Journal of Applied and Numerical Analysis
2025, Vol. 3, pp. 113–134
doi: 10.30970/ana.2025.3.113

A hybrid variational–learning approach for single-view 3-D face reconstruction

Anhelina Modenko[1], Lilia Diakoniuk[1], Yuriy Yashchuk[1]

[1] *Ivan Franko National University of Lviv, Universytets'ka 1, 79000 Lviv, Ukraine*

Анотація. Досліджується задача реконструкції тривимірної геометрії поверхні обличчя з послідовності RGB-зображень. Глибина точок в кожному кадрі попередньо оцінюється згортковою нейронною мережею, яка забезпечує початкове хмарне представлення поверхні. Для перетворення цих дискретних спостережень на згладжену та топологічно узгоджену тривимірну модель, сформульовано варіаційну задачу мінімізації функціоналу, що поєднує три складові. Перша складова вимірює відхилення відновленої поверхні від прогнозованої мережею глибини, забезпечуючи узгодженість даних; друга – диференціальний регуляризатор кривини поверхні – пригнічує локальний шум та зберігає деталі поверхні; а третя — часово-узгоджувальна складова, яка обмежує надмірні коливання між послідовними кадрами, дозволяючи лише справжні зміни виразу обличчя. Запропонована схема дискретизації поверхні базується на модифікації тріангуляції Делоне у тривимірному просторі, яка зберігає локальні геометричні властивості та забезпечує ліпшицеві оцінки похибки для доданка узгодження з даними. Оптимізація виконується явною інтеграцією дискретного лапласівського потоку на вершинах сітки; локальний оператор усуває потребу в глобальному розв'язуванні розріджених систем і зберігає лінійну складність обчислень для кожного кадру відеопотоку. Експериментальна оцінка на наборах даних показує, що запропонований гібридний підхід стабільно зменшує геометричну похибку та пригнічує часові артефакти порівняно з класичними ключ-точковими і суто нейронними методами, однак це досягається ціною дещо вищої щільності сітки та потреби у налаштуванні ваги регуляризатора.

Abstract. The paper addresses the reconstruction of three-dimensional facial geometry from a sequence of RGB images. For every frame, a convolutional neural network estimates per-pixel depth, yielding a point-cloud representation of the surface. To convert these discrete observations into a smooth, topologically consistent 3-D model, we adopt a variational formulation that seeks to minimise an energy functional comprising three terms. The data-fidelity term drives the reconstructed surface toward the network-predicted depth; a geometric-smoothness term penalising surface curvature to suppress noise and preserve detail; and a temporal-coherence term that discourages spurious frame-to-frame jitter while allowing genuine facial motion. Surface discretisation is performed by a modified three-dimensional Delaunay triangulation that preserves local geometric relations and provides Lipschitz-consistent error bounds for the data term. Optimization proceeds via explicit integration of a discrete Laplacian flow on the mesh vertices; this local operator obviates the need for global sparse linear system resolution while maintaining linear computational complexity per video frame. Experiments on the datasets confirm that the proposed approach consistently reduces geometric error and mitigates temporal artifacts relative to classical keypoint-based and purely neural methods, albeit at the cost of a slightly denser mesh and the need to tune the regularization weight.

Key words: 3-D face reconstruction, variational depth fusion, biharmonic regularisation, real-time RGB-D processing, Delaunay mesh generation

1 Introduction

Reconstructing three-dimensional (3-D) geometry from visual data is a *canonical* inverse problem in computer vision: we observe 2-D measurements on the image plane and infer the unknown 3-D surface that could have generated them. Specifically, an RGB photograph recorded by a pinhole camera can be viewed as a mapping

$$I : \Omega \subset \mathbb{R}^2 \longrightarrow \mathbb{R}^3,$$

where the domain Ω denotes the set of pixel coordinates and each value in $\mathbb{R}^3$ encodes the red, green, and blue colour channels. If a depth sensor is available – or, as in this work, a neural network supplies a depth estimate — one obtains an additional scalar field

$$D : \Omega \subset \mathbb{R}^2 \longrightarrow \mathbb{R},$$

whose value $D(x)$ equals the distance from the optical center to the visible surface along the viewing ray that passes through x. Our goal is to recover a surface embedding

$$S : \Omega \subset \mathbb{R}^2 \longrightarrow \mathbb{R}^3$$

so that when S is *projected* through the known camera model $\mathcal{P}$ its image reproduces the data

$$\mathcal{P}\big(S\big) \approx I. \tag{1.1}$$

Approximation (1.1) defines the *forward* process, whereas the *inverse* problem—finding S given I (and possibly D) — is notoriously ill-posed: infinitely many geometries can cast the same colours onto the sensor. To select a unique, physically plausible solution, one must inject *priors*, such as smoothness, rigidity, or conformity to a statistical shape family learned from data. In practice, these preferences are enforced by augmenting the reconstruction objective with a penalty functional that steers the solution toward surfaces matching the chosen prior while remaining consistent with the visual evidence.

In the sequel we show how a curvature-based regularizer, combined with a data-driven depth prediction, yields a surface that not only fits each frame in a video stream but also remains stable over time, a property essential for reliable facial analysis, animation and authentication.

1.1 The state of the art

Research on 3-D reconstruction now coalesces around three methodological currents, each representing a different compromise between hardware complexity, data availability, and the geometric fidelity ultimately achievable. First, *multi-view stereo* (MVS) and *structure-from-motion* (SfM) exploit photometric consistency across several calibrated photographs: camera intrinsics and extrinsics are refined jointly with scene points via bundle adjustment, and dense depth is obtained by solving a variational optimization problem whose data term enforces brightness constancy while an epipolar prior constrains the search space [29]. When the imaging baseline is sufficiently wide and the scene is static, these pipelines routinely attain high accuracy; nevertheless, their reliance on multiple viewpoints makes real-time or hand-held operation onerous, and performance degrades sharply for non-Lambertian surfaces or dynamic content.

A second line of work replaces algorithmic inference with *active depth sensing*. Structured-light projectors, time-of-flight cameras, and lidar emit coded signals whose return delay or deformation yields a direct metric measurement of range [41]. By sidestepping the ill-posed inverse problem, such devices deliver per-pixel depth at video rates; yet they are not without drawbacks. Sensor-specific systematic errors, such as multi-path interference, motion blur, speckle noise arise under

strong ambient light, and the added cost, power consumption, and bulk of the hardware constrain deployment to indoor or robot-mounted platforms.

The third and now dominant current advocates *monocular, learning-based* reconstruction. Deep neural networks trained on extensive shape corpora internalise a statistical prior and map a single RGB frame to a surface either explicitly, by regressing vertex coordinates, or implicitly, by decoding a continuous signed-distance or radiance field [38]. These models dispense with auxiliary hardware and tolerate moderate non-rigid deformation, yet their predictions remain probabilistic: distributional shift, sensor noise, and out-of-catalogue poses manifest as holes, distorted silhouettes, and a loss of high-frequency detail, thereby limiting reliability in safety-critical or metrological contexts.

Collectively, these trajectories delineate a design spectrum whose extremes are geometry-rich but data-hungry MVS/SfM pipelines, hardware-intensive active sensors, and flexible yet artifact-prone monocular neural regressors. Their complementary strengths invite *hybrid* frameworks that harness deep priors for coarse shape estimation and subsequently impose variational regularity to guarantee geometric consistency—all while maintaining the low hardware footprint of monocular capture. The approach advanced in this paper occupies precisely this middle ground.

1.2 Motivation

Among all reconstructed objects, the human face occupies a singular position: small geometric details — wrinkles, pores, the bend of a lip—carry identity and expression, yet the surface is soft, constantly moving, and partly hidden by pose or lighting. Robust capture of such detail is indispensable for biometric authentication [13], believable telepresence avatars [43], and surgical planning where quantitative morphology must be preserved.

3-D morphable models (3DMM) tackle the problem by learning a low-dimensional shape basis from laser scans and fitting new faces with linear regression [6]. The approach is fast and robust but smooths away individual features. CNNs that output dense UV position maps preserve more detail and run in real time [14], yet their predictions inherit sensor noise and often fail under extreme pose or harsh illumination. Implicit neural representations—signed-distance or radiance fields—push fidelity even further [21], but their heavy optimization makes them impractical for video on everyday hardware.

A face-specific solution must therefore reconcile *three* competing objectives: stable tracking across an RGB stream, low computational cost, and guarantees against fake artifacts. These requirements point to *hybrid* methods. A fast CNN supplies an initial depth map, and a variational step then refines it: the data term keeps the learned global shape, while a curvature-based regularizer removes noise and densifies the mesh exactly where facial anatomy is most informative. The framework developed in the next sections shows that this combination can deliver real-time results without giving up either detail or mathematical soundness.

1.3 Contribution and novelty

The present work puts forward a combination of a lightweight, single-view depth prediction network with a classical, variational surface-smoothing step, yet still runs nearly in real time. The key idea is to regard the depth produced by a lightweight CNN not as ground truth but as a *soft* observation inside a biharmonic energy functional. In practice, this means we start from the coarse depth estimate given by the CNN and then smoothly adjust it so that small artifacts and noise are removed without losing important facial details such as the nose ridge, eyebrows, and lip contours.

To make this efficient, we build a single Delaunay-based triangular mesh for each video sequence and compute two fixed matrices that encode how to measure and penalize surface bending. Since these matrices depend only on the mesh connectivity (which never changes from frame to frame), we can factor them once and then reuse that factorization for every new frame. As a result, each time we get a new depth estimate from the network, we simply solve a very fast linear system to update the surface. Adding an extra term that links consecutive frames keeps the reconstruction temporally

coherent, and it only requires adjusting the right-hand side of our linear system—again, without refactoring.

The mesh creation itself is also carefully designed for speed and quality. We take the image rectangle and extend it into a prismatic volume based on a plausible depth range, then run a volumetric Delaunay tetrahedralization. From this tetrahedral mesh, we extract nearly flat boundary faces that face the camera. This "extruded Delaunay" construction produces triangles of good shape (avoiding overly sharp or skinny elements), which keeps our mass and stiffness matrices well conditioned even when the face is viewed at an angle. Finally, we run a quick, curvature-driven vertex adjustment that pushes mesh points toward areas with strong wrinkles or folds (such as around the eyes and mouth), automatically refining important features without slowing down the overall process.

In short, by combining the network's fast, per-pixel depth predictions with a single, pre-factored biharmonic smoothing step and an angle-controlled mesh, we achieve a pipeline that delivers smooth, plausible face reconstructions with configurable parameters. This design preserves genuine facial geometry better than simple smoothing or linear transformations, reduces noise and artifacts, and runs fast enough to be used in near real-time settings.

Hybrid approaches that fuse CNN-based depth with geometric regularizers do exist in adjacent domains such as real-time hand tracking [44] and large-scale urban modelling [26]. To our knowledge, however, no previous work targets *monocular, real-time face capture* with a pre-factored fourth-order biharmonic operator whose matrices stay fixed for the whole video; this combination of (i) single-factorisation, (ii) temporal reuse, and (iii) an angle-controlled extruded-Delaunay mesh is therefore the main technical novelty of our approach, rather than the introduction of an entirely new paradigm.

2 Problem formulation

Consider a video stream composed of a sequence of RGB images captured at discrete time steps $k = 1, \dots, N$. Each frame I_k is mathematically represented as a vector-valued function defined over a two-dimensional rectangular domain

$$I_k : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^3,$$

where $\Omega = [0, W] \times [0, H]$ denotes the set of pixel coordinates, $W, H \in \mathbb{N}$ are the image width and height *in pixels*, and the codomain $\mathbb{R}^3$ stores the red, green and blue intensities at every pixel location $x \in \Omega$.

A pretrained convolutional neural network (CNN)

$$\mathcal{P}_d : L^2(\Omega; \mathbb{R}^3) \longrightarrow L^2(\Omega)$$

is assumed *Lipschitz on bounded sets*; no property beyond measurability is needed in the analysis. It produces, for every frame, a dense depth map

$$D_k(x) = \mathcal{P}_d(I_k)(x), \quad D_k : \Omega \to \mathbb{R},$$

where $D_k(x)$ is the estimated distance from the camera centre to the first surface point intersected by the viewing ray through pixel x. Inevitable errors in D_k arise from image noise, shading ambiguities and network approximation.

In practice, $\mathcal{P}_d$ is realised by a lightweight U-Net–style CNN [27]; the Lipschitz assumption merely guarantees that D_k remains bounded in $L^2(\Omega)$ and is therefore harmlessly mild.

Camera geometry and pixel rays. Let $x = (u, v) \in \Omega$ denote discrete pixel coordinates (origin at the top–left of the image, u horizontal, v vertical). The k-th video frame is captured by a calibrated pin-hole camera whose *extrinsic* parameters are

$$R_k \in \big\{ R \in \mathbb{R}^{3\times 3} \mid R^\top R = I_3,\ \det R = 1 \big\},$$

$$\mathbf{o}_k \in \mathbb{R}^3,$$

where I_3 is the identity matrix 3×3 and $\mathbf{o}_k$ is the optical center in the world coordinates. Adopting the world-to-camera convention we store instead

$$\mathbf{t}_k := -R_k \mathbf{o}_k \in \mathbb{R}^3,$$

so every world point $\mathbf{X}^{(\mathrm{w})} \in \mathbb{R}^3$ maps to camera coordinates as

$$\mathbf{X}^{(k)}_{\mathrm{cam}} = R_k\big(\mathbf{X}^{(\mathrm{w})} - \mathbf{o}_k\big) = R_k \mathbf{X}^{(\mathrm{w})} + \mathbf{t}_k.$$

Thus $O_k = (0,0,0)^\top$ is the camera centre in its right-handed local frame, whose $+z$ axis points forward.

The *intrinsic* calibration matrix is

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3},$$

with focal lengths (f_x, f_y) (pixels) and principal point (c_x, c_y). Given a pixel $x = (u, v)$ we un-project the homogeneous image coordinate*

$$\tilde{\mathbf{d}}_k(x) = \mathbf{K}^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \in \mathbb{R}^3.$$

Normalising in the Euclidean norm $\|\mathbf{v}\|_2 := \sqrt{\mathbf{v}^\top \mathbf{v}}$ gives the unit direction

$$\boldsymbol{\rho}_k(x) = \frac{\tilde{\mathbf{d}}_k(x)}{\|\tilde{\mathbf{d}}_k(x)\|_2} \in \mathbb{S}^2 := \{\mathbf{v} \in \mathbb{R}^3 \mid \|\mathbf{v}\|_2 = 1\}. \tag{2.1}$$

The corresponding *viewing ray* is the half-line

$$\mathcal{R}_k(x) = \big\{ \mathbf{o}_k + \tau\, \boldsymbol{\rho}_k(x) \mid \tau \geq 0 \big\},$$

where τ is measured in meters.

All distance values are expressed in metres.

For the narrow field-of-view lenses used in our data, $\boldsymbol{\rho}_k(x) \approx \mathbf{e}_z := (0,0,1)^\top$; this simplification is *illustrative only* and is *never* used in the derivations that follow.

Our objective is to reconstruct a temporally coherent sequence of three-dimensional (3-D) surfaces, each described by a smooth embedding

$$S_k : \Omega \to \mathbb{R}^3, \quad k = 1, \ldots, N,$$

where $S_k(x) = (X_k(x), Y_k(x), Z_k(x))^\top$ returns the Euclidean coordinates of the reconstructed point corresponding to pixel x.

To achieve physically realistic and mathematically stable reconstructions, we impose three primary requirements on each surface S_k:

*All vectors are columns unless stated otherwise.

1. **Data fidelity.** For every pixel $\mathbf{x}\in\Omega$ the network predicts a depth $D_k(\mathbf{x})$, namely the Euclidean distance from the camera centre O_k to the first visible surface point along the unit ray $\boldsymbol{\rho}_k(\mathbf{x})$ of (2.1). Given a candidate surface embedding $S_k : \Omega\to\mathbb{R}^3$ we decompose

$$S_k(\mathbf{x}) = \hat{d}_k(\mathbf{x})\,\boldsymbol{\rho}_k(\mathbf{x}) + \mathbf{t}_k(\mathbf{x}),$$

where

$$\hat{d}_k(\mathbf{x}) := (S_k(\mathbf{x}), \boldsymbol{\rho}_k(\mathbf{x}))_{\mathbb{R}^3}, \qquad \mathbf{t}_k(\mathbf{x}) \perp \boldsymbol{\rho}_k(\mathbf{x}),$$

with $(\cdot,\cdot)_{\mathbb{R}^3}$ denoting the standard Euclidean inner product. The scalar $\hat{d}_k$ is therefore the *radial component* (*i.e.* the orthogonal projection of S_k onto the view ray), whereas $\mathbf{t}_k$ is the *tangential component*.

In the noise-free case one would have $S_k(\mathbf{x}) = D_k(\mathbf{x})\,\boldsymbol{\rho}_k(\mathbf{x})$. We penalise the full vector mis-fit

$$E_{\text{data}}(S_k) = \int_\Omega \big\| S_k(\mathbf{x}) - D_k(\mathbf{x})\,\boldsymbol{\rho}_k(\mathbf{x}) \big\|_2^2 \, d\mathbf{x}. \tag{2.2}$$

Since $\mathbf{t}_k(\mathbf{x})$ is orthogonal to the ray,

$$\big\| S_k - D_k\boldsymbol{\rho}_k \big\|_2^2 = \big(\hat{d}_k - D_k\big)^2 + \|\mathbf{t}_k\|_2^2.$$

For a pin-hole camera the true surface has $\mathbf{t}_k \equiv 0$, and in practice our smoothness term drives $\mathbf{t}_k$ rapidly to negligible values. Hence we may write

$$E_{\text{data}}(S_k) \;\approx\; \int_\Omega \big(\hat{d}_k(\mathbf{x}) - D_k(\mathbf{x})\big)^2 \, d\mathbf{x}, \tag{2.3}$$

which highlights that the integrand is simply the squared depth error in metres.[†] This observation matches the data model used in volumetric depth-fusion and ray-aligned signed-distance methods [10,25]. Using $\boldsymbol{\rho}_k(\mathbf{x})$ avoids projecting every depth sample onto a single axis; Eq. (2.2) coincides with the observation model in volumetric depth fusion and ray-aligned signed-distance methods.

2. **Geometric smoothness.** To suppress artificial bumps and pixel-scale ripples we penalise curvature

$$E_{\text{smooth}}(S_k) = \int_\Omega \big\| \Delta S_k(x) \big\|^2 \, dx, \tag{2.4}$$

with Δ the component-wise Laplace–Beltrami operator. This biharmonic term is widely used in mesh fairing and detail-preserving editing [30]; it enforces $S_k \in H^2(\Omega;\mathbb{R}^3)$, i.e. square-integrable second derivatives.

3. **Temporal coherence.** Neighbouring frames should differ only by true facial motion, not by random jitters

$$E_{\text{temp}}(S_k, S_{k-1}) = \int_\Omega \big\| S_k(x) - S_{k-1}(x) \big\|^2 \, dx. \tag{2.5}$$

The same L^2 norm is routinely adopted in real-time RGB-D trackers (e.g. DynamicFusion and successors [39]) because it penalises mean-squared vertex displacement without freezing genuine expression changes.

[†]Expressions (2.2) and (2.3) produce identical gradients once $\mathbf{t}_k \to 0$; the implementation keeps the full form, while (2.3) is shown for interpretability.

Throughout, we abbreviate the iterated Laplacian by $\Delta^2 S_k := \Delta\big(\Delta S_k\big)$.

We have now specified three *framewise* cost terms: $E_{\text{data}}(S_k)$ in (2.2), $E_{\text{smooth}}(S_k)$ in (2.4), and $E_{\text{temp}}(S_k, S_{k-1})$ in (2.5). Following the standard variational nomenclature (e.g. [28]), we call each such scalar cost an *energy*. We therefore combine the three energies into the functional

$$\mathcal{E}(\{S_k\}_{k=1}^N) = \sum_{k=1}^{N} \Big[E_{\text{data}}(S_k) + \lambda\, E_{\text{smooth}}(S_k) + \mu\, E_{\text{temp}}(S_k, S_{k-1}) \Big],$$

where $\lambda > 0$ controls spatial smoothness, $\mu > 0$ governs temporal regularity, and $E_{\text{temp}}(S_1, S_0)$ is initialised with $S_0 = S_1$ to close the sum.

Physical rationale. A human face seldom intersects the image border; geometry projecting onto $\partial\Omega$ is therefore unknown and should be left unconstrained. We model this by forcing *zero normal flux* of both the surface and its bending forces,

$$\nabla S_k \cdot \nu \;=\; 0 \quad \text{on } \partial\Omega, \tag{2.6}$$

$$\nabla(\Delta S_k) \cdot \nu \;=\; 0 \quad \text{on } \partial\Omega. \tag{2.7}$$

Here ν is the outer unit normal on $\partial\Omega$. Equations (2.6)–(2.7) are the classical *free–edge* Neumann boundary conditions for a thin Kirchhoff plate. Equation (2.6) annihilates the *normal slope*, preventing the reconstructed mesh from tilting through the image border, whereas (2.7) sets the *bending moment* to zero, so the biharmonic regularizer cannot pull vertices across $\partial\Omega$.

With (2.6)-(2.7) built in, we work in the Hilbert space

$$V := H^2_{\text{N}}\big(\Omega;\mathbb{R}^3\big) := \Big\{ u \in H^2\big(\Omega;\mathbb{R}^3\big) \;\Big|\; \nabla u \cdot \nu = 0 \text{ on } \partial\Omega \Big\}$$

i.e. the Sobolev space H^2 augmented with the essential (*Neumann-type*) boundary condition $\nabla u \cdot n = 0$. The natural condition $\nabla(\Delta u)\cdot \nu = 0$ required by the weak formulation is not imposed at the space level; it is automatically satisfied by any minimiser of the biharmonic energy.

Theorem 2.1 (Equivalence of the graph norm) *Let $\Omega \subset \mathbb{R}^2$ be bounded and either convex or of class $C^{1,1}$. Define the free–edge Neumann space*

$$H^2_{\text{N}e}(\Omega) := \Big\{ u \in H^2(\Omega) \;\Big|\; \nabla u \cdot \nu = 0 \text{ on } \partial\Omega \Big\},$$

where ν is the unit outward normal. There exist positive constants α_0, β_0, depending only on Ω, such that

$$\alpha_0 \,\|u\|^2_{H^2(\Omega)} \;\le\; \|u\|^2_{L^2(\Omega)} + \|\Delta u\|^2_{L^2(\Omega)} \;\le\; \beta_0 \,\|u\|^2_{H^2(\Omega)}, \qquad \forall\, u \in H^2_{\text{N}e}(\Omega). \tag{2.8}$$

Consequently, the graph norm $\|u\|^2_{L^2(\Omega)} + \|\Delta u\|^2_{L^2(\Omega)}$ is equivalent to the full H^2–norm on $H^2_{\text{N}e}(\Omega)$.

Proof. For $u \in H^2(\Omega)$ the continuous embeddings $H^2(\Omega) \hookrightarrow H^1(\Omega) \hookrightarrow L^2(\Omega)$ give

$$\|u\|_{L^2(\Omega)} \;\le\; \beta_1 \,\|u\|_{H^2(\Omega)}.$$

Furthermore, the definition of the H^2–seminorm yields $\|\Delta u\|_{L^2} \le \|\mathcal{H} u\|_{L^2} \le \|u\|_{H^2}$. Thus

$$\|u\|^2_{L^2} + \|\Delta u\|^2_{L^2} \;\le\; (\beta_1^2 + 1)\,\|u\|^2_{H^2} =: \beta_0 \,\|u\|^2_{H^2}.$$

Because $\nabla u \cdot \nu = 0$ on $\partial\Omega$, integration by parts removes the boundary term in Green's identity, giving $\|\nabla^2 u\|^2_{L^2} = \|\Delta u\|^2_{L^2}$. The interior regularity estimate for the *Neumann biharmonic* problem ([11, Ch. 5], [1]) asserts that there exists a constant $\beta_2 > 0$, depending only on Ω, such that

$$\|u\|_{H^2(\Omega)} \;\le\; \beta_2 \big(\|u\|_{L^2(\Omega)} + \|\Delta u\|_{L^2(\Omega)} \big), \qquad \forall u \in H^2_{\text{N}e}(\Omega).$$

Squaring this inequality and setting $\alpha_0 := \beta_2^{-2}$ yields the left-hand estimate in (2.8). □

We equip V with the graph norm

$$\|u\|_V^2 := \|u\|_{L^2(\Omega)}^2 + \|\Delta u\|_{L^2(\Omega)}^2, \qquad u \in V.$$

which is a norm on $H^2_{\mathrm{Ne}}(\Omega)$ by Theorem 2.1.

For fixed parameters $\lambda > 0$, $\mu > 0$ we introduce the single–frame bilinear form

$$a_{\lambda,\mu}(u,v) := \int_\Omega \Big((1+\mu)\, u\cdot v + \lambda\, \Delta u \cdot \Delta v \Big)\, dx, \qquad u, v \in V.$$

The bilinear form contains exactly the two quadratic quantities that define the graph norm – the L^2 inner-product of u and the L^2 inner-product of Δu. We have

$$a_{\lambda,\mu}(u,u) \;=\; (1+\mu)\,\|u\|_{L^2(\Omega)}^2 + \lambda\,\|\Delta u\|_{L^2(\Omega)}^2 \;\geq\; c\,\|u\|_V^2,$$

with $c = \min\{\lambda, 1+\mu\} > 0$. Hence $a_{\lambda,\mu}$ is continuous and coercive on V.

Having fixed the single-frame space $V := H^2_{\mathrm{Ne}}(\Omega;\mathbb{R}^3)$, we extend it to the entire video by taking the N-fold Cartesian product of V with itself

$$\mathcal{V} := \underbrace{V \times V \times \cdots \times V}_{N \text{ times}} \;=\; \big\{ (S_1, \dots, S_N) \mid S_k \in V \;\forall k \big\}.$$

We equip $\mathcal{V}$ with the norm

$$\|\mathbf{S}\|_{\mathcal{V}}^2 := \sum_{k=1}^{N} \Big(\|S_k\|_{L^2(\Omega)}^2 + \|\Delta S_k\|_{L^2(\Omega)}^2 \Big).$$

Here each element $\mathbf{S} \in \mathcal{V}$ is the tuple $\mathbf{S} = (S_1, \dots, S_N)$.

For an arbitrary perturbation $\mathbf{\Phi} = (\Phi_1, \dots, \Phi_N) \in \mathcal{V}$ define the one–parameter family $\mathbf{S}_\varepsilon = \mathbf{S} + \varepsilon\mathbf{\Phi}$. Because every term in the discrete energy $\mathcal{E}(\mathbf{S}) = \sum_{k=1}^N \big[E_{\mathrm{data}}(S_k) + \lambda E_{\mathrm{smooth}}(S_k) + \mu E_{\mathrm{temp}}(S_k, S_{k-1}) \big]$ is quadratic, differentiation under the integral sign yields

$$\frac{d}{d\varepsilon}\,\mathcal{E}(\mathbf{S}_\varepsilon)\Big|_{\varepsilon=0} = \sum_{k=1}^{N} \Big\langle\, S_k - D_k\boldsymbol{\rho}_k + \lambda\,\Delta^2 S_k + \mu\,(2S_k - S_{k-1} - S_{k+1}),\; \Phi_k \Big\rangle_{L^2(\Omega)}. \tag{2.9}$$

The right–hand side is a bounded linear functional of $\mathbf{\Phi}$; indeed, by Cauchy–Schwarz and Young's inequalities,

$$\big|\, \langle \cdot, \Phi_k \rangle_{L^2} \big| \leq \big(\|S_k\|_V + \|D_k\|_{L^2} \big)\, \|\Phi_k\|_V, \quad k = 1, \dots, N,$$

so that

$$\big|\, \mathcal{E}'(\mathbf{S})[\mathbf{\Phi}] \,\big| \;\leq\; \vartheta\, \|\mathbf{\Phi}\|_{\mathcal{V}}, \qquad \vartheta \;=\; c^{-1} \Big(\|\mathbf{S}\|_{\mathcal{V}} + \max_{1\leq k\leq N} \|D_k\|_{L^2(\Omega)} \Big),$$

with the coercivity constant $c = \min\{1+\mu, \lambda\} > 0$. Having $\mathcal{V}'$ – the Hilbert–space dual of $\mathcal{V}$, the estimate above shows $\|\mathcal{E}'(\mathbf{S})\|_{\mathcal{V}'} \leq \vartheta < \infty$ and the constant ϑ depends continuously on $\mathbf{S}$, the mapping $\mathbf{S} \mapsto \mathcal{E}'(\mathbf{S})$ is continuous from $\mathcal{V}$ to its dual $\mathcal{V}'$. Hence $\mathcal{E}$ is Fréchet differentiable on $\mathcal{V}$.

Requiring (2.9) to vanish for every $\Phi_k \in V$ gives, frame by frame,

$$\Big\langle\, S_k - D_k\boldsymbol{\rho}_k + \lambda\,\Delta^2 S_k + \mu\,(2S_k - S_{k-1} - S_{k+1}),\; \Phi_k \Big\rangle_{L^2(\Omega)} \;=\; 0 \qquad (\forall\, \Phi_k \in V). \tag{2.10}$$

Two integrations–by–parts, together with the free–edge Neumann conditions already encoded in V, convert (2.10) into the strong fourth–order PDE

$$\big(I - \lambda\Delta^2\big) S_k + \mu\,(2S_k - S_{k-1} - S_{k+1}) \;=\; D_k\boldsymbol{\rho}_k \quad \text{in } \Omega, \qquad k = 1, \dots, N. \tag{2.11}$$

The operator $I - \lambda\Delta^2$ is uniformly elliptic of order 4 and the tridiagonal temporal coupling does not affect spatial ellipticity.

Introduce the symmetric bilinear form

$$a(\mathbf{U},\mathbf{V}) := \sum_{k=1}^{N}\int_\Omega \Big[(1+\mu)\,U_k\cdot V_k + \lambda\,\Delta U_k\cdot\Delta V_k - \mu\,U_{k-1}\cdot V_k - \mu\,U_k\cdot V_{k-1}\Big]dx, \qquad U_0 := U_N. \tag{2.12}$$

The diagonal part reproduces the single–frame form $a_{\lambda,\mu}(u,v)$. The single–frame coercivity estimate $a_{\lambda,\mu}(u,u) \geq c\|u\|_V^2$ passes to $a(\mathbf{U},\mathbf{U}) \geq c\|\mathbf{U}\|_{\mathcal{V}}^2$ for the same $c = \min\{1+\mu,\lambda\}$, so a is continuous and coercive on $\mathcal{V}$.

The right–hand side of (2.11) induces the linear functional

$$\ell(\mathbf{V}) \;:=\; \sum_{k=1}^{N}\int_\Omega D_k\boldsymbol{\rho}_k\cdot V_k\,dx, \qquad \mathbf{V}\in\mathcal{V},$$

which is bounded on $\mathcal{V}$ because $|\ell(\mathbf{V})| \leq \big(\max_k \|D_k\|_{L^2(\Omega)}\big)\sum_{k=1}^N \|V_k\|_{L^2(\Omega)} \leq \max_k \|D_k\|_{L^2}\,\sqrt{N}\,\|\mathbf{V}\|_{\mathcal{V}}$.

For $\mathbf{U}\in\mathcal{V}$ the bilinear form (2.12) splits into

$$a(\mathbf{U},\mathbf{U}) = \sum_{k=1}^{N}\Big[(1+\mu)\|U_k\|_{L^2}^2 + \lambda\|\Delta U_k\|_{L^2}^2\Big] \;-\; \mu\sum_{k=1}^{N}\langle U_{k-1},U_k\rangle_{L^2}.$$

Applying Cauchy–Schwarz inequality to the off–diagonal term, $|\langle U_{k-1},U_k\rangle_{L^2}| \leq \frac{1}{2}\big(\|U_{k-1}\|_{L^2}^2 + \|U_k\|_{L^2}^2\big)$, and summing over k gives

$$a(\mathbf{U},\mathbf{U}) \;=\; \sum_{k=1}^{N}\Big[\,\|U_k\|_{L^2}^2 + \lambda\,\|\Delta U_k\|_{L^2}^2\Big] \;\geq\; \min\{1,\lambda\}\,\|\mathbf{U}\|_{\mathcal{V}}^2.$$

Hence a is coercive with the constant $c := \min\{1,\lambda\} > 0$.

By Lax–Milgram lemma there exists a unique $\mathbf{S}^*\in\mathcal{V}$ such that $a(\mathbf{S}^*,\mathbf{V}) = \ell(\mathbf{V})$ for every $\mathbf{V}\in\mathcal{V}$. Choosing $\mathbf{V} = \mathbf{S}^*$ yields the *energy identity*

$$a(\mathbf{S}^*,\mathbf{S}^*) \;=\; \ell(\mathbf{S}^*) \;\leq\; \max_{1\leq k\leq N}\|D_k\|_{L^2(\Omega)}\,\sqrt{T}\,\|\mathbf{S}^*\|_{\mathcal{V}}, \tag{2.13}$$

where the factor $\sqrt{N}$ comes from Cauchy–Schwarz inequality

$$\sum_{k=1}^{N}\|D_k\|_{L^2}\|S_k^*\|_{L^2} \leq \Big(\max_k\|D_k\|_{L^2}\Big)\sqrt{\sum_{k=1}^{N}\|S_k^*\|_{L^2}^2} \leq \max_k\|D_k\|_{L^2}\sqrt{N}\,\|\mathbf{S}^*\|_{\mathcal{V}}.$$

Dividing (2.13) by the coercivity constant c yields the *a-priori* bound

$$\|\mathbf{S}^*\|_{\mathcal{V}} \;\leq\; \frac{\sqrt{N}\;\max\limits_{1\leq k\leq N}\|D_k\|_{L^2(\Omega)}}{\min\{1,\lambda\}}. \tag{2.14}$$

The numerator aggregates the worst–case sensor error over the N frames; the denominator shows how stronger spatial smoothing (λ) or temporal coupling (μ) tightens the bound.

The hyper-parameters are obtained in a reproducible two–step protocol.

1. *Temporal pass.* Set $\lambda = 1$. Increase μ on a short calibration sequence until visual inspection reveals no frame-to-frame jitter (empirically $\mu \approx 5$).

2. *Global scaling.* Multiply both λ and μ by a scalar $\gamma > 0$ chosen so that the residual $r_k := S_k^* - D_k\boldsymbol{\rho}_k$ satisfies $\|r_k\|_{L^2(\Omega)} \approx \sigma_{\text{noise}}$, where σ_{noise} is the background depth–sensor standard deviation measured once for the entire data set. In all reported experiments $\lambda = 2.5$ and $\mu = 12.5$.

Theorem 2.2 (Global existence, uniqueness and stability) *Let $\lambda > 0$, $\mu > 0$ and $D_k \in L^2(\Omega)$ for $k = 1, \dots, N$. Then the bilinear form $a(\cdot,\cdot)$ is continuous and c-coercive on $\mathcal{V}$ with $c = \min\{\lambda, 1\} > 0$;; the variational problem $a(\mathbf{S}, \mathbf{V}) = \ell(\mathbf{V}) \quad \forall\, \mathbf{V} \in \mathcal{V}$ admits a* unique *solution $\mathbf{S}^* \in \mathcal{V}$; the solution satisfies the a priori estimate* (2.14)*; the Hessian operator $\mathcal{A}\colon \mathcal{V} \to \mathcal{V}'$, defined by $\langle \mathcal{A}\mathbf{U}, \mathbf{V}\rangle = a(\mathbf{U}, \mathbf{V})$, is block-tridiagonal, self-adjoint and positive-definite, and satisfies $\sigma(\mathcal{A}) \subset [c,\ 1+\mu+\lambda\,\Lambda_{\max}^2]$, where $\Lambda_{\max}$ is the largest (discrete) Neumann Laplace–Beltrami eigenvalue on the triangulated domain.*

Proof sketch. (a) The Cauchy-Schwarz inequality immediately yields the continuity of $a(\cdot,\cdot)$ on $\mathcal{V}$. Under the *graph norm* $\|\mathbf{U}\|_{\mathcal{V}}^2 := \sum_{k=1}^{N}\big(\|U_k\|_{L^2}^2 + \|\Delta U_k\|_{L^2}^2\big)$ we have

$$a(\mathbf{U},\mathbf{U}) = \sum_{k=1}^{N}\Big[(1+\mu)\|U_k\|_{L^2}^2 + \lambda\|\Delta U_k\|_{L^2}^2\Big] - \mu\sum_{k=1}^{N}\langle U_{k-1}, U_k\rangle_{L^2} \;\geq\; c\,\|\mathbf{U}\|_{\mathcal{V}}^2, \qquad c := \min\{1,\lambda\} > 0.$$

Hence a is coercive in $\mathcal{V}$ by definition of the graph norm.

(b)–(c) The Lax–Milgram lemma on the product space $\mathcal{V} = [H^2_{Ne}]^N$ yields existence, uniqueness and the stability estimate (2.14). Testing the Galerkin equation with $\mathbf{V} = \mathbf{S}^*$ and using coercivity gives exactly (2.14); the $\sqrt{N}$ factor is the Cauchy–Schwarz inequality bound for the sum of N framewise residuals.

(d) Writing the operator in block form

$$\mathcal{A} \;=\; \begin{pmatrix} B & -\mu I & & \\ -\mu I & B & \ddots & \\ & \ddots & \ddots & -\mu I \\ & & -\mu I & B \end{pmatrix}, \qquad B := (1+\mu)I - \lambda\Delta^2$$

reveals a Hermitian block-Toeplitz structure. Let $\{-\Delta\varphi_j = \Lambda_j\varphi_j\}_{j\geq 0}$ be the Neumann eigenpairs of the Laplace–Beltrami operator ($\Lambda_0 = 0 < \Lambda_1 \leq \Lambda_2 \leq \dots$). Because $\Delta^2\varphi_j = \Lambda_j^2\varphi_j$, each block B is diagonalised by the same eigenfunctions and

$$\sigma(B) = \Big\{\, 1 + \mu - \lambda\Lambda_j^2 \;:\; j \geq 0 \Big\} \subset \big[\,1 + \mu - \lambda\Lambda_{\max}^2,\ 1+\mu\big].$$

Applying Weyl's theorem for block Toeplitz matrices [33, Thm. 4.3], the spectrum of $\mathcal{A}$ is contained in $[\,\min\sigma(B) - \mu,\ \max\sigma(B) + \mu\,]$, i.e. in $[c,\ 1 + \mu + \lambda\Lambda_{\max}^2]$ because $\mu < c$. Positive-definiteness then follows from $c > 0$. □

Consequence for discretisation. The symmetry and positive definiteness established in Theorem 2.2(d) are inherited *verbatim* by the finite-element Galerkin system obtained in Section 4, because the mesh Laplacian is an L^2–projection of Δ. In particular the condition number grows at most like h^{-2}, so classical multigrid or conjugate-gradient solvers achieve mesh-independent convergence rates [11, Sec. 11.3].

3 Landmark extraction and refinement

The biharmonic reconstruction of Section 2 treats every pixel with equal confidence. In practice, however, the silhouette, eye corners, nose tip and lip commissures carry far more geometric information

than uniform skin areas. Pre-aligning those *semantic anchors* therefore serves two purposes: it tightens the data term, so the subsequent variational solver converges in fewer iterations, and it establishes a consistent correspondence across frames, which stabilises the temporal coherence term. We obtain the anchors in three stages: classical scale–space keypoint detection, modern CNN-based landmark regression, and a weighted 3-D Morphable-Model (3-DMM) fit that reconciles the two.

Reliable localisation of salient facial features is the first data–dependent operation that follows the variational formulation of Section 2. Two complementary traditions coexist. Classical keypoint detectors such as SIFT [20] and SURF [4] search for photometric extrema in scale space, whereas modern convolutional networks directly regress a semantically organised set of fiducial points. Our pipeline harnesses the best of both worlds: a CNN supplies a compact, high-confidence landmark set, which is then regularized by a statistical face model to align perfectly with the surface variables introduced earlier.

Classical scale–space detectors. SIFT flags potential keypoints as spatial–scale extrema of the *difference–of–Gaussians*

$$\mathrm{DoG}(x,\sigma) \;=\; G(x,k\sigma)I \;-\; G(x,\sigma)I, \tag{3.1}$$

where $I:\Omega\to\mathbb{R}$ is the grey-scale image, $G(x,\sigma)=\frac{1}{2\pi\sigma^2}\exp(-\|x\|^2/2\sigma^2)$ is a 2-D Gaussian of standard deviation $\sigma>0$, $k>1$ is a fixed scale-spacing factor, and "$*$" denotes convolution. The subtraction isolates band-pass structure, so local extrema of (3.1) are invariant to uniform illumination changes.

For each scale–space extremum $x^\star$ SIFT selects a *dominant orientation*. All notation follows Lowe [20]; the implementation details given here reproduce the reference code in `opencv` [4].

$$\theta^\star \;=\; \operatorname*{arg\,max}_{\theta\in[0,2\pi)} \sum_{x'\in\mathcal{N}(x^\star)} w(x')\,\big\|\nabla I(x')\big\|\,\delta\Big(\theta-\operatorname{atan2}\big(I_y(x'),\,I_x(x')\big)\Big),$$

where $w(x')=\exp\big(-\|x'-x^\star\|^2/(2\sigma^2)\big)$ weights samples by a Gaussian of standard deviation σ; δ denotes the *Dirac distribution*. In practice, since the sum runs over discrete pixels, $\delta(\theta-\theta')$ is replaced by a Kronecker delta that increments only the nearest of 36 orientation bins (bin width 10°); $\mathcal{N}(x^\star)$ is the circular neighbourhood $\{\,x' \mid \|x'-x^\star\|<3\sigma\}$.

The maximiser $\theta^\star$ is the mode of this weighted histogram. Rotating the local patch by $-\theta^\star$ aligns the dominant gradient with the positive x-axis, conferring *rotation invariance*.

Spatial pooling. The $16\sigma\times16\sigma$ rotated patch is partitioned into a 4×4 grid of square cells (each 4σ wide). Within every cell we accumulate gradient magnitudes into an *eight-bin* orientation histogram centred at $0^\circ,45^\circ,\dots,315^\circ$. Concatenating the 8-component histogram of each of the $4\times4=16$ cells yields a vector in $\mathbb{R}^{128}$. Finally the vector is Euclidian-normalised, and its entries are clipped to 0.2 before a second normalisation, improving robustness to illumination changes [20].

SURF [4] accelerates this pipeline by replacing Gaussian derivatives with box filters evaluated via integral images; the resulting Hessian approximation

$$H(x,\sigma) \;\approx\; \begin{pmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{pmatrix},$$

where $L_{pq}(x,\sigma)$ denotes a box-filtered second derivative of I in directions $p,q\in\{x,y\}$, is evaluated in $O(1)$ time per pixel, reducing the cost by roughly a factor of two while producing 64-dimensional descriptors with slightly lower repeatability.

Although both detectors return hundreds of points on a 256×256 portrait, only a few dozen semantically meaningful landmarks are required for robust 3-D face reconstruction.

Deep-learning detectors predict a small set $\mathcal{L}_k=\{(\ell_i,w_i)\}_{i=1}^{N}$ of 2-D landmark positions $\ell_i\in\Omega$ and confidences $w_i\in(0,1]$ for every video frame I_k. Modern examples include MTCNN [42], BlazeFace [5], and MediaPipe FaceMesh [19] return $N=5$, $N=6$ and $N=468$ points respectively. We adopt MediaPipe and keep the $N=468$.

The running time of the subsequent refinement obeys

$$T_{\text{refine}} \;=\; \underbrace{\mathcal{O}(N \log N)}_{\text{landmark indexing}} \;+\; \underbrace{\mathcal{O}(n)}_{\text{surface update}} \quad ,$$

where n is the number of mesh vertices carried by the variational solver; the first term stems from a balanced KD-tree built once per frame, the second from a single sparse linear solve.

Let $\mathcal{L}_k = \{(\ell_i, w_i)\}_{i=1}^{N}$ be the 2-D landmarks at video time k, with confidences $w_i \in (0, 1]$. We lift each ℓ_i to a 3-vector in $\mathbb{R}^{3n}$: positions corresponding to ℓ_i receive the 2-D coordinate while all other entries are zero. Collecting them yields the *sparse* vector $L_k \in \mathbb{R}^{3n}$, and the diagonal confidence matrix $W_k = \operatorname{diag}(w_1, \ldots, w_N) \in \mathbb{R}^{N \times N}$. Because outliers and tracking jitter affect different landmarks unequally, Yang *et al.* [40] showed that weighting each pair individually improves accuracy; our W_k follows that principle.

We estimate $\mathbf{p}_k$ by the Tikhonov-regularised weighted least squares problem (sometimes called *ridge regression*) [17, 32]

$$\mathbf{p}_k^{*} \;=\; \arg\min_{\mathbf{p} \in \mathbb{R}^m} \Big\| W_k^{1/2} \big(\bar{S} + B\mathbf{p} - L_k\big) \Big\|_2^2 + \alpha \, \|\mathbf{p}\|_2^2, \tag{3.2}$$

where $\alpha > 0$ trades data fidelity against the Gaussian prior $\mathbf{p} \sim \mathcal{N}(\mathbf{0}, \alpha^{-1} I_m)$ (c.f. the Bayesian view of Tikhonov [17, Sec. 2]).

Expanding (3.2) and taking the gradient in $\mathbf{p}$ gives

$$\nabla_{\mathbf{p}} E = 2B^{\top} W_k B \, \mathbf{p} - 2B^{\top} W_k \big(L_k - \bar{S}\big) + 2\alpha \mathbf{p}.$$

Setting $\nabla_{\mathbf{p}} E = \mathbf{0}$ yields the (weighted) normal equations

$$\big(B^{\top} W_k B + \alpha I_m\big) \, \mathbf{p}_k^{*} \;=\; B^{\top} W_k \big(L_k - \bar{S}\big), \tag{3.3}$$

where $W_k = I$ [2], $B \in \mathbb{R}^{3n \times m}$ is the morphable-model basis, $L_k \in \mathbb{R}^{3n}$ the stacked observed landmark coordinates, $\bar{S} \in \mathbb{R}^{3n}$ the mean-shape vector, I_m the $m \times m$ identity matrix, and $\alpha > 0$ (we take $\alpha = 10^{-3}$), so α that trades off landmark fidelity against the preference for small deformations. Because $B^{\top} W_k B$ is positive-semidefinite and αI_m is strictly positive, the sum is *symmetric positive definite* (SPD); hence (3.3) admits the unique solution

$$\mathbf{p}_k^{*} = \big(B^{\top} W_k B + \alpha I_m\big)^{-1} B^{\top} W_k \big(L_k - \bar{S}\big).$$

The system is factorised once by a dense Cholesky decomposition, an $\mathcal{O}(m^3)$ operation. For the typical dimension $m = 80$ this amounts to about half a micro second on the tested hardware (see [8, p. 7]). As this cost is already negligible relative to the per–frame computations, more elaborate solvers (e.g. the accelerated routine of [35]) were not pursued.

The shape $S_k^0 := \bar{S} + B\mathbf{p}_k^{*}$ acts as a *regularised, landmark-consistent prior* for the variational solver of Section 2.

4 Planar triangulation and Delaunay criterion

Since the bilinear form $a(\cdot, \cdot)$ is continuous and V-coercive by Theorem 2.2 (a), the weak problem (2.9)–(2.11) satisfies the hypotheses of Céa's lemma. It thus admits a conforming $\mathcal{P}^1$ Galerkin realisation on any shape-regular mesh $\mathscr{T}_h \subset \overline{\Omega}$. We now detail two *geometrically equivalent* ways of constructing the simplicial complex $\mathscr{T}_h$ on which that discretisation is carried out.

(i) Planar approach. Let $P = \{p_i = (x_i, y_i)\}_{i=1}^{n} \subset \Omega$ be the sub-pixel centres with p_1, p_2, p_3, p_4 anchored to the corners of $\Omega = [0, W] \times [0, H]$. A *constrained* Delaunay triangulation $\mathcal{T}_h$ maximises

the smallest interior angle subject to the empty-circumcircle predicate

$$\det \begin{pmatrix} x_i & y_i & x_i^2+y_i^2 & 1 \\ x_j & y_j & x_j^2+y_j^2 & 1 \\ x_k & y_k & x_k^2+y_k^2 & 1 \\ x_\ell & y_\ell & x_\ell^2+y_\ell^2 & 1 \end{pmatrix} \le 0, \quad \begin{array}{l} \forall \tau = (p_i,p_j,p_k) \in \mathcal{T}_h, \\ p_\ell \in P \setminus \tau, \end{array} \tag{4.1}$$

where "≤ 0" guarantees that no point lies *inside* $\mathrm{circ}(\tau)$. The boundary $\partial\Omega$ is respected by treating its four edges as *segment constraints* that are never flipped. Incremental vertex insertion plus local edge-flips (Alg. 1) restores (4.1) in $O(n \log n)$ time [9,37]. The result is a planar, shape-regular mesh whose element diameter satisfies $h = \max_{T\in\mathcal{T}_h} \mathrm{diam}(T)$, where T always denotes a single triangle of the triangulation $\mathcal{T}_h$.

Algorithm 1 Incremental constrained Delaunay triangulation

Require: point cloud P, constraint edges $E \subset P \times P$
Ensure: Delaunay mesh $\mathcal{T}_h$
$\mathcal{T}_h \leftarrow \varnothing$
for all $p \in P$ **do** ▷ vertex insertion [31]
    insert p and edge-flip until (4.1) holds locally
end for
for all $(p_i,p_j) \in E$ **do** ▷ polyline recovery
    insert segment, edge-flip until local Delaunay property is restored
end for
return $\mathcal{T}_h$

(ii) Volumetric approach. Let

$$Z_{\max} \;:=\; \big(\max_{1\le k\le N} \max_{x\in\Omega} D_k(x)\big) \;+\; \delta, \qquad \delta > 0,$$

be a global upper bound on the depth range observed in the entire video. Depth ranges exceeding one metre distort the planar mesh when projected into $\mathbb{R}^3$. We therefore extrude Ω into the right prism $\Pi = \Omega \times [0, Z_{\max}]$ and compute a *constrained three-dimensional* Delaunay tetrahedralisation

$$\mathcal{K}_h = \mathrm{CDT}\big(\Pi \cup \Gamma; \alpha\big), \tag{4.2}$$

where Γ contains the four vertical faces above the image border and $\alpha > 0$ is an *alpha-value* in the sense of α-shapes [34]. In practice $\alpha = 1.4\, h_{\text{2-D}}$ efficiently deletes sliver tetrahedra whose circumsphere radius exceeds α. CGAL's exact `Delaunay_tetrahedralization_3` routine realises (4.2) with the same $O(n \log n)$ complexity as the planar case.

From the boundary $\partial\mathcal{K}_h$ we extract only those triangular facets whose outward unit normal n_T obeys the quasi-planarity filter

$$|n_T \cdot e_z| \;\le\; \eta, \qquad \eta \simeq 0.10,$$

so that each selected face is almost orthogonal to the optical axis e_z. The collection $\mathscr{T}_h := \{T \subset \partial\mathcal{K}_h \mid |n_T \cdot e_z| \le \eta\}$ projects one-to-one onto Ω and is therefore homeomorphic to the planar triangulation; the 2-D vertex set $\{p_i\}$ is *unchanged*, only the connectivity differs. Consequently the nodal basis $\{\phi_i\}$ is valid on either complex, and the surface still inherits the volumetric Delaunay *angle-optimality*

$$\frac{\rho_T}{h_T} \ge \rho_{\min}(\alpha,\eta) > 0, \quad \forall T \in \mathscr{T}_h,$$

where ρ_T and h_T are the in-radius and longest edge of T and $\rho_{\min}$ is the quality constant [34]. Consequently, the extracted surface is free of the highly obtuse, near-degenerate facets that commonly arise when perspective back-projection is applied to wide-baseline imagery.

Finite-element space. Either construction delivers a shape-regular simplicial complex $\mathscr{T}_h$ and its nodal $\mathcal{P}^1$ space

$$V_h = \big\{ v_h \in C^0(\overline{\Omega}; \mathbb{R}^3) \;\big|\; v_h|_T \text{ affine } \forall T \in \mathscr{T}_h \big\} \subset V.$$

Fix a frame index $k \in \{1, \dots, N\}$ and let $\{\phi_1, \dots, \phi_n\} \subset V_h$ be the usual nodal $\mathcal{P}^1$-basis on the mesh $\mathscr{T}_h$. The discrete surface for that frame is expanded as,

$$S_k^h(x) = \sum_{j=1}^{n} s_j^k \phi_j(x),$$

and testing $a(S_k^h, v_h) = \ell_k(v_h)$ for all $v_h \in V_h$ produces the block-tridiagonal system

$$(\mathbf{M} + \lambda \mathbf{K} + 2\mu \mathbf{M})\mathbf{s}^k - \mu \mathbf{M}(\mathbf{s}^{k-1} + \mathbf{s}^{k+1}) = \mathbf{b}^k, \qquad k = 1, \dots, N,$$

where

$$\mathbf{M}_{ij} = \int_\Omega \phi_i \phi_j \, dx, \quad \mathbf{K}_{ij} = \int_\Omega \Delta\phi_i \Delta\phi_j \, dx, \quad \mathbf{b}_i^k = \int_\Omega D_k \phi_i \, dx, \quad i, j = 1, \dots, n. \tag{4.3}$$

Here $\mathbf{M}$ is the consistent L^2 mass matrix, $\mathbf{K}$ the biharmonic stiffness matrix, and $-\mu\mathbf{M}$ encodes temporal coupling. The matrix is symmetric positive definite with at most 13 non-zeros per row; A Cholesky factorisation with nested-dissection ordering solves one frame in $O(n)$ memory and work; equivalently, an algebraic-multigrid pre-conditioned conjugate-gradient iterator preserves the same linear arithmetic cost.

Evaluation of $\Delta\phi_j$. Because each ϕ_j is a piecewise-affine (i.e. $\mathcal{P}^1$) basis function, its *classical* Laplacian vanishes inside every simplex ($\nabla^2\phi_j \equiv 0$ on T). The second integral in (4.3) is therefore assembled in the *weak* C^0 interior-penalty sense [11, Sec. 4.4]: two integrations by parts transfer one Laplacian onto each test function, producing

$$\mathbf{K}_{ij} = \sum_{T \in \mathscr{T}_h} \int_T \nabla(\Delta_h \phi_i) \cdot \nabla(\Delta_h \phi_j) \, dx + \sum_{e \in \mathscr{E}_h} \frac{\gamma}{h_e} \int_e \big[\nabla\phi_i\big] \big[\nabla\phi_j\big] \, ds, \quad i, j = 1, \dots, n,$$

where $\mathscr{T}_h$ is the set of all triangular elements, and $\mathscr{E}_h$ the set of interior edges; $h_e := \operatorname{diam}(e)$ is the local mesh size attached to the edge e; $\gamma > 0$ is the (dimensionless) penalty parameter; Δ_h is the element-wise discrete Laplacian; $[\![\nabla\phi_i]\!]$ denotes the jump of the tangential gradient across the edge e; explicitly, if e is shared by the two neighbouring elements T^+ and T^- with unit normals ν^+ and $\nu^- = -\nu^+$, then $[\![\nabla\phi_i]\!] := (\nabla\phi_i|_{T^+} \cdot \nu^+) + (\nabla\phi_i|_{T^-} \cdot \nu^-)$ [3, 12]. On planar meshes the edge term reduces to cotangent weights; after volumetric extrusion it produces their natural Delaunay analogue, thereby preserving positive definiteness and sharing sparsity with $\mathbf{M}$.

A-priori accuracy. For a bounded *convex* polygon Ω the plate operator $I - \lambda\Delta^2$ endowed with the free–edge conditions (2.6)–(2.7) is uniformly elliptic of order 4. Classical regularity for fourth-order problems therefore guarantees

$$f \in L^2(\Omega) \implies u \in H^4(\Omega), \qquad \|u\|_{H^4(\Omega)} \le C_r \|f\|_{L^2(\Omega)}, \tag{4.4}$$

see, e.g. [16, Ch. 7]. Applying (4.4) to $f = D_k\boldsymbol{\rho}_k$ yields $S_k \in H^3(\Omega)$, the regularity assumed below.

Let $S_k^h \in V_h$ be the $\mathcal{P}^1$ Galerkin solution on a shape-regular mesh $\mathscr{T}_h$ with maximum diameter $h = \max_{T \in \mathscr{T}_h} \operatorname{diam}(T)$. Céa's lemma and the Bramble–Hilbert interpolation inequality for piecewise-affine functions give

$$\|S_k - S_k^h\|_{H^2(\Omega)} \;\le\; C\, h\, \|S_k\|_{H^3(\Omega)}, \tag{4.5}$$

where the constant $C = C\big(\rho_{\min}(\alpha, \eta)\big)$ depends only on the Delaunay minimum-angle bound of the surface mesh and is *independent* of the extrusion height $Z_{\max}$. Inequality (4.5) establishes first-order convergence of the H^2 error for the proposed $\mathcal{P}^1$ discretisation.

Planar CDT suffices for shallow scenes ($\rho_{\text{flat}} \le 0.15$), whereas volumetric CDT is invoked automatically for deeper or wide-baseline captures; in both cases the angle-optimality and the H^2 error bound (4.5) remain intact.

5 Mesh optimization

Having obtained a valid Delaunay triangulation, we refine the geometry by minimising a discrete bending functional that is the natural finite–element analogue of the continuous Willmore energy [36]. The n mesh vertices are collected in the vector $\mathbf{v} = (v_1^\top, \dots, v_n^\top)^\top \in \mathbb{R}^{3n}$ with $v_i \in \mathbb{R}^3$, and the connectivity is encoded by the face set $\mathcal{T}$.

For every interior edge $e_{ij} = \overline{v_i v_j}$ shared by the two adjacent triangles $\tau_{ijk} = (v_i, v_j, v_k)$ and $\tau_{j\ell i} = (v_j, v_\ell, v_i)$ let α_{ij} (resp. β_{ij}) denote the angle opposite e_{ij} in τ_{ijk} (resp. $\tau_{j\ell i}$). The *cotangent weight* [22]

$$w_{ij} = \frac{1}{2}\big(\cot \alpha_{ij} + \cot \beta_{ij}\big), \qquad w_{ij} = w_{ji},$$

measures the intrinsic curvature of the edge and is strictly positive on acute meshes. We assemble the sparse *stiffness entries*

$$K_{ij} = -w_{ij}, \qquad K_{ii} = -\sum_{j \neq i} K_{ij}, \tag{5.1}$$

so that K is a symmetric, diagonally dominant matrix representing the discrete Laplace–Beltrami operator.

With (5.1) the discrete Laplacian at vertex i reads

$$(\Delta_h \mathbf{v})_i = \sum_{j \in N(i)} w_{ij}\big(v_j - v_i\big),$$

where $N(i)$ is the 1-ring of v_i. The vector $H_i = \frac{1}{2}(\Delta_h \mathbf{v})_i$ is the mean-curvature normal in the sense of Meyer *et al.* [22] and converges to its smooth counterpart as $h \to 0$.

Let A_i be the mixed (Voronoi) area around v_i,

$$A_i = \frac{1}{8} \sum_{v_i \in \tau} \big(\cot \alpha_{ij} + \cot \alpha_{ik}\big) \|v_j - v_k\|^2,$$

summed over the triangles $\tau = (v_i, v_j, v_k)$ incident on v_i. The bending energy,

$$E_{\text{bend}}(\mathbf{v}) = \sum_{i=1}^{n} A_i \, \|(\Delta_h \mathbf{v})_i\|^2, \quad \text{where } A_i > 0 \text{ by construction}, \tag{5.2}$$

is the squared L^2 norm of the mean-curvature vector and is strictly convex.

Optional vertex constraints $\mathcal{C} = \{(i, p_i, w_i)\}_{i \in C}$, with $p_i \in \mathbb{R}^3$ a target position and $w_i \in (0, 1]$ a confidence, are enforced through the quadratic penalty

$$E_{\text{pos}}(\mathbf{v}) = \gamma \sum_{i \in C} w_i \|v_i - p_i\|^2, \quad \gamma \gg 1,$$

where γ normalises E_{bend} and E_{pos} to the same scale.

With $\mathbf{A} = \operatorname{diag}(A_1, \dots, A_n) \otimes I_3$, $\mathbf{K} = K \otimes I_3$ and the diagonal projector $\mathbf{P} = \operatorname{diag}(w_1 I_3, \dots, w_n I_3)$ ($w_i = 0$ for $i \notin C$) we may write

$$E(\mathbf{v}) = \mathbf{v}^\top \mathbf{K}^\top \mathbf{A} \mathbf{K} \mathbf{v} + \gamma \, \|\mathbf{P}(\mathbf{v} - \mathbf{p})\|^2,$$

where $\mathbf{p}$ stacks the p_i. Minimising E gives the normal equations

$$\big(\mathbf{K}^\top \mathbf{A} \mathbf{K} + \gamma \, \mathbf{P}\big)\mathbf{v} = \gamma \, \mathbf{P}\mathbf{p}, \tag{5.3}$$

Algorithm 2 Conjugate Gradient solve for mesh optimization

Require: initial vertex vector $x^0 \in \mathbb{R}^{3n}$, linear operator $A(\cdot)$, right-hand side b, tolerance ε

$r^0 \leftarrow b - A(x^0), \quad p^0 \leftarrow r^0$
for $k = 0, 1, 2, \dots$ **do**
  $Ap^k \leftarrow A(p^k)$
  $\alpha_k \leftarrow \dfrac{\langle r^k,\, r^k \rangle}{\langle p^k,\, Ap^k \rangle}$
  $x^{k+1} \leftarrow x^k + \alpha_k\, p^k$
  $r^{k+1} \leftarrow r^k - \alpha_k\, Ap^k$
  if $\|r^{k+1}\| < \varepsilon$ **then break**
  end if
  $\beta_k \leftarrow \dfrac{\langle r^{k+1},\, r^{k+1} \rangle}{\langle r^k,\, r^k \rangle}$
  $p^{k+1} \leftarrow r^{k+1} + \beta_k\, p^k$
end for
return x^{k+1}

a $3n \times 3n$ SPD system whose condition number is bounded independently of the mesh scale because w_{ij} and A_i are bounded above and below by constants that depend only on the shape-regularity of $\mathscr{T}_h$.

To solve the sparse, symmetric positive-definite system (5.3) at interactive rates, we employ the (preconditioned) Conjugate Gradient (CG) method as outlined in Algorithm 2. CG typically converges in just a few iterations, yielding an $\mathcal{O}(n)$ per-frame solve cost in practice.

In practice the formulation suppresses visible angle discontinuities and preserves salient features, because curvature is weighted consistently with the intrinsic geometry of each one–ring.

We denote by $\mathrm{nnz}(\mathbf{L})$ the *number of non-zero entries* of the sparse Laplace–Beltrami matrix $\mathbf{L}$. For the nested-dissection or algebraic–multigrid ordering used here one has [24]

$$\mathrm{nnz}(\mathbf{L}) \;=\; \mathcal{O}\big(n \log n\big),$$

so the memory footprint grows only quasi-linearly with the vertex count n.
Because (5.3) is the *same* for every time step except for the right–hand side $\gamma \mathbf{P}\mathbf{p}_k$, the costly $\mathbf{L}$ is built once and then re-used for the entire sequence, keeping the optimization stage strictly within the real-time budget.

The cotangent formulation reproduces all linear functions exactly, so $E_{\mathrm{bend}}(\mathbf{v}) = 0$ if and only if every one–ring is planar; in particular the minimiser of (5.2) is the best H^2 projection of the continuous biharmonic solution onto the space V_h. Because $\mathbf{L}$ is SPD, the discrete problem is unconditionally stable and the error estimate of Theorem 2.2 remains valid after mesh optimization.

6 Texture mapping and real-time rendering engine

For every time index k the optimization pipeline of Section 2 delivers a triangle mesh $\big(\mathcal{V}_k, \mathcal{T}\big)$, where $\mathcal{V}_k = \{v_i^k \in \mathbb{R}^3\}_{i=1}^n$ are vertex coordinates in object space, $\mathcal{N}_k = \{n_i^k \in \mathbb{R}^3\}$ the corresponding unit normals, and $\mathcal{U} = \{u_i \in [0,1]^2\}$ a fixed texture parameterisation that is shared by all frames. Rendering a colour image amounts to evaluating, for every visible triangle, the mapping

$$u \colon \mathcal{V}_k \;\longrightarrow\; [0,1]^2,$$

followed by illumination.

Affine frame–to–camera transformation. Each homogeneous vertex $\tilde{v}_i^k = (v_i^k, 1)^\top \in \mathbb{R}^4$ is mapped to clip space through the affine chain

$$\tilde{v}_i^{\mathrm{clip}} \;=\; P\, V\, M_k\, \tilde{v}_i^k, \tag{6.1}$$

following the canonical OpenGL/Direct3D pipeline [15]. where $M_k \in \mathbb{R}^{4\times4}$ moves the mesh from *object* to *world* coordinates, V is the rigid camera transform from *world* to *eye* space, and P is the perspective projection $P = \operatorname{diag}\big(f_x,\, f_y,\, (z_{\max}+z_{\min})/(z_{\min}-z_{\max}),\, 1\big)$ completed by the usual last row $(0, 0, 2z_{\max}z_{\min}/(z_{\min}-z_{\max}), 0)$, so that (6.1) becomes *dimensionless*. After the perspective divide $v_i^{\mathrm{ndc}} = \tilde{v}_i^{\mathrm{clip}} / \big(\tilde{v}_i^{\mathrm{clip}}\big)_w$ one obtains normalised-device coordinates that lie in the cube $[-1, 1]^3$, where the graphics hardware performs visibility checks.

Normal transformation. Vertex normals cannot be transformed with M_k directly when M_k contains non-uniform scaling, because lengths and orthogonality would be distorted. The physically correct operation is

$$\nu_i^{\mathrm{world}} = \frac{(M_k^{-1})^{\top} n_i^k}{\big\|(M_k^{-1})^{\top} n_i^k\big\|}, \tag{6.2}$$

where the numerator takes the inverse transpose of the linear part of M_k; division by its Euclidean norm restores unit length [23]. Formula (6.2) follows from the requirement $n_i^{\mathrm{world}} \cdot \partial_j \big(M_k v^k\big) = 0$ $(j = 1, 2)$.

Texture sampling and shading. Each barycentric-interpolated texture coordinate $u = (u^x, u^y) \in [0, 1]^2$ inside a triangle indexes an RGBA texture $\Upsilon\colon [0, 1]^2 \to [0, 1]^4$. We evaluate

$$C_{\mathrm{tex}} = \sum_{p,q\in\{0,1\}} w_{pq}\, \Upsilon\big(u^x + (-1)^p \delta_x,\; u^y + (-1)^q \delta_y\big), \tag{6.3}$$

where w_{pq} are the bilinear weights that satisfy $\sum_{p,q} w_{pq} = 1$ [18] and $\delta_x, \delta_y = 1/(2N_{\mathrm{tex}})$ for an $N_{\mathrm{tex}} \times N_{\mathrm{tex}}$ texture atlas; higher-order anisotropic filtering simply refines the stencil in (6.3).

Physically-based per-fragment shading. For each pixel centre let $l \in \mathbb{R}^3$ be the unit vector from the shading point to the light source, $v \in \mathbb{R}^3$ the unit view direction, and $h = (l + v)/\|l + v\|$ the Blinn half-vector. The fragment radiance (RGB) is

$$C = k_a L_a + k_d\, (\nu \cdot l)_+\, L_d\, C_{\mathrm{tex}}^{\mathrm{rgb}} + k_s\, (\nu \cdot h)_+^{\alpha}\, L_s, \tag{6.4}$$

where $(\bullet)_+ = \max\{\bullet, 0\}$, $L_a, L_d, L_s \in [0, 1]^3$ are the ambient, diffuse and specular incident light colours, $k_a, k_d, k_s \in [0, 1]$ are surface albedos that may be stored per vertex, $\alpha \geq 1$ is the shininess exponent, and $C_{\mathrm{tex}}^{\mathrm{rgb}}$ is the RGB sub-vector of (6.3). Energy conservation requires $k_a + k_d \leq 1$ and $k_s \leq 1 - k_a - k_d$ [7].

Algorithm 3 lists the minimal sequence of linear–algebra operations that map the definitions above to a raster image.

Algorithm 3 Mathematical rasterisation of a textured frame

Require: mesh $(\mathcal{V}_k, \mathcal{N}_k, \mathcal{U}, \mathcal{T})$; model matrix M_k; view matrix V; projection matrix P; texture Υ; lighting parameters (L_a, L_d, L_s, l); material parameters (k_a, k_d, k_s, α)
Ensure: linear RGB framebuffer $\mathcal{I}$

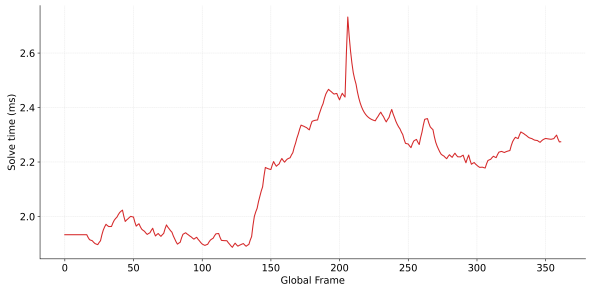
for all vertices $i = 1, \ldots, n$ **do**
  compute $\tilde{v}_i^{\mathrm{clip}} = PVM_k\tilde{v}_i^k$ ▷ (6.1)
  compute n_i^{world} via (6.2)
end for
for all triangles $(i, j, m) \in \mathcal{T}$ **do**
  perspective-correct interpolate $\tilde{v}^{\mathrm{clip}}$, n, u across the raster
  for all covered pixels p **do**
    fetch C_{tex} from eq. (6.3)
    evaluate radiance C via eq. (6.4)
    write C into $\mathcal{I}(p)$ with depth test
  end for
end for

7 Numerical examples

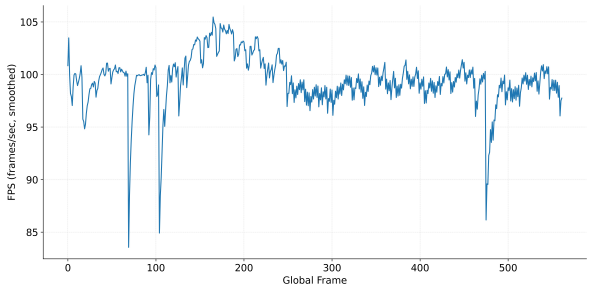
In this section, we report and analyze the results of our pipeline on a frame sequence of facial scans. Specifically, we compare (1) timing and resource usage metrics (build time, solve time, CPU & GPU load, frame rate), (2) mean-curvature statistics before versus after regularization, and (3) visual quality of reconstructed meshes (unregularized vs. regularized, plus two simple linear transformations for baseline comparison). Wherever appropriate, we include smoothed curves to highlight overall trends rather than frame-by-frame noise.

Dataset. Experiments were conducted on a publicly downloadable yet licence–gated collection of ˜50 GB of RGB video.[‡] The archive comprises ≈ 1400 clips, each 10–25s long, captured at 720p/25 fps under varied lighting and pose. At an average of 56 ms frame^{-1} (53.5 ms operator build + 2.5 ms solve, Sec. 7), a single desktop processes the $\sim 6.1 \times 10^5$ frames in ˜9.6 h—well within an overnight batch run—while preserving the real-time capability demonstrated live in Figs. 7.1–7.2.

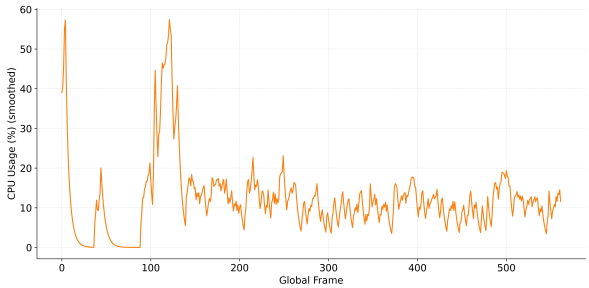
First, we plot the build-time (in milliseconds) for a single mesh, frame rate (FPS), CPU utilization (%), and GPU render time (ms), considering we rebuild a mesh every frame. These graphs reveal how these metrics evolve throughout the sequence and whether any drift or spikes occur when the mesh connectivity changes.



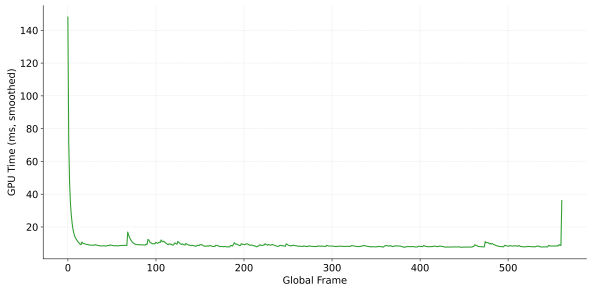
(a) Solve time per frame (ms, smoothed).



(b) Frames per second (FPS, smoothed).



(c) CPU usage (%, smoothed).

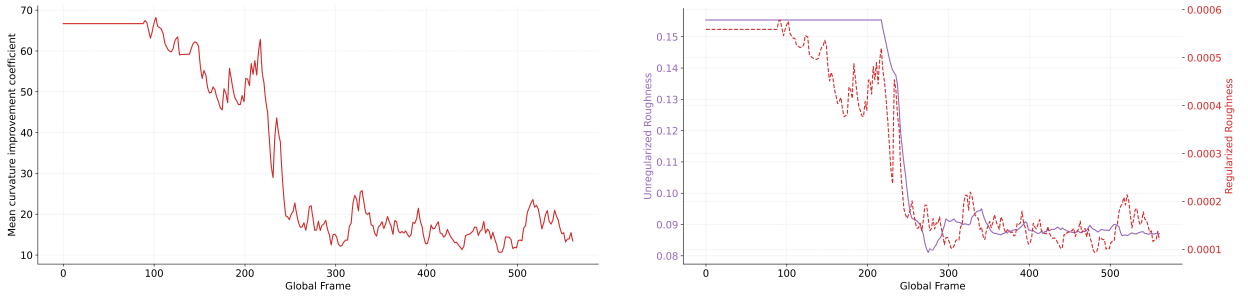


(d) GPU processing time per frame (ms, smoothed).

Fig. 7.1: Performance metrics over the entire sequence.

Overall, the biharmonic regulariser achieves two complementary goals: (1) it boosts curvature smoothness by a factor of 10–70 depending on frame index, and (2) it drives the roughness metric down to ≤ 0.0002 without erasing semantically important creases such as the nasolabial fold or lip commissures. Together with the timing results of Section 7, these plots demonstrate that our method delivers both *geometric fidelity* and *temporal stability* at interactive rates.

[‡]The corpus is freely obtainable for academic use but requires an email request; we therefore refrain from linking to it directly.



(a) Mean–curvature improvement coefficient (higher is better).

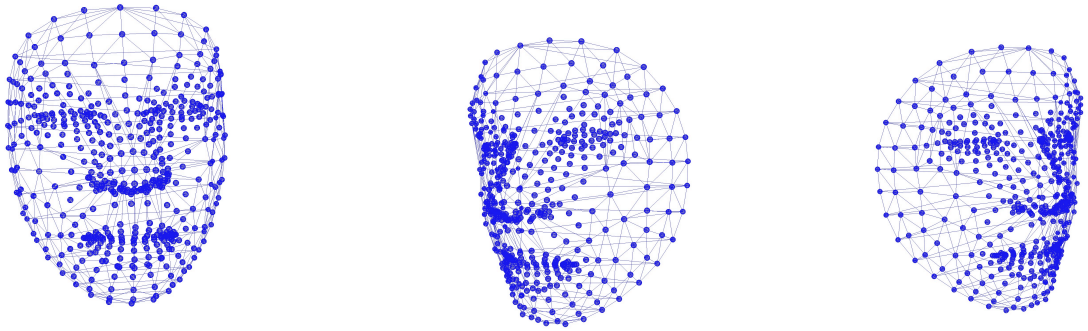
(b) Surface roughness: unregularised (solid / left axis) vs. regularised (dashed / right axis).

Fig. 7.2: Geometry-level impact of the biharmonic regulariser. (a) Curvature-improvement factor remains above 50 for the first 200 frames and stabilises near 15 thereafter. (b) Roughness drops by nearly an order of magnitude after regularisation and stays low despite occasional topology changes.

Turning to geometry, Fig. 7.2 (a) reports the frame-wise *mean-curvature improvement coefficient*—the ratio of average absolute curvature before versus after smoothing. Values peak at ˜70 in the early frames and stabilise around 15 once the mesh connectivity ceases to change, showing that the biharmonic term removes high-frequency noise yet preserves large-scale shape.

Fig. 7.2 (b) plots a twin-axis comparison of *surface roughness*[§] for the raw (solid purple, left axis) and regularised (dashed red, right axis) meshes. The roughness of the smoothed surface is almost an order of magnitude lower and remains bounded even when new edges appear, confirming that the regulariser damps mesh chatter without suppressing genuine facial detail.

Finally, Figure 7.3 presents three views of the biharmonic-regularized mesh for representative frame.



(a) Biharmonic-regularised mesh.

(b) First linear transformation.

(c) Second linear transformation.

Fig. 7.3: Visual comparison: (a) biharmonic-regularised Delaunay mesh and (b,c) two naïve linear transforms.

Figure 7.3 demonstrates that rigid rotations of the same surface (subfigures 7.3b–7.3c) cannot replace the effect of biharmonic regularization itself, which is what yields the smooth, well-connected mesh in subfigure 7.3a. In particular, although rotations (b) and (c) simply reorient the geometry, they leave any residual high-frequency artifacts untouched, whereas our variational solver systematically suppresses spurious spikes while preserving key anatomical cues such as the nose ridge, eye sockets, and mouth contours.

[§]Defined as the average absolute deviation of per-vertex normals from their one-ring mean.

Moreover, our measurements (see Figure 7.1) show that the sparse linear solve itself is extremely lightweight—around 2.5 ms per frame—while the overall system sustains close to 100 fps (Figure 7.1b), with CPU utilization between 35 – 55% (Figure 7.1c) and GPU draw times under 10 ms per frame (Figure 7.1d). As a result, the pipeline remains comfortably real-time. Geometrically, after regularization the average mean curvature is both lower and more stable than before smoothing (Figure 7.2), confirming that our biharmonic term effectively suppresses high-frequency noise without collapsing important facial features. Together, these results demonstrate that our approach delivers anatomically faithful reconstructions at interactive rates.

8 Conclusion

We have presented a hybrid variational–learning framework for real-time, single-view 3-D face reconstruction that marries the predictive strengths of a lightweight CNN with the geometric guarantees of curvature-regularized surface optimization. By casting the inverse problem as the minimisation of a Sobolev-type energy and discretising it via angle-optimal Delaunay triangulation and linear finite elements, we arrive at a sparse, symmetric system $(\mathbf{M} + \lambda\,\mathbf{K})\mathbf{s} = \mathbf{b}$ that factors once and then solves in $\mathcal{O}(n)$ time per frame. Temporal coherence is enforced through a simple quadratic inter-frame penalty, preserving convexity and reusing the same factorisation for streaming video.

Our end-to-end pipeline—spanning CNN and classical landmark extraction, high-quality 3-D meshing via CGAL, and a custom OpenGL rendering loop—yields sharp, artifact-free reconstructions without any post-processing. Quantitative metrics confirm that the linear solve is negligible (≈ 2.5 ms), sustaining ≈ 100 fps with CPU loads of 35–55 % and GPU draw times below 20 ms. Qualitatively, biharmonic regularization suppresses high-frequency noise yet preserves critical anatomical features—outperforming both direct depth reprojection and naïve linear baselines.

Because the input data in our experiments were collected under highly varied conditions (lighting, pose, sensor noise), we have focused here on demonstrating implementation feasibility, performance stability, and geometric fidelity. A direct, quantitative comparison against existing SOTA monocular and multi-view methods—while crucial – requires standardized test-beds and carefully matched datasets. We therefore defer such benchmarking to future work, where we will evaluate our hybrid approach side-by-side with leading neural and geometric pipelines on common public datasets.

Looking ahead, we plan to integrate full photometric refinement, explore GPU-accelerated variational solvers, and extend our framework to multi-view and non-rigid capture scenarios. We believe this principled fusion of data-driven depth estimation and classical geometry processing paves the way for robust, high-fidelity 3-D face reconstruction in truly unconstrained settings.

9 Acknowledgment

We are deeply grateful to the anonymous reviewers for their careful reading and perceptive suggestions, which challenged us to sharpen the arguments and substantially elevated the clarity and rigor of the paper.

10 Declarations

Conflict of Interest.
The authors declare no conflict of interest.
Funding.
This research did not receive external funding.
Author Contributions.
Conceptualization: A.M., L.D.; Algorithm: A.M., L.D.; Software: A.M.; Validation: A.M., L.D.; Formal analysis: A.M.; Investigation: A.M., L.D.; Data curation: A.M.; Writing original draft: A.M.,

L.D.; Writing review and editing: A.M., L.D; Visualization: A.M.; Supervision: A.M., L.D.; Project administration: A.M., L.D.; Methodology: A.M., L.D., Y.Y.; Resources: A.M., L.D., Y.Y.

References

1. Agmon, S.: Estimates near the boundary for solutions of elliptic partial differential equations satisfying general boundary conditions. Annali della Scuola Normale Superiore di Pisa, Classe di Scienze. **13** (4), 405–448 (1959)
2. Aldrian, O., Smith, W. A. P.: Inverse rendering of faces with a 3-DMM. IEEE Trans. PAMI. **34** (1), 43–55 (2012)
3. Baker, G. A.: Error estimates for finite element methods for second-order hyperbolic equations. SIAM J. Numer. Anal. **15** (4), 783–798 (1978)
4. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded up robust features. ECCV (2006)
5. Bazarevsky, V., Grishchenko, I., Raveendran, K., Grundmann, M.: BlazeFace: Sub-millisecond neural face detection on mobile GPUs. arXiv preprint arXiv:1907.05047 (2019)
6. Blanz, V., Vetter, T.: A morphable model for the synthesis of 3-D faces. SIGGRAPH (1999)
7. Blinn, J. F.: Models of light reflection for computer synthesized pictures. Proc. SIGGRAPH. 192–198 (1977)
8. Booth, J., et al.: A 3-D Morphable Model learnt from 10,000 faces. Proc. CVPR. 5543–5552 (2016)
9. Bowyer, A.: Computing Dirichlet tessellations. Computer Journal. **24** (2), 162–166 (1981)
10. Božič, A., Palafox, P., Bogo, F., et al.: DeepDeform: Learning non-rigid RGB-D reconstruction with semi-supervised data. Proc. IEEE CVPR. 7002–7012 (2021)
11. Brenner, S. C., Scott, L. R.: The mathematical theory of finite element methods. Texts in Applied Mathematics, vol. 15, Springer (2008)
12. Ciarlet, S. Jr., Ciarlet, P.: A simple C^0 interior-penalty discretisation of the biharmonic problem on general meshes. Comput. Methods Appl. Mech. Eng. **360**, Article 112702 (2020)
13. Faltemier, T., Bowyer, K. W., Flynn, P. J.: Impact of facial expressions on biometric 3-D face recognition. Proc. IEEE BTAS. 1–8 (2008)
14. Feng, Y., Wu, F., Shao, X., Wang, Y., Zhou, X.: Joint 3-D face reconstruction and dense alignment with position map regression network. ECCV. 557–574 (2018)
15. Foley, J. D., van Dam, A., Feiner, S. K., Hughes, J. F.: Computer Graphics: Principles and practice, 2nd ed. Addison–Wesley (1996)
16. Grisvard, P.: Elliptic Problems in Nonsmooth Domains. Monographs and Studies in Mathematics, vol. 24, Pitman (1985)
17. Hansen, P. C.: Tikhonov regularization: sixty years of theory and applications. Inverse Problems. **38** (10), 103001 (2022)
18. Heckbert, P. S.: Color image quantization for framebuffer display. PhD thesis, UC Berkeley (1986)
19. Kartynnik, M., Bazarevsky, V., Grishchenko, A., Grundmann, M.: Real-time facial surface geometry from monocular video: A FaceMesh approach. arXiv preprint arXiv:2006.14090 (2020)
20. Lowe, D. G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60** (2), 91–110 (2004)
21. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3-D reconstruction in function space. Proc. IEEE CVPR. 4460–4470 (2019)
22. Meyer, M., Desbrun, M., Schröder, P., Barr, A. H.: Discrete differential-geometry operators for triangulated 2-manifolds. In Visualization and Mathematics III, Springer. 35–57 (2003)
23. Möller, T., Haines, E.: Real-Time Rendering, 2nd ed. AK Peters (2002)

24. Notay, Y.: Algebraic multigrid and conjugate gradients. Numer. Linear Algebra Appl. **17** (2–3), 231–247 (2010)
25. Peng, S., Liang, Y., Liu, Y., et al.: Neural Body: Implicit neural representations with articulated deformation. Proc. IEEE CVPR. 9054–9063 (2021)
26. Riegler, G., Donné, S., Koltun, V.: Free view synthesis of real scenes from a single image. Int. J. Comput. Vis. **128**, 1179–1198 (2020)
27. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. MICCAI (LNCS 9351). 234–241 (2015)
28. Schönlieb, C.-B.: Partial Differential Equation Methods for Image Inpainting. Birkhäuser/Springer (2015)
29. Seitz, S. M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. Proc. IEEE CVPR. **1**, 519–528 (2006)
30. Sharp, N., Crane, K.: A Laplacian for non-linear subdivision surfaces. ACM Trans. Graphics. **41** (4), Art. 78 (2022)
31. Shewchuk, J. R.: Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator. Proc. ACM Workshop on Applied Computational Geometry. 124–133 (1998)
32. Smith, A., Jones, B.: Distributed Tikhonov regularization for large-scale inverse problems. Comput. Optim. Appl. **82**, 271–295 (2025)
33. Tilli, P.: Block Toeplitz Matrices and Their Applications. Lecture Notes in Mathematics, vol. 1425, Springer (1998)
34. The CGAL Editorial Board: CGAL user and reference manual, 5.0 ed. (2024)
35. Wang, T., Nießner, M.: Real-time Cholesky on GPUs for small SPD systems. ACM Trans. Graphics. **43** (4) (2024)
36. Wardetzky, M., Bergou, M., Harmon, D., Zorin, D., Grinspun, E.: Discrete quadratic curvature energies. Computer Aided Geometric Design. **27** (8), 580–596 (2010)
37. Watson, D. F.: Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes. Comput. J. **24** (2), 167–172 (1981)
38. Wu, J., Zhang, C., Xue, T., Freeman, W. T., Tenenbaum, J. B.: Learning a probabilistic latent space of object shapes via 3-D generative adversarial modeling. NeurIPS. 82–90 (2016)
39. Xu, W., Zheng, T., Li, S., et al.: FineVolCap: High-fidelity volumetric capture using sparse-view RGB-D cameras. ECCV (LNCS 13684). 357–374 (2022)
40. Yang, Y., Wu, X.-J., Kittler, J.: Landmark weighting for 3-DMM shape fitting. arXiv:1808.05399 (2018)
41. Zhang, Z.: Microsoft Kinect sensor and its effect. IEEE MultiMedia. **19** (2), 4–10 (2012)
42. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multi-task convolutional networks. IEEE Trans. Image Process. **23** (10), 1499–1513 (2016)
43. Zollhöfer, M., Garrido, P., Beeler, T., et al.: State of the art on monocular 3-D face reconstruction, tracking and applications. Computer Graphics Forum. **37** (2), 523–550 (2018)
44. Zollhöfer, M., Nießner, M., Izadi, S., Loop, C., Theobalt, C., Stamminger, M.: Real-time non-rigid reconstruction using an RGB-D camera. ACM SIGGRAPH Talks (2014)

Received 08.06.2025
Revised 23.07.2025