermat Days 85: Mathematics for Optimization, vol. 129, North-Holland, 1986, pp. 249–271.
[36] A. Updegrove, N. M. Wilson, J. Merkow, H. Lan, A. L. Marsden, and S. C. Shadden, *Simvascular: an open source pipeline for cardiovascular simulation*, Annals of Biomedical Engineering, 45 (2017), pp. 525–541.
[37] H. Wang, Y. Liu, and S. Wang, *Dense velocity reconstruction from particle image velocimetry/particle tracking velocimetry using a physics-informed neural network*, Physics of fluids, 34 (2022).
[38] S. Wang, B. Li, Y. Chen, and P. Perdikaris, *Piratenets: Physics-informed deep learning with residual adaptive networks*, Journal of Machine Learning Research, 25 (2024), pp. 1–51.
[39] Y. Wang, A. Quaini, and S. Čanić, *A higher-order discontinuous galerkin/arbitrary lagrangian eulerian partitioned approach to solving fluid–structure interaction problems with incompressible, viscous fluids and elastic structures*, Journal of Scientific Computing, 76 (2018), pp. 481–520.
[40] Y. Xiao, T. Zeng, J. Yu, and M. K. Ng, *Restoration of images corrupted by mixed gaussian-impulse noise via l1–l0 minimization*, Pattern Recognition, 44 (2011), pp. 1708–1720.
[41] Z. Yan, Z. Yao, W. Guo, D. Shang, R. Chen, J. Liu, X.-C. Cai, and J. Ge, *Impact of pressure wire on fractional flow reserve and hemodynamics of the coronary arteries: A computational and clinical study*, IEEE Trans. on Biomedical Engineering, 70 (2022), pp. 1683–1691.
[42] G. Yu and G. Sapiro, *Dct image denoising: A simple and effective image denoising algorithm*, Image Processing on Line, 1 (2011), pp. 292–296.

[43] D. Zhang and L. M. Lui, *Topology-preserving 3d image segmentation based on hyperelastic regularization*, Journal of Scientific Computing, 87 (2021), p. 74.
[44] H. Zhang, R. H. Chan, and X.-C. Tai, *A meshless solver for blood flow simulations in elastic vessels using a physics-informed neural network*, SIAM Journal on Scientific Computing, 46 (2024), pp. C479–C507.
[45] H. Zhang, Q. Chen, and L. M. Lui, *Deformation-invariant neural network and its applications in distorted image restoration and analysis*, Neural Networks, (2025), p. 107378.
[46] H. Zhang, T. L. Ip, and L. M. Lui, *Quasi-conformal convolution: A learnable convolution for deep learning on riemann surfaces*, arXiv preprint arXiv:2502.01356, (2025).
[47] H. Zhang and L. M. Lui, *A learning-based framework for topology-preserving segmentation using quasiconformal mappings*, Neurocomputing, 600 (2024), p. 128124.
[48] H. Zhang and X.-C. Tai, *Full 3d blood flow simulation in curved deformable vessels using physics-informed neural networks*, Acta Mathematica Universitatis Comenianae, 93 (2024), pp. 235–250.
[49] K. Zhang, W. Zuo, and L. Zhang, *Ffdnet: Toward a fast and flexible solution for cnn-based image denoising*, IEEE Transactions on Image Processing, 27 (2018), pp. 4608–4622.

Appendix A. Proof For Theorem 3.3.

Proof.

Conclusion (a): $c_n \to \boldsymbol{id}$ in $L^\infty(D)$.

Since each f_n is a diffeomorphism of D, the limit f_* is also a diffeomorphism. Thus, $f_n \to f_*$ uniformly on D.

By Lemma A.1, we have $c_n \to \boldsymbol{id}$ in $L^\infty(D)$. This implies that the objective functional $\|\tilde{u} - u_n \circ c_n\|_{L^2(D)}$ converges to $\|\tilde{u} - u\|^2_{L^2(D)}$ as the iterations proceed given that D is compact. Such convergence reflects the diminishing need for geometric correction.

Conclusion (b) - f_* is a partial and local minimizer

From assumption (1) and (2), and $u_n = \mathcal{U}[f_n]$ by definition, we conclude that

$$\mathcal{U}[f_n] \to \mathcal{U}[f_*] = u_* \quad \text{in } L^\infty(D),$$

which implies

$$J_{\mathcal{U}}(f_n) = \|\tilde{u} - \mathcal{U}[f_n]\|_{L^2(D)} \to \|\tilde{u} - \mathcal{U}[f_*]\|_{L^2(D)} = J_{\mathcal{U}}(f_*).$$

- Step 1:

Let $\varepsilon > 0$ be arbitrary. By assumption (1), $u_n \to u_*$ in $L^\infty(D)$. So there exists N_1 such that for all $n \geq N_1$,

$$\|u_n - u_*\|_{L^\infty(D)} < \frac{\varepsilon}{3}.$$

Now fix a small perturbation $c \in \mathcal{H}(D)$, close to identity, which we will quantify later. Define

$$f = c^{-1} \circ f_n \quad \text{and} \quad \mathcal{U}[f] := \mathcal{U}\left[c^{-1} \circ f_n\right].$$

With $c_n = \arg\min_{c \in \mathcal{H}(D)} \|\tilde{u} - u_n \circ c\|_{L^2(D)}$, we have

$$\|\tilde{u} - u_n \circ c_n\|_{L^2(D)} \leq \|\tilde{u} - u_n \circ c\|_{L^2(D)}, \quad \forall c \in \mathcal{H}(D). \tag{A.1}$$

- Step 2: Inequality for $\left\|\mathcal{U}\left[c^{-1} \circ f_n\right] - \mathcal{U}[f_n]\right\|_{L^2(D)}$

By the continuity of the mapping in assumption (3), for every $\varepsilon > 0$, there exists $\delta > 0$ such that for $f, f_n \in \mathcal{H}(D)$,

$$\|f - f_n\|_{L^\infty(D)} < \delta \quad \Rightarrow \quad \|\mathcal{U}[f] - \mathcal{U}[f_n]\|_{L^2(D)} < \frac{\varepsilon}{3}.$$

By conclusion (a), we have $\|c^{-1} \circ f_n - f_n\|_{L^\infty(D)} < \delta$ for n large enough. Hence, we get

$$\left\|\mathcal{U}\left[c^{-1} \circ f_n\right] - \mathcal{U}[f_n]\right\|_{L^2(D)} < \frac{\varepsilon}{3}. \tag{A.2}$$

- Step 3: Inequality for $\|u_n \circ c - u_n\|_{L^2(D)}$

As u_n satisfies (3.7) for fixed f_n, with assumption (4), there is a constant M such that

$$|u_n(c(x)) - u_n(x)|^2 \leq M^2 |c(x) - x|^2.$$

Integrating over D, we get

$$\|u_n \circ c - u_n\|^2_{L^2(D)} = \int_D |u_n(c(x)) - u_n(x)|^2 \, dx \leq M^2 \int_D |c(x) - x|^2 dx$$

Now since $\|c - \mathrm{id}\|_{L^\infty(D)} < \delta$, if we choose

$$\delta < \frac{\varepsilon}{3M\sqrt{|D|}}$$

We can have

$$\|u_n \circ c - u_n\|_{L^2(D)} < M\delta\sqrt{|D|} < \frac{\varepsilon}{3} \tag{A.3}$$

- Step 4:

With (A.2) and (A.3), we have

$$\left\|u_n \circ c - \mathcal{U}\left[c^{-1} \circ f_n\right]\right\|_{L^2(D)} \leq \|u_n \circ c - u_n\|_{L^2(D)} + \left\|u_n - \mathcal{U}\left[c^{-1} \circ f_n\right]\right\|_{L^2(D)} < \frac{2\varepsilon}{3}. \tag{A.4}$$

We now compare

$$J_{\mathcal{U}}(f) = \left\|\tilde{u} - \mathcal{U}\left[c^{-1} \circ f_n\right]\right\|_{L^2(D)}, \quad J_{\mathcal{U}}\left(f_{n+1}\right) = \left\|\tilde{u} - \mathcal{U}\left[c_n^{-1} \circ f_n\right]\right\|_{L^2(D)}.$$

By the triangle inequality

$$\left\|\tilde{u} - \mathcal{U}\left[c^{-1} \circ f_n\right]\right\| \geq \|\tilde{u} - u_n \circ c\| - \left\|u_n \circ c - \mathcal{U}\left[c^{-1} \circ f_n\right]\right\|.$$

Using (A.4)

$$\left\|\tilde{u} - \mathcal{U}\left[c^{-1} \circ f_n\right]\right\| > \|\tilde{u} - u_n \circ c\| - \frac{2\varepsilon}{3}.$$

Together with (A.1), we have

$$J_{\mathcal{U}}(f) \geq \|\tilde{u} - u_n \circ c_n\| - \frac{2\varepsilon}{3}. \tag{A.5}$$

Similarly, we can have $\left\|u_n \circ c_n - \mathcal{U}\left[c_n^{-1} \circ f_n\right]\right\| < \varepsilon/3$ by the techniques as above, and obtain

$$\|\tilde{u} - u_n \circ c_n\| \geq \|\tilde{u} - \mathcal{U}\left[f_{n+1}\right]\| - \|\mathcal{U}\left[f_{n+1}\right] - u_n \circ c_n\| > J\left(f_{n+1}\right) - \frac{\varepsilon}{3}. \tag{A.6}$$

Hence

$$J_{\mathcal{U}}(f) > J_{\mathcal{U}}\left(f_{n+1}\right) - \varepsilon.$$

Take $n \to \infty$, then $f_{n+1} \to f_*$,$\mathcal{U}\left[f_{n+1}\right] \to \mathcal{U}\left[f_*\right]$, so

$$\liminf_{f \to f_*} J_{\mathcal{U}}(f) \geq J_{\mathcal{U}}\left(f_*\right). \tag{A.7}$$

Thus, f_* is a partial and local minimizer of the misfit.

Conclusion (c) - u_* is a partial and local minimizer

Denote the solution set for u as

$$\mathcal{A} := \left\{ \mathcal{U}[f] \in L^2(D) \mid f \in \mathcal{H}(D) \right\}.$$

Define the functional as

$$J(u) = \|\tilde{u} - u\|_{L^2(D)}.$$

As $f_* \in \mathcal{H}(D)$ is a local minimizer for $f \mapsto J(\mathcal{U}[f])$, there exists $\delta_1 > 0$ such that for all $f \in \mathcal{H}(D)$ and $\|f - f_*\|_{L^\infty(D)} < \delta_1$ such that

$$J(\mathcal{U}[f]) = \|\tilde{u} - \mathcal{U}[f]\|_{L^2(D)} \geq \|\tilde{u} - \mathcal{U}[f_*]\|_{L^2(D)} = J(u_*). \tag{A.8}$$

Let $\varepsilon > 0$, by assumption (3), there exists $\delta_2 > 0$ such that for all $f \in \mathcal{H}(D)$, which satisfy $\|f - f_*\|_{L^\infty(D)} < \delta_2$, we have

$$\|\mathcal{U}[f] - \mathcal{U}[f_*]\|_{L^2(D)} < \varepsilon,$$

where we can make $\delta_2 < \delta_1$ without loss of generality.
Then, for any $u \in \mathcal{A}$ and $\|\mathcal{U}[f] - \mathcal{U}[f_*]\|_{L^2(D)} < \varepsilon$, we have

$$\|f - f_*\|_{L^\infty(D)} < \delta_2 < \delta_1.$$

By (A.8), we have

$$J(u) = J(\mathcal{U}[f]) \geq J(\mathcal{U}[f_*]) = J(u_*), \tag{A.9}$$

which means u_* is a partial and local minimizer of J over $\mathcal{A}$. ∎

In the theory below, we show that if the sequence of mappings f_n, obtained from some iterative algorithm for minimizing $\|\tilde{u} - u\|^2_{L^2(D)}$ under (3.7), converges, then the corresponding correction mappings $c_n = f_{n+1} \circ f_n^{-1}$ will converge to the identity mapping.

Lemma A.1. *Let $D \subset \mathbb{C}$ be a compact set. Given a sequence of diffeomorphisms $f_n : D \to D$. We have*

$$f_n \to f \quad \textit{uniformly on } D, \ \textit{where } f : D \to D \textit{ is a diffeomorphism,} \tag{A.10}$$

if and only if

$$f_{n+1} \circ f_n^{-1} \to \boldsymbol{id}_D \quad \textit{uniformly on } D. \tag{A.11}$$

Proof.
(=>):

Let $\varepsilon > 0$ be arbitrary. Since $f_n \to f$ uniformly on D, there exists $N_1 \in \mathbb{N}$ such that for all $n \geq N_1$,

$$\sup_{x \in D} \|f_n(x) - f(x)\| < \frac{\varepsilon}{2}. \tag{A.12}$$

Additionally, the uniform convergence of $f_n \to f$ and the fact that each f_n and f are diffeomorphisms implies that $f_n^{-1} \to f^{-1}$ uniformly on D by Lemma A.2.

Let $N = \max(N_1, N_2)$. For all $n \geq N$ and all $x \in D$, define $y := f_n^{-1}(x)$. Then $x = f_n(y)$, and we compute

$$f_{n+1} \circ f_n^{-1}(x) = f_{n+1}(y), \tag{A.13}$$

so that

$$\|f_{n+1} \circ f_n^{-1}(x) - x\| = \|f_{n+1}(y) - f_n(y)\|. \tag{A.14}$$

Now, we insert $f(y)$ and use the triangle inequality

$$\|f_{n+1}(y) - f_n(y)\| \leq \|f_{n+1}(y) - f(y)\| + \|f(y) - f_n(y)\| < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon. \tag{A.15}$$

Since this holds for all $x \in D$, we have

$$\sup_{x \in D} \|f_{n+1} \circ f_n^{-1}(x) - x\| < \varepsilon, \quad \forall n \geq N. \tag{A.16}$$

Hence,

$$f_{n+1} \circ f_n^{-1} \to \boldsymbol{id}_D \quad \text{uniformly on } D. \tag{A.17}$$

(<=):
With $f_{n+1} \circ f_n^{-1} \to \boldsymbol{id}_D$ uniformly on D for every $\varepsilon > 0$, there exists $N \in \mathbb{N}$ such that for all $n \geq N$ and all $x \in D$,

$$|f_{n+1}(f_n^{-1}(x)) - x| < \varepsilon. \tag{A.18}$$

Let $y = f_n^{-1}(x)$, so $x = f_n(y)$. Then the above becomes

$$|f_{n+1}(y) - f_n(y)| < \varepsilon, \quad \forall y \in D. \tag{A.19}$$

Thus,

$$\sup_{y \in D} |f_{n+1}(y) - f_n(y)| < \varepsilon. \tag{A.20}$$

This shows that the sequence $\{f_n\}$ is uniformly Cauchy in $C(D)$, the space of continuous functions on the compact set D with the sup norm.
Since D is compact and each f_n is continuous, there exists a continuous function $f : D \to D$ such that

$$f_n \to f \quad \text{uniformly on } D. \tag{A.21}$$

■

Lemma A.2. *Let $D \subset \mathbb{C}$ be a compact set, and let $\{f_n\}_{n=1}^{\infty}$ be a sequence of diffeomorphism from D onto itself such that $f_n \to f$ uniformly on D. Assume that the limit function $f : D \to D$ is also a diffeomorphism. Then*

$$f_n^{-1} \to f^{-1} \quad \text{uniformly on } D. \tag{A.22}$$

Proof. For any $\varepsilon > 0$. Since f_n^{-1} is continuous on the compact set D, there exists $\delta > 0$ such that for any $x_1, x_2 \in D$,

$$|x_1 - x_2| < \delta \quad \Rightarrow \quad |f_n^{-1}(x_1) - f_n^{-1}(x_2)| < \varepsilon. \tag{A.23}$$

Now fix any $x \in D$, and let $y = f^{-1}(x)$. We can have $x = f(y)$ and $x_n := f_n(y)$. For δ mentioned above, we can find a sufficiently large n, as $f_n \to f$ uniformly, to obtain

$$|x_n - x| = |f_n(y) - f(y)| < \delta. \tag{A.24}$$

Since f_n is bijective, we have $f_n^{-1}(x_n) = y = f^{-1}(x)$. Together with (A.23), we have

$$\begin{aligned} |f_n^{-1}(x) - f^{-1}(x)| &\le |f_n^{-1}(x) - f_n^{-1}(x_n)| + |f_n^{-1}(x_n) - f^{-1}(x)| \\ &< \varepsilon + 0 = \varepsilon. \end{aligned} \tag{A.25}$$

Therefore, $f_n^{-1} \to f^{-1}$ uniformly on D. ∎

Appendix B. Metrics Definition.

To evaluate the accuracy and speed of our methods, some accuracy measurements are reported in the following sections.

The relative error (RE) is defined as

$$\mathrm{RE}(\boldsymbol{u}^*, \boldsymbol{u}^{\mathrm{GT}}) = \sum_{\boldsymbol{x} \in \mathcal{G}} \frac{|\boldsymbol{u}^*(\boldsymbol{x}) - \boldsymbol{u}^{\mathrm{GT}}(\boldsymbol{x})|^2}{|\boldsymbol{u}^{\mathrm{GT}}(\boldsymbol{x})|^2}, \tag{B.1}$$

where $\boldsymbol{u}^*$ is the reconstructed flow, $\boldsymbol{u}^{\mathrm{GT}}$ is the ground truth flow, and $\mathcal{G}$ denotes the set of pixel points.

Peak Signal-to-Noise Ratio (PSNR) is defined as

$$\mathrm{PSNR}(\boldsymbol{u}^*, \boldsymbol{u}^{\mathrm{GT}}) = 10 \log_{10} \left(\frac{\max_{\boldsymbol{x} \in \mathcal{G}} (\boldsymbol{u}^{\mathrm{GT}}(\boldsymbol{x}))^2}{\frac{1}{|\mathcal{G}|} \sum_{\boldsymbol{x} \in \mathcal{G}} |\boldsymbol{u}^*(\boldsymbol{x}) - \boldsymbol{u}^{\mathrm{GT}}(\boldsymbol{x})|^2} \right), \tag{B.2}$$

where $|\mathcal{G}|$ is number of points in $\mathcal{G}$.

Structural Similarity Index (SSIM) is defined as

$$\mathrm{SSIM}(\boldsymbol{u}^*, \boldsymbol{u}^{\mathrm{GT}}) = \frac{1}{|\mathcal{G}|} \sum_{\boldsymbol{x} \in \mathcal{G}} \frac{(2\mu_{\boldsymbol{u}^*}(\boldsymbol{x})\mu_{\boldsymbol{u}^{\mathrm{GT}}}(\boldsymbol{x}) + C_1)\left(2\sigma_{\boldsymbol{u}^*, \boldsymbol{u}^{\mathrm{GT}}}(\boldsymbol{x}) + C_2\right)}{(\mu_{\boldsymbol{u}^*}(\boldsymbol{x})^2 + \mu_{\boldsymbol{u}^{\mathrm{GT}}}(\boldsymbol{x})^2 + C_1)\,(\sigma_{\boldsymbol{u}^*}(\boldsymbol{x})^2 + \sigma_{\boldsymbol{u}^{\mathrm{GT}}}(\boldsymbol{x})^2 + C_2)}, \tag{B.3}$$

where C_1, C_2 are small constants to stabilize the division, and

$$\mu_{\boldsymbol{u}^*} = \boldsymbol{u}^* * w, \qquad \sigma^2_{\boldsymbol{u}^*} = (\boldsymbol{u}^*)^2 * w - (\mu_{\boldsymbol{u}^*})^2,$$
$$\mu_{\boldsymbol{u}^{\mathrm{GT}}} = \boldsymbol{u}^{\mathrm{GT}} * w, \quad \sigma^2_{\boldsymbol{u}^{\mathrm{GT}}} = (\boldsymbol{u}^{\mathrm{GT}})^2 * w - (\mu_{\boldsymbol{u}^{\mathrm{GT}}})^2,$$
$$\sigma_{\boldsymbol{u}^*,\boldsymbol{u}^{\mathrm{GT}}} = (\boldsymbol{u}^* \cdot \boldsymbol{u}^{\mathrm{GT}}) * w - \mu_{\boldsymbol{u}^*} \cdot \mu_{\boldsymbol{u}^{\mathrm{GT}}},$$

where ω is a Gaussian kernel of mean as 0 and standard deviation as 1.5. The convolution is with reflect padding

The Dice coefficient measures the spatial overlap of two binary masks and is defined as

$$\mathrm{Dice}(\Omega_*, \Omega_{\mathrm{GT}}) = \frac{2|\Omega_* \cap \Omega_{\mathrm{GT}}|}{|\Omega_*| + |\Omega_{\mathrm{GT}}|}, \tag{B.4}$$

where Ω_* and Ω_{GT} represent the predicted and ground-truth regions. $|\Omega|$ is the area of Ω. A value of 1 indicates perfect overlap.

The 95th percentile Hausdorff distance (HD95) measures the boundary discrepancy between two segmented regions by taking the 95th percentile of the bidirectional surface distances, thereby reducing the influence of outlier points or noisy boundaries. It is defined as

$$\mathrm{HD}_{95}(\Omega_*, \Omega_{\mathrm{GT}}) = \max\Big\{\mathrm{P}_{95}\big(\{d(x, \Omega_{\mathrm{GT}}) \mid x \in \Omega_*\}\big),\ \mathrm{P}_{95}\big(\{d(y, \Omega_*) \mid y \in \Omega_{\mathrm{GT}}\}\big)\Big\}, \tag{B.5}$$

where $d(x, \Omega_{\mathrm{GT}}) = \inf_{y \in \Omega_{\mathrm{GT}}} \|x - y\|$, $d(y, \Omega_*) = \inf_{x \in \Omega_*} \|y - x\|$ and $\mathrm{P}_{95}(\{z_i\})$ denotes the 95% smallest of the set of $\{z_i\}$.