ISSN: 2834-7706

Research Article

Journal of Mathematical Techniques and Computational Mathematics

Spectrally Regularized Spline Mapping with Recursive Cubic Residual Correction

Greg Passmore*

PassmoreLab, Austin, Texas, USA

***Corresponding Author**
Greg Passmore, PassmoreLab, Austin, Texas, USA.

Submitted: 2026, Jan 02; **Accepted:** 2026, Mar 26; **Published:** 2026, Apr 13

Citation: Passmore, G. (2026). Spectrally Regularized Spline Mapping with Recursive Cubic Residual Correction. *J Math Techniques Comput Math, 5*(3), 01-28.

Abstract

*This work introduces the **Fourier-Space Spline with Recursive Polynomial Residuals (FSS-RPR)**, a hybrid empirical georegistration and inversion method intended for ISR exploitation when physical sensor models are incomplete, denied, vendor-specific, or operationally impractical. The design fuses two mature mathematical lineages; spectral representations originating in Fourier's heat-diffusion analysis and spline-based smooth function estimation formalized by Schoenberg and later developed for computation by de Boor; with a residual-stacking estimator path that is structurally analogous to stagewise additive modeling [1-4]. The principal competing approach is the rigorous sensor-model/bundle-adjustment family, which can exceed surrogate accuracy when calibration/metadata are complete, but often fails operationally due to metadata availability, datum ambiguity, or workflow constraints; the competing critique is that any surrogate can overfit and can degrade sharply under extrapolation outside its fitted domain.*

The method targets a specific ISR failure mode: georegistration error dominated by smooth, low-frequency spatial bias (platform/attitude bias, timing skew, ephemeris drift, or systematic terrain/height convention mismatch), with a residual component that is structured but not captured by a single global smooth warp. The FSS component enforces controlled smoothness and bandwidth in a way that is auditable and computationally efficient, especially when knots are uniform and spectral penalties are applied using FFT-class methods [5]. The RPR component then absorbs remaining deterministic structure in the residual field using a fixed low-order polynomial basis and recursive residual fitting, preserving the simplicity of evaluation and enabling explicit failure-mode diagnosis through layerwise residual decay.

Background

Joseph Fourier (1768–1830) was a French mathematician and physicist whose 1822 Théorie analytique de la chaleur introduced Fourier series as a practical representation for diffusion phenomena, establishing the operational logic of separating low- and high-frequency content in a signal [1]. James Cooley (1926–2016) and John Tukey (1915–2000) published the 1965 FFT algorithm that made large-scale spectral computation routine on digital systems; Tukey also contributed foundational ideas in exploratory data analysis that strongly shaped modern signal exploitation workflows [5]. Isaac Jacob Schoenberg (1903–1990), a Romanian-born American mathematician, formalized spline functions as a systematic approximation tool; Carl de Boor (1937–) produced the standard computational treatment for B-splines and their stable evaluation [2, 3]. Michael Unser's signal-processing synthesis emphasizes splines as continuous-discrete bridges with tractable frequency-domain characterizations, directly relevant to Fourier-domain regularization choices [6]. Grace Wahba's smoothing-spline framework supplies the canonical statistical viewpoint on penalized smoothness [7]. Jerome Friedman's boosting formulation formalizes stagewise residual fitting as functional optimization, which motivates the residual-stacking estimator path used in the RPR stage [4].

Method

FSS-RPR is structured as a two-stage (or two-block) map for the normalized forward function $\mathbf{f} : [x_n, y_n, H_n]^\top \mapsto [\lambda_n, \phi_n, \hat{H}_n]^\top$. The Fourier-space spline provides the low-frequency, globally smooth component:

$$\mathbf{f}_{\mathrm{FSS}}(x_n, y_n, H_n) = \sum_{p,q,r} \mathbf{c}_{pqr}\, B_p(x_n) B_q(y_n) B_r(H_n),$$

where $B_p(\cdot)$ are B-spline basis functions (typically cubic), and $\mathbf{c}_{pqr} \in \mathbb{R}^3$ are control coefficients. The "Fourier-space" attribute refers to solving for $\mathbf{c}$ under a spectral smoothness constraint applied to the control lattice (or to its finite-difference derivatives) using a discrete Fourier transform operator $\mathcal{F}$, typically implemented with FFT methods:

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} \|\mathbf{A}\mathbf{c} - \mathbf{q}\|_2^2 + \gamma \|\mathbf{W}\,\mathcal{F}\mathbf{c}\|_2^2,$$

where $\mathbf{A}$ is the spline design matrix, $\mathbf{q}$ stacks the target observations, $\gamma \geq 0$ is the smoothness weight, and $\mathbf{W}$ weights spectral components to suppress high-frequency energy consistent with a mission-defined bandwidth prior [4,6]. This spectral form makes the smoothing mechanism explicit: high-frequency degrees of freedom are penalized by construction rather than relying on ad hoc knot spacing alone.

After fitting the spline, the residual field is computed in normalized coordinates:

$$\mathbf{r}_0(x_n, y_n, H_n) = \mathbf{q}_n - \mathbf{f}_{\mathrm{FSS}}(x_n, y_n, H_n).$$

The recursive polynomial residuals (RPR) stage fits a fixed low-order basis to $\mathbf{r}_0$ using stagewise residual modeling:

$$\mathbf{f}_{\mathrm{RPR}}(x_n, y_n, H_n) = \sum_{m=0}^{M} \mathbf{C}_m^{\top} \mathbf{T}(x_n, y_n, H_n), \qquad \mathbf{C}_m \in \mathbb{R}^{20\times 3},$$

with $\mathbf{T}(\cdot)$ the shared 20-term cubic monomial basis and $\mathbf{C}_m$ fitted to the residual at layer m using ridge/Tikhonov regularization. The total forward model is then

$$\mathbf{f}(x_n, y_n, H_n) = \mathbf{f}_{\mathrm{FSS}}(x_n, y_n, H_n) + \mathbf{f}_{\mathrm{RPR}}(x_n, y_n, H_n),$$

and denormalization maps $\mathbf{f}$ into $(\lambda, \phi, \hat{H})$ under the previously defined output offsets and scales.

Value

The technical claim is not that FSS-RPR is a physical sensor model. It is a controlled surrogate with two explicitly different bias models: a bandwidth-limited smooth warp (FSS) and a structured low-order residual corrector (RPR). The primary operational benefits in ISR are deterministic evaluation, explicit regularization knobs tied to bandwidth and residual decay, and compatibility with iterative inversion using Gauss–Newton/LM updates when H or (x, y) are unknown. The principal failure modes are predictable: extrapolation outside the normalization domain; height-type mismatch (ellipsoid vs orthometric) contaminating both spline and residual terms; and inversion instability when local Jacobians are near-singular due to weak observability. These are not incidental; they define the conditions under which a rigorous sensor model or multi-view geometry becomes the correct substitute.

Keywords: georegistration, rational polynomial coefficients, recursive residual fitting, ISR exploitation, spectral regularization

1. Scope, Operational Context, and I/O Contract

This section fixes the operational contract for ISR georegistration and mensuration in WGS84, specifies the required inputs and fitted parameters, and specifies the expected forward and inverse outputs, while identifying the lineage of polynomial surrogates in imagery exploitation standards and in classical least-squares orbit determination, noting the persistent controversy between surrogate fidelity and physically parameterized sensor models, and the known instability under extrapolation beyond calibrated support [8-12].

The operational intent is georegistration/mensuration for ISR pipelines where a deterministic mapping from image coordinates to geodetic coordinates is required but sensor metadata are incomplete, vendor RPCs are unavailable or untrusted, or an empirical correction layer is required above an existing mapping. The surrogate lineage is consistent with rational polynomial camera models and RPC-style interchange conventions that became a practical mechanism for decoupling exploitation from proprietary sensor physics, particularly in NITF ecosystems used for imagery dissemination and exploitation [8]. This class of surrogate is not a physical sensor model. It is an operational approximation that trades first-principles fidelity for portability, repeatability, and controllable regression stability, and it therefore inherits the standard risks of polynomial/rational regression, with extrapolation failure being the dominant operational hazard [11,12].

The estimator structure is supervised regression for the forward map plus iterative numerical inversion for the inverse map. The forward map is trained from correspondences between image-space samples and ground-truth geodetic coordinates, using least-squares estimation principles that trace to early orbit determination where noisy observations are fit by minimizing quadratic residual energy [9,10]. Those origins remain operationally relevant because the contemporary ISR problem retains the same structural features: noisy measurements, incomplete models, and the requirement for stable parameter estimation under limited support.

The expected deployment context is a DoD exploitation workflow that must remain compatible with positional reference policy and ISR product standards. Compatibility with DoD positional reference procedures is required for coordinate reporting and downstream interoperability in mission systems [13]. Interchange assumptions are constrained by imagery container standards, with NITF remaining the primary still imagery and metadata interchange mechanism in many pipelines [8]. The mention of MIL-STD-2500 / NITF identifies the compliance environment for still imagery packaging and associated metadata interchange; this is treated as a practical boundary condition on how the mapping, metadata, and provenance are represented in production exploitation toolchains.

Mission relevance is direct. The forward map supports target coordinate generation and mensuration outputs used as inputs to joint targeting processes, including coordinate refinement, aimpoint definition, and other mensuration-adjacent exploitation tasks that feed operational targeting workflows [14]. The inverse map supports back-projection tasks such as chip-to-map alignment, image-to-ground ray intersection under terrain constraints, and iterative elevation solving when H is unknown or must be reconciled with a DEM or geoid-consistent height convention. In ISR terms, the forward map is the production interface for exploitation tooling, while the inversion interface is the enabling mechanism for interactive mensuration, cueing, and alignment.

The I/O contract is defined as follows. Inputs include raw image-space coordinates (x, y) in pixels or local sensor coordinates in meters; an elevation variable H in meters when externally supplied (DEM-constrained) or treated as unknown in inversion; and the normalization constants (x_0, y_0, H_0) and (x_s, y_s, H_s) used to map inputs into a bounded domain (typically $[-1, 1]$) to stabilize regression. Outputs include the forward-mapped geodetic coordinates (λ, ϕ) in degrees and a predicted elevation output $\hat{H}$ in meters when elevation is modeled as an output channel; the associated denormalization constants (λ_0, ϕ_0) and (λ_s, ϕ_s) are required when normalized outputs are used in the regression. Fitted parameters include the polynomial residual coefficient stacks (and any spline control coefficients if the Fourier-space spline stage is present), along with any regularization weights used to condition the least-squares solve. The contract further assumes that all coordinates and heights are interpreted under a consistent WGS84 datum definition and height convention; if orthometric heights are supplied but ellipsoidal heights are assumed (or vice versa), systematic bias will contaminate the learned mapping and its inversion behavior, and this must be treated as a failure condition rather than a minor nuisance [13].

The primary controversy is whether a surrogate of this type should be used when a rigorous physical model is available. Rational polynomial surrogates can be extremely accurate within calibrated support, but they do not carry the same physical interpretability, error propagation guarantees, or extrapolation safety as full photogrammetric or sensor-physics models. This controversy is explicit in the RPC literature: the method family is widely used in production exploitation due to practicality, yet is repeatedly criticized for instability outside the support region and for sensitivity to the distribution of calibration points, particularly when the scene geometry departs from the training set or when height dependence is weakly observed [11,12]. Operationally, this means the method is appropriate when the support volume is controlled and verified, and it is inappropriate when extrapolation is required, when the calibration set does not bracket the mission scene, or when the intended product requires physically meaningful uncertainty propagation beyond empirical validation.

Input contract (forward evaluation). Required runtime inputs are (x, y) in the sensor's image coordinate system (pixels or sensor-plane units), and a height variable H (or a proxy mapped into H) consistent with the training normalization. If local vertical coordinates are available (e.g., LiDAR/project frame z_{local}), a pre-map may be applied to construct an RPC-compatible height variable h or to enforce a consistent height definition across sensors.

Parameter contract (model instance). Each deployed model instance requires the normalization offsets and scales (x_0, y_0, H_0) and (x_s, y_s, H_s), output offsets/scales (λ_0, ϕ_0, H_0) and (λ_s, ϕ_s, H_s) (or distinct , $\hat{H}_0, \hat{H}_s$ if elevation is treated as an output with a different range), the recursion depth M (often $M = 6$ for seven layers), and the coefficient tensors for each output channel. The regression dataset requires N correspondences between input tuples (x_i, y_i, H_i) and targets $(\lambda_i, \phi_i, \hat{H}_i)$ in the chosen datum. DoD policy and many exploitation systems assume WGS84 latitude/longitude and ellipsoid height unless explicitly stated otherwise [13,15]. If orthometric heights are used anywhere in the pipeline, a geoid model identifier and conversion must be specified [16,13].

Expected outputs. The forward model returns $(\lambda, \phi, \hat{H})$ in physical units by denormalizing the stacked polynomial outputs. Operationally, (λ, ϕ) are typically consumed as WGS84 geodetic coordinates for mapping, mensuration, or target location products; $\hat{H}$ is either treated as a predicted elevation consistent with the training convention or as a stabilization residual in inversion when H is unknown and a terrain constraint is present. The solver component returns estimated H given (x, y) and desired (λ^*, ϕ^*), or estimated (x, y) given (λ^*, ϕ^*, H^*), using nonlinear least-squares iterations.

The method assumes: $(x_s, y_s, H_s) \neq 0$; training data span the operational domain; and the basis terms are sufficiently excited to avoid rank deficiency (e.g., height variation is required to estimate coefficients multiplying H_n terms). Failure is expected when the model is extrapolated outside its normalization box, when (x, y, H) have inconsistent datums/height definitions versus training, or when inversion is attempted under near-zero sensitivity $\partial(\lambda, \phi)/\partial H$ (height unobservable). In ISR terms, this corresponds to poor geometry, insufficient terrain relief, or underconstrained metadata.

Before the section summary table, note that this I/O framing primarily competes with "rigorous" physical sensor models and bundle adjustment, which can outperform surrogates when calibration and metadata are available, but can be operationally heavier or metadata-dependent.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Least squares objective $\min \lVert A\theta - b \rVert_2^2$	1805 Legendre (Paris) [9] 1809 Gauss (Göttingen) [10]	Noise model assumptions; sensitivity to conditioning	Rigorous sensor model / collinearity + bundle adjustment [20,21] vendor RPC/RFM [11]	Produces deterministic mapping for ISR georegistration and mensuration
WGS84 positional reference requirement	DoD policy (CJCSI 3900.01E) [13]	Datum mismatch and height-type ambiguity (ellipsoid vs MSL)	Use of local datums/ vertical datums with explicit transforms [15,16]	Prevents systematic targeting/mensuration offsets from datum inconsistency
NITF/RPC metadata normalization concept	NIMA STDI-0002 [8]	RPC/RFM is a surrogate model; extrapolation and bias issues	Bias-corrected RPC block adjustment [12] rigorous physical sensor models [11]	Interoperable representation for ISR imagery exploitation systems

2. Lineage and Foundational Sources

Let's trace the mathematical and systems lineage from RPC/RFM standards through least-squares estimation and modern residual-fitting viewpoints, because my method is an original synthesis that reuses established components and adapts them to multi-output ISR georegistration/inversion with explicit solver structure [8-10,17].

Least Squares: Adrien-Marie Legendre (1752–1833) was a French mathematician who introduced least squares in the context of comet orbit determination; he also made foundational contributions to elliptic integrals used widely in mathematical physics [9]. Carl Friedrich Gauss (1777–1855), a German mathematician and astronomer, independently developed least squares and applied it broadly in geodesy and astronomy; his work is central to modern estimation theory in geospatial measurement systems [10]. The least-squares objective is the core estimator for polynomial coefficients in this method.

Regularization / Ridge: Andrey N. Tikhonov (1906–1993) was a Soviet mathematician and geophysicist known for formalizing regularization for ill-posed inverse problems, providing a principled way to stabilize solutions in the presence of noise and ill-conditioning [17]. In statistics, ridge regression formalized an equivalent quadratic penalty to address multicollinearity in linear regression [18]. This method uses a Tikhonov/ridge penalty at each residual stage.

RPC/RFM Basis Lineage: The 20-term cubic monomial basis in three normalized variables is not an arbitrary decision: it is the same degree/order structure used in common rational function models for satellite imagery, where numerator and denominator polynomials are cubic in normalized coordinates (NIMA 2000; Tao and Hu 2001). Hartley and Saxena analyzed the cubic rational polynomial camera model and its properties as a projective camera approximation, emphasizing that it is a flexible surrogate rather than a physical model [19]. Operationally, RPC/RFM are entrenched because they enable sensor-agnostic exploitation and interchange; the controversy is that they can hide systematic sensor errors and can be brittle in extrapolation or under incorrect height conventions, motivating bias correction and block adjustment [12].

Recursive Residual Fitting Lineage: The distinctive component here is not "polynomial regression" itself, but repeated residual fitting on a fixed basis and summation of layers. Sequential residual fitting is a known strategy in statistical learning (stagewise additive modeling / boosting), but this synthesis constrains the base learner to a fixed 20-term cubic basis aligned with RPC conventions and extends it to multi-output geodetic mapping plus inversion solvers [4]. The major critique is overfitting to systematic and random error if recursion depth is too large or if regularization is too weak; this is operationally visible as degraded transfer to new imagery or to out-of-box geometry. I mitigate this through limiting recursion at the noise floor.

Competing methods: Rigorous sensor models (collinearity equations, physical camera models) and bundle adjustment remain the primary competitor for high-accuracy geolocation when metadata/calibration exist [20,21]. For point clouds and 3D alignment, rigid/affine methods and ICP are common competitors [22]. For smooth warping, spline and radial basis function approaches compete [23]. In ISR environments, the selection is often constrained by available metadata, denial/deception risk, and interoperability requirements.

Before the section summary table, note that the lineage is intentionally dual: it anchors to DoD/NGA interchange standards (RPC) and to classical estimation, while borrowing the residual-stacking concept from modern regression to create an implementable refinement stack.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Least squares $\min_\theta \|\|A\theta - b\|\|_2^2$	1805 Legendre [9] 1809 Gauss [10]	Requires modeling error assumptions; unstable under ill-conditioning	Robust M-estimators [44] total least squares [42]	Baseline estimator for coefficient identification from ISR correspondences
Tikhonov/ridge $\min \|\|A\theta - b\|\|_2^2 + \lambda_{\text{reg}} \|\|\theta\|\|_2^2$	1963 Tikhonov [17] 1970 ridge [18]	Bias introduced by penalty; choosing λ_{reg} is nontrivial	Cross-validation, L-curve selection, generalized cross-validation [42]	Stabilizes coefficients for fielded ISR pipelines with noisy tie points
RPC/RFM cubic polynomials in normalized variables	2000 NIMA STDI-0002 controlled extension [8]	Surrogate, not physical; extrapolation and height-datum errors	Rigorous sensor models; bias-corrected RPC block adjustment [12]	Interoperable mapping layer for imagery exploitation and mensuration
Stagewise residual fitting $r_{m+1} = r_m - Ac_m$	2001 Friedman (Stanford) [4]	Overfitting risk; layer count becomes a hyperparameter	Single-pass ridge/OLS; boosting with shrinkage; spline/RBF warps [23]	Allows controlled complexity increase without changing polynomial degree/order

3. Symbols and Variable Conventions

This section fixes notation and units for implementation under ISR constraints, because ambiguity in height type, datum, and symbol reuse is a primary operational failure mode; historically, geospatial exploitation standards emphasize explicit datum/height conventions, and competing sensor-model approaches often fail in practice due to inconsistent conventions rather than algorithmic weakness [8,13].

The following table is preserved from the original draft and then extended where rigor requires additional symbols. The single largest ambiguity in the original is the reuse of λ for both longitude and the Tikhonov regularization parameter.

In this revision, the regularization parameter is denoted λ_{reg} to avoid collision with longitude λ .

Symbol	Meaning	Units / Domain	Notes
x, y	Image-space pixel coordinates or local sensor coordinates	pixels or meters	Input coordinates in image or LiDAR reference frame
z_{local}	Local vertical coordinate in LiDAR/project frame	meters	Vertical coordinate in local reference frame prior to conversion
h	Height used in RPC equations	meters	May be derived from z_{local} via an affine or bilinear pre-map
H	Height variable used in the cubic basis	meters	In the original text, H plays the role of the RPC height; treat $H \equiv h$ unless explicitly distinguished
X, Y, Z	World or map coordinates	meters	Output coordinates in global reference frame (e.g., UTM, ECEF)
λ, ϕ	Ground longitude and latitude	degrees	Geographic coordinates on Earth (for image RPC)
x_n, y_n, z_n, h_n	Normalized input variables	dimensionless	Centered and scaled to range $[-1, 1]$
H_n	Normalized height variable used in basis	dimensionless	$H_n = (H - H_0)/H_s$
X_n, Y_n, Z_n	Normalized world coordinates	dimensionless	Range $[-1, 1]$ after scaling
λ_n, ϕ_n	Normalized longitude and latitude	dimensionless	Derived from λ and ϕ by centering and scaling
x_0, y_0, z_0, h_0	Offset (mean or midpoint) for input variables	same as x, y, z, h	Used for normalization
H_0	Height offset used for normalization	meters	For rigor, allow separate $\hat{H}_0$ if elevation is also an output
X_0, Y_0, Z_0	Offset (mean or midpoint) for world coordinates	meters	Used for normalization in 3D→3D mapping
λ_0, ϕ_0	Offset (mean or midpoint) for ground coordinates	degrees	Used for normalization
x_s, y_s, z_s, h_s	Scale factors (half-range) for input variables	same as x, y, z, h	Used to map inputs to $[-1, 1]$
H_s	Height scale factor (half-range)	meters	Must be nonzero
X_s, Y_s, Z_s	Scale factors (half-range) for world coordinates	meters	Used for denormalization in 3D→3D mapping

λ_s, ϕ_s	Scale factors (half-range) for ground coordinates	degrees	Used for denormalization
$h = \alpha_0 + \alpha_1 z_{\text{local}} + \alpha_2 x + \alpha_3 y + \alpha_4 x z_{\text{local}} + \alpha_5 y z_{\text{local}}$	Affine/bilinear vertical pre-map	meters	Converts local vertical to RPC height variable
α_i	Coefficients of affine or bilinear vertical transformation	dimensionless	Determined from calibration
$P_i(u, v, w)$	Third-order polynomial	dimensionless	Represents numerator or denominator in rational model
N_λ, D_λ	Numerator and denominator for longitude	dimensionless	Each is a weighted sum of 20 cubic monomials
N_ϕ, D_ϕ	Numerator and denominator for latitude	dimensionless	Analogous to N_λ, D_λ
N_X, D_X	Numerator and denominator for X axis	dimensionless	Used in LiDAR 3D→3D mapping
N_Y, D_Y	Numerator and denominator for Y axis	dimensionless	Used in LiDAR 3D→3D mapping
N_Z, D_Z	Numerator and denominator for Z axis	dimensionless	Used in LiDAR 3D→3D mapping
$a_k^{(\lambda)}, \tilde{a}_k^{(\lambda)}$	Longitude polynomial coefficients	dimensionless	Numerator and denominator coefficients (20 each)
$a_k^{(\phi)}, \tilde{a}_k^{(\phi)}$	Latitude polynomial coefficients	dimensionless	Numerator and denominator coefficients (20 each)
$a_k^{(X)}, \tilde{a}_k^{(X)}$	X-axis polynomial coefficients	dimensionless	For LiDAR RPC model
$a_k^{(Y)}, \tilde{a}_k^{(Y)}$	Y-axis polynomial coefficients	dimensionless	For LiDAR RPC model
$a_k^{(Z)}, \tilde{a}_k^{(Z)}$	Z-axis polynomial coefficients	dimensionless	For LiDAR RPC model
$T_k(x_n, y_n, h_n)$	k-th monomial term	dimensionless	Terms such as $x_n, y_n, h_n, x_n y_n, x_n^2, h_n^3$
$\mathbf{t}(x_n, y_n, h_n)$	20-element cubic basis vector	dimensionless	Column vector of monomial terms
$\mathbf{a}_{\text{num}}, \mathbf{a}_{\text{den}}$	Coefficient vectors	dimensionless	Contain 20 polynomial coefficients each
$\boldsymbol{\theta}_\lambda, \boldsymbol{\theta}_\phi$	Complete parameter sets for longitude and latitude	dimensionless	Concatenated numerator and denominator coefficients
$\boldsymbol{\theta}_X, \boldsymbol{\theta}_Y, \boldsymbol{\theta}_Z$	Parameter sets for LiDAR mapping axes	dimensionless	Used in 3D→3D rational regression
$\mathbf{A}_\lambda, \mathbf{A}_\phi$	Design matrices for least squares	dimensionless	Contain monomial evaluations for samples

$\boldsymbol{\varepsilon}_\lambda, \boldsymbol{\varepsilon}_\phi$	Residual vectors	dimensionless	Difference between predicted and observed coordinates
$\varepsilon_X, \varepsilon_Y, \varepsilon_Z$	Axis-wise residuals for LiDAR fit	meters	$X_{\text{obs}} - X_{\text{pred}}$, etc.
$N(\lambda, \phi)$	Geoid undulation	meters	Ellipsoidal minus orthometric height
$\lambda_n = \frac{N_\lambda}{D_\lambda}$	Normalized longitude	dimensionless	Rational function of (x_n, y_n, h_n)
$\phi_n = \frac{N_\phi}{D_\phi}$	Normalized latitude	dimensionless	Rational function of (x_n, y_n, h_n)
$X_n = \frac{N_X}{D_X}$ explains similarly	dimensionless	Rational functions of normalized inputs	Included for completeness
$\lambda = \lambda_0 + \lambda_s \lambda_n$	Denormalized longitude	degrees	Converts normalized to physical value
$\phi = \phi_0 + \phi_s \phi_n$	Denormalized latitude	degrees	Converts normalized to physical value
$X = X_0 + X_s X_n$	Denormalized X coordinate	meters	Converts normalized to global frame
$Y = Y_0 + Y_s Y_n$	Denormalized Y coordinate	meters	Converts normalized to global frame
$Z = Z_0 + Z_s Z_n$	Denormalized Z coordinate	meters	Converts normalized to global frame
$\varepsilon_{\lambda,i}, \varepsilon_{\phi,i}$	Individual residuals for each sample	dimensionless	Used in least-squares minimization
$\sum_i (\cdot)^2$	Sum of squared residuals	—	Objective function minimized during regression
N	Number of GCPs or tie points	integer	≥20 for solvability of one 20-term stage; larger required for stability; height variation required for H terms
$\mathbf{f}(x_n, y_n, H_n)$	Stacked normalized outputs [	$\mathbb{R}^{3\times1}$	Forward map in normalized units
$\mathbf{f}_0$	Output offsets $[\lambda_0, \phi_0, H_0]^\top$	$\mathbb{R}^{3\times1}$	Denormalization offsets
$\mathbf{S}$	Output scaling matrix $\text{diag}([\lambda_s, \phi_s, H_s])$	$\mathbb{R}^{3\times3}$	Denormalization scales
$\mathbf{A}^{(\lambda)}, \mathbf{A}^{(\phi)}, \mathbf{A}^{(\hat{H})}$	Coefficient matrices by output axis	$\mathbb{R}^{7\times20}$ each	Rows: recursion layers $(m = 0 \ldots 6)$, columns: basis terms $(k = 0 \ldots 19)$
$\mathcal{A}_{j,m,k}$	Rank-3 coefficient tensor	$\mathbb{R}^{3\times7\times20}$	j: axis, m: recursion layer, k: basis term

$\mathcal{A}_{:,m,k}$	Axis slice $[a^{(\lambda)}_{m,k}, a^{(\phi)}_{m,k}, a^{(\hat{H})}_{m,k}]^\top$	$\mathbb{R}^{3\times 1}$	Column vector of per-axis coefficients
j	Axis index	$\{0,1,2\}$	$j=0\to\lambda, j=1\to\phi,$ $j=2\to\hat{H}$
m	Recursion layer index	$\{0,\ldots,6\}$	Seven-layer recursive stack
k	Basis-term index	$\{0,\ldots,19\}$	Monomial ordering fixed across all axes
$\hat{H}, \hat{H}_n$	Predicted elevation (raw, normalized)	scalar	Used instead of H' for clarity
n subscript	Indicates normalized variable	—	Used consistently for $(x_n, y_n, H_n, \lambda_n, \phi_n, \hat{H}_n)$
$\operatorname{diag}(\cdot)$	Diagonal matrix operator	—	Constructs $\mathbf{S}$ from scale vector
$\mathcal{A}$	Full tensor stack of all coefficient matrices	$3\times 7\times 20$	Represents the complete coefficient set
420 coefficients	Total polynomial-stack parameter count	scalar	3 axes × 7 layers × 20 terms
λ_{reg}	Tikhonov/ridge regularization parameter	scalar, ≥ 0	Renamed from λ to avoid collision with longitude

Before the section summary table, note that the primary controversy in notation is not aesthetic: mixing height conventions (ellipsoid vs orthometric) and overloading symbols produces ISR-scale geolocation biases comparable to or larger than the regression residuals [13,15].

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
RPC normalization to $[-1,1]$ using offsets/scales	2000 NIMA STDI-0002 [8]	Failure when applied outside the validity box; hidden datum/height assumptions	Rigorous sensor models [20] bias-corrected RPC [12]	Prevents numeric blow-up; aligns with ISR imagery interchange conventions
WGS84 height and geoid model identification	2014 NGA.STND.0036 [15] 2023 CJCSI 3900.01E [13]	Operational products mix MSL and ellipsoid heights; geoid model ambiguity	Explicit vertical datum pipelines using EGM models [16]	Prevents systematic height/position errors in mensuration and targeting

4. Recursive Cubic Polynomial Mapping and Solver

This section discusses my method and expands its mathematical derivation for implementation in ISR workflows. The historical basis is the cubic polynomial structure used in RPC/RFM and the regression/solver foundations are least squares with regularization and Gauss–Newton-class nonlinear least squares, with key controversies being conditioning, overfitting, and inversion observability under poor geometry [8,11,9,10,17,24,25].

The following section describes the process of cubic polynomial mapping and the manner in which the solver operates. The discussion includes how recursive layers are applied and how each iteration refines the mapping to reduce residual error.

Before the section summary table, note that this is a surrogate model class; the competitor method is a rigorous physical sensor model and bundle adjustment, which can dominate accuracy when metadata exist but can be less deployable when metadata are denied or unreliable [12,11].

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Cubic polynomial surrogate in normalized variables	Standardized in RPC/ RFM form [8] analyzed as camera surrogate	Extrapolation risk; does not enforce physics	Rigorous sensor models and bundle adjustment [20], [21] bias-corrected RPC block [12]	Enables ISR mapping when metadata are incomplete or when empirical refinement is required

5. Normalization

This section stabilizes ISR regression and inversion by mapping inputs and outputs to dimensionless ranges, reflecting normalization conventions in imagery exploitation standards and numerical linear algebra practice; historically, RPC/RFM encode offsets and scales explicitly to keep normalized variables near [−1, 1] (NIMA 2000), and the controversy is that normalization defines the operational validity box and failure outside it is deterministic rather than stochastic.

Raw image-space coordinates (x, y) and elevation H are centered and scaled to stabilize regression:

$$x_n = \frac{x - x_0}{x_s} \quad y_n = \frac{y - y_0}{y_s} \quad H_n = \frac{H - H_0}{H_s}$$

Outputs (longitude, latitude, elevation) are likewise normalized:

$$\lambda_n = \frac{\lambda - \lambda_0}{\lambda_s} \quad \phi_n = \frac{\phi - \phi_0}{\phi_s} \quad \hat{H}_n = \frac{\hat{H} - H_0}{H_s}$$

Here $\hat{H}$ denotes the model's predicted elevation (when elevation is treated as an output to be solved), while H denotes an externally supplied elevation when known (e.g., DEM constraint).

Implementation requirements: Choose offsets as midpoints or means and scales as half-ranges so that typical values satisfy $|x_n| \le 1$, $|y_n| \le 1$, $|H_n| \le 1$. This aligns with RPC conventions that explicitly store offsets/scales for normalized coordinates (NIMA 2000). Hard failure occurs if any scale is zero. Soft failure (ill-conditioning) occurs if scales are extremely small relative to measurement precision, producing amplified quantization and roundoff.

Height convention requirement: Under CJCSI positional reference policy, the preferred height reference is WGS84 ellipsoid height unless mission dictates otherwise, and when geoid-referenced heights are used the associated EGM must be reported (Chairman of the Joint Chiefs of Staff 2023). If the training used orthometric heights (MSL), then H must be converted to ellipsoid height before normalization:

$$h_{\mathrm{ell}} = H_{\mathrm{MSL}} + N(\lambda, \phi).$$

The geoid undulation $N(\lambda, \phi)$ must be computed under a declared model (e.g., EGM2008) [16]. If $N(\lambda, \phi)$ is omitted or mismatched, a spatially varying bias is injected into Hn and therefore into (λ, ϕ) predictions through H-dependent terms.

Before the section summary table, note that normalization is a mission-level interoperability constraint, not only a numeric trick; datum/height mismatch is a known root cause of geolocation failure across coalition systems.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
$x_n = (x - x_0)/x_s$, $\lambda_n = (\lambda - \lambda_0)/\lambda_s$	2000 NIMA STDI-0002 normalization convention [8]	Defines a strict validity box; extrapolation is unstable	Orthogonal polynomial bases with better conditioning [42] spline/RBF warps [23]	Stabilizes coefficient estimation and evaluation for ISR exploitation systems
$h_{\text{ell}} = H_{\text{MSL}} + N(\lambda, \phi)$	Geodesy practice; EGM models (Pavlis et al. 2012)	Wrong geoid model yields spatial bias; mixed height definitions break mensuration	Explicit vertical datum pipelines [15]	Prevents height-convention error propagating into geolocation

6. Shared 3-Variable Cubic Basis

Here, let's define the explicit cubic monomial basis used for all channels, which operationally enables a compact surrogate consistent with common RPC/RFM structures; historically, the cubic polynomial structure in three normalized variables is the standard building block for RPC numerator/denominator polynomials (NIMA 2000; Tao and Hu 2001), while the controversy is that monomial bases are ill-conditioned compared to orthogonal alternatives and can overemphasize boundary behavior.

Let the 20-term cubic basis in (x_n, y_n, H_n) be

$$\mathbf{T}(x_n, y_n, H_n) = \begin{bmatrix} T_0 \\ T_1 \\ \vdots \\ T_{19} \end{bmatrix} = \begin{bmatrix} 1,\ x_n,\ y_n,\ H_n,\ x_n y_n,\ x_n H_n,\ y_n H_n,\ x_n^2,\ y_n^2,\ H_n^2, \\ x_n y_n H_n,\ x_n^3,\ x_n^2 y_n,\ x_n^2 H_n,\ x_n y_n^2,\ x_n H_n^2,\ y_n^3,\ y_n^2 H_n,\ y_n H_n^2,\ H_n^3 \end{bmatrix}^\top .$$

All three outputs $(\lambda_n, \phi_n, \hat{H}_n)$ are represented on this shared basis.

Combinatorial completeness: The basis enumerates all monomials of total degree ≤ 3 in three variables. The count is $\binom{3+3}{3} = 20$. This matters operationally: raising degree increases parameter count rapidly and increases GCP requirements and overfitting risk; the method instead increases expressive power by stacking layers on the same 20-term basis.

Ordering convention: RPC00B defines a specific term ordering for numerator/denominator coefficient storage (NIMA 2000). If interoperability with stored RPC coefficients is required, the basis ordering must be exactly matched. In this synthesis, the basis is fixed and shared across channels and layers, which makes tensor storage and SIMD evaluation straightforward.

If H_n is constant across all training samples, then any columns of $\mathbf{T}$ involving H_n become either constant multiples of other columns or constants, producing rank deficiency and non-identifiability of height-coupled coefficients. In ISR terrain that is nearly planar, this is a common condition; mitigation requires height-layered samples or imposing structural constraints (e.g., remove high-order H_n terms, or treat height as fixed and solve a 2D basis).

Before the section summary table, note that the major competitor at this "basis choice" level is not a different sensor model but a different function space: orthogonal polynomials, splines, and RBFs can dominate numeric stability when the validity box is large.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
20-term cubic monomial basis $\mathbf{T}(x_n, y_n, H_n)$	Standard in RPC/RFM cubic polynomials [8], [11]	Monomial conditioning; high correlation among terms	Orthogonal polynomial bases; spline/RBF warps [23]	Provides compact, interoperable surrogate aligned with imagery exploitation conventions

7. Recursive Residual Modeling (Stacked Polynomials)

This section defines the core original synthesis element: stagewise residual fitting on a fixed cubic basis to increase representational power without increasing polynomial degree; historically, sequential residual fitting is the backbone of boosting and stagewise additive models, while in photogrammetry "bias correction" and residual modeling are established practices for RPC refinement, and the controversy is controlled overfitting versus improved systematic-error capture [4,12].

I construct the mapping as a sum of cubic polynomials on the same basis, fitted recursively to residuals. For each output channel $q \in \{\lambda, \phi, H\}$ we write

$$q_n(x_n, y_n, H_n) = \sum_{m=0}^{M} \sum_{k=0}^{19} c_{m,k}^{(q)} \, T_k(x_n, y_n, H_n),$$

or, in vector form,

$$q_n(x_n, y_n, H_n) = \sum_{m=0}^{M} \left(\mathbf{c}_m^{(q)}\right)^{\top} \mathbf{T}(x_n, y_n, H_n), \qquad \mathbf{c}_m^{(q)} \in \mathbb{R}^{20}.$$

The coefficients are determined stage-by-stage. Let $\mathbf{q}_n \in \mathbb{R}^N$ be the normalized target values over N correspondences, and $\mathbf{T} \in \mathbb{R}^{N \times 20}$ be the design matrix whose i-th row is $\mathbf{T}(x_{n,i}, y_{n,i}, H_{n,i})^{\top}$.

Stage 0 fit:

$$\mathbf{T}\, \mathbf{c}_0^{(q)} \approx \mathbf{q}_n.$$

Residual after stage 0:

$$\mathbf{r}_1^{(q)} = \mathbf{q}_n - \mathbf{T}\, \mathbf{c}_0^{(q)}.$$

Stage 1 fit (on the residual):

$$\mathbf{T}\, \mathbf{c}_1^{(q)} \approx \mathbf{r}_1^{(q)}.$$

Continuing to stage m,

$$\mathbf{r}_m^{(q)} = \mathbf{q}_n - \sum_{j=0}^{m-1} \mathbf{T}\, \mathbf{c}_j^{(q)}, \qquad \mathbf{T}\, \mathbf{c}_m^{(q)} \approx \mathbf{r}_m^{(q)}.$$

After M stages the stacked solution is

$$\sum_{m=0}^{M} \mathbf{T}\, \mathbf{c}_m^{(q)} \approx \mathbf{q}_n \quad \Longleftrightarrow \quad q_n(x_n, y_n, H_n) = \sum_{m=0}^{M} \sum_{k=0}^{19} c_{m,k}^{(q)} \, T_k(x_n, y_n, H_n).$$

A Tikhonov-regularized normal equation is used at each stage:

$$\left(\mathbf{T}^{\top}\mathbf{T} + \lambda_{\text{reg}}\mathbf{I}\right) \mathbf{c}_m^{(q)} = \mathbf{T}^{\top}\, \mathbf{r}_m^{(q)}.$$

With $\mathbf{r}_0^{(q)} \equiv \mathbf{q}_n$, this yields the stacked coefficients by definition:

$$\mathbf{c}_{\Sigma}^{(q)}(M) = \sum_{m=0}^{M} \mathbf{c}_m^{(q)}.$$

Operator form (implementation-useful). Define the ridge "hat" operator

$$\mathbf{P} \doteq \mathbf{T}\big(\mathbf{T}^\top\mathbf{T} + \lambda_{\text{reg}}\mathbf{I}\big)^{-1}\mathbf{T}^\top \in \mathbb{R}^{N\times N}.$$

Then the stagewise residual update is

$$\mathbf{r}_{m+1} = (\mathbf{I} - \mathbf{P})\mathbf{r}_m, \qquad \mathbf{r}_0 = \mathbf{q}_n,$$

and the predicted outputs in data space after M stages satisfy

$$\hat{\mathbf{q}}^{(M)} = \sum_{m=0}^{M} \mathbf{T}\mathbf{c}_m = \sum_{m=0}^{M} \mathbf{P}\mathbf{r}_m = \big(\mathbf{I} - (\mathbf{I} - \mathbf{P})^{M+1}\big)\mathbf{q}_n.$$

This identity is the cleanest way to reason about convergence and overfitting: if the spectral radius $\rho(\mathbf{I} - \mathbf{P}) < 1$, then $\hat{\mathbf{q}}^{(M)} \to \mathbf{q}_n$ as $M \to \infty$, i.e., the procedure can drive training residuals toward zero even with ridge at each stage.

Operationally, this is a warning: training residual reduction is not equivalent to improved geolocation on new data.

The method introduces two coupled hyperparameters: layer count M and λ_{reg} . If M is large and λ_{reg} small, the procedure can behave like iterative de-regularization and can overfit measurement noise. If M is small and λ_{reg} large, systematic errors remain unmodeled. A defensible ISR implementation requires explicit selection criteria tied to heldout residuals or mission error budgets, not only training SSE reduction.

Before the section summary table, note that the true competitor to residual stacking is not "another polynomial," but either a single-pass ridge/OLS on the same basis or a different function family (splines/RBF/NN), and the controversy is largely about generalization and stability under domain shift.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Stagewise residual fit $\mathbf{r}_{m+1} = \mathbf{r}_m - \mathbf{T}\mathbf{c}_m$	Stagewise additive modeling lineage [4]	Can overfit; convergence to training data not mission success	Single-pass ridge/ OLS; boosting with shrinkage [4]	Captures systematic residual structure without increasing polynomial degree
Ridge stage solve $(T^\top T + \lambda_{\text{reg}} I)c_m = T^\top r_m$	Tikhonov regularization [17]; ridge regression [18]	Choice of λ_{reg}; bias vs variance	L-curve and related selection methods [42]	Stabilizes per-layer solutions under ISR tie-point noise
Data-space identity $\hat{q}^{(M)} = (I - (I-P)^{M+1})q$	Linear-operator consequence of fixed-basis residual fits	Makes overfitting explicit; shows potential to drive training error to zero	Early stopping; shrinkage; robust loss functions	Guides safe layer-count selection tied to mission validation

8. Coordinate-Wise Expansion

Here, let's make the channel-wise structure explicit for implementation and error budgeting, since ISR systems often require separate uncertainty analysis for latitude, longitude, and height; historically, channel separation is standard in RPC usage where longitude and latitude are modeled by separate rational functions (NIMA 2000), and the controversy is whether solving channels independently ignores cross-channel correlations that can matter in inversion.

Longitude (normalized):

$$\lambda_n(x_n, y_n, H_n) = \sum_{m=0}^{M} \sum_{k=0}^{19} a_{m,k}\, T_k(x_n, y_n, H_n).$$

Latitude (normalized):

$$\phi_n(x_n, y_n, H_n) = \sum_{m=0}^{M} \sum_{k=0}^{19} b_{m,k}\, T_k(x_n, y_n, H_n).$$

Elevation (normalized prediction):

$$\hat{H}_n(x_n, y_n, H_n) = \sum_{m=0}^{M} \sum_{k=0}^{19} d_{m,k}\, T_k(x_n, y_n, H_n).$$

The fully expanded scalar version makes the large-matrix structure explicit. Writing the first stage ($m = 0$) in full and eliding higher stages for brevity:

$$\begin{gathered}
\lambda_n = a_{0,0} + a_{0,1}x_n + a_{0,2}y_n + a_{0,3}H_n + a_{0,4}x_n y_n + a_{0,5}x_n H_n + a_{0,6}y_n H_n \\
+a_{0,7}x_n^2 + a_{0,8}y_n^2 + a_{0,9}H_n^2 + a_{0,10}x_n y_n H_n + a_{0,11}x_n^3 + a_{0,12}x_n^2 y_n + a_{0,13}x_n^2 H_n \\
+a_{0,14}x_n y_n^2 + a_{0,15}x_n H_n^2 + a_{0,16}y_n^3 + a_{0,17}y_n^2 H_n + a_{0,18}y_n H_n^2 + a_{0,19}H_n^3 \\
+ \sum_{m=1}^{M} \sum_{k=0}^{19} a_{m,k}\, T_k(x_n, y_n, H_n).
\end{gathered}$$

An identical expansion holds for ϕ_n and $\hat{H}_n$ with $b_{m,k}$ and $d_{m,k}$, respectively. Denormalization returns geodetic/ elevation units:

$$\lambda = \lambda_0 + \lambda_s\, \lambda_n, \qquad \phi = \phi_0 + \phi_s\, \phi_n, \qquad \hat{H} = H_0 + H_s\, \hat{H}_n.$$

Although channels may be fit independently, inversion couples them through the residual vector $[\lambda - \lambda^*, \phi - \phi^*]^\top$ (and optionally height residual). If channel scalings are inconsistent, inversion step conditioning degrades. Implementation should either normalize residuals by expected measurement noise or explicitly weight them.

Before the section summary table, note that channel-wise expansion competes with joint multi-output regression chiefly in inversion stability, not in forward evaluation cost.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
$\lambda = \lambda_0 + \lambda_s \sum_{m,k} a_{m,k} T_k(\cdot)$ (and analogs)	RPC/RFM channel separation convention [8]	Independent fits can ignore correlated errors; impacts inversion	Joint multi-output regression; rigorous sensor models [20]	Enables separate error budgeting for lat/lon/ height in ISR products remember CEP-like requirements [14]

9. Stacked Matrix View (All Channels)

Here is the correct linear-algebra form for fitting all channels and this clarifies a dimensional issue; its operational relevance is implementation correctness, and efficient batching (e.g., GPU evaluation). Historically multi-response regression is a straightforward extension of least squares, while the controversy is whether channel coupling is required for accuracy or only for inversion stability [10].

My original thoughts were:

$$\underbrace{\begin{bmatrix}\mathbf{T}\\ \mathbf{T}\\ \mathbf{T}\end{bmatrix}}_{\tilde{\mathbf{T}}\in\mathbb{R}^{3N\times 20}} \underbrace{\begin{bmatrix}\mathbf{c}_\Sigma^{(\lambda)} & \mathbf{c}_\Sigma^{(\phi)} & \mathbf{c}_\Sigma^{(H)}\end{bmatrix}}_{\mathbf{C}\in\mathbb{R}^{20\times 3}} \approx \underbrace{\begin{bmatrix}\boldsymbol{\lambda}_n\\ \boldsymbol{\phi}_n\\ \hat{\mathbf{H}}_n\end{bmatrix}}_{\tilde{\mathbf{q}}\in\mathbb{R}^{3N}}, \quad \mathbf{C} = \arg\min_{\mathbf{C}} \left\|\tilde{\mathbf{T}}\mathbf{C} - \tilde{\mathbf{q}}\right\|_2^2 + \lambda\|\mathbf{C}\|_F^2.$$

However to be a consistent multi-output formulation, let's define the output matrix

$$\mathbf{Q} \doteq \begin{bmatrix}\boldsymbol{\lambda}_n & \boldsymbol{\phi}_n & \hat{\mathbf{H}}_n\end{bmatrix} \in \mathbb{R}^{N\times 3},$$

and coefficient matrix

$$\mathbf{C} \doteq \begin{bmatrix}\mathbf{c}_\Sigma^{(\lambda)} & \mathbf{c}_\Sigma^{(\phi)} & \mathbf{c}_\Sigma^{(H)}\end{bmatrix} \in \mathbb{R}^{20\times 3}.$$

Then the correct regression statement is

$$\mathbf{TC} \approx \mathbf{Q}, \qquad \mathbf{C} = \arg\min_{\mathbf{C}} \|\mathbf{TC} - \mathbf{Q}\|_F^2 + \lambda_{\text{reg}}\|\mathbf{C}\|_F^2,$$

with closed-form ridge solution

$$\mathbf{C} = \left(\mathbf{T}^\top\mathbf{T} + \lambda_{\text{reg}}\mathbf{I}\right)^{-1}\mathbf{T}^\top\mathbf{Q}.$$

Vectorized (stacked) alternative. If a single vector equation is preferred, define $\tilde{\mathbf{T}} = \mathbf{I}_3 \otimes \mathbf{T} \in \mathbb{R}^{3N\times 60}$, $\tilde{\mathbf{c}} = \text{vec}(\mathbf{C}) \in \mathbb{R}^{60}$, and $\tilde{\mathbf{q}} = \text{vec}(\mathbf{Q}) \in \mathbb{R}^{3N}$. Then

$$\tilde{\mathbf{T}}\tilde{\mathbf{c}} \approx \tilde{\mathbf{q}}$$

and ridge regularization proceeds in the standard way.

I originally used λ both as longitude and as regularization; in this revision, I've changed the regularizer to λ_{reg}.

Before the section summary table, note that the primary competitor is independent per-channel regression; the joint formulation differs mainly in implementation efficiency and optional covariance modeling.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
$\mathbf{C} = (T^\top T + \lambda_{\text{reg}} I)^{-1} T^\top Q$	Least squares lineage [10] with Tikhonov [17]	Frobenius penalty assumes equal regularization across channels	Independent per-channel fits; weighted multi-output regression	Correct batching for ISR implementations and avoids dimension errors that silently corrupt geolocation

10. Solver with Unknown Elevation (Inversion)

This section defines the elevation inversion for ISR use cases where (x, y) are observed and a terrain/height parameter must be inferred to match geodetic coordinates; historically, nonlinear least squares uses Gauss–Newton-type iterations and their damped variants, while in imagery exploitation height is often the dominant residual driver under off-nadir geometry, and the controversy is observability and local convergence in nonphysical surrogates [24,25].

When (x, y) are given and elevation is unknown, we estimate H by minimizing geodetic residuals to target (λ^*, ϕ^*) while the model predicts (λ, ϕ):

$$r_\lambda = \lambda(x, y, H) - \lambda^*, \qquad r_\phi = \phi(x, y, H) - \phi^*.$$

Using the recursive polynomial maps,

$$\lambda(x, y, H) = \lambda_0 + \lambda_s \sum_{m,k} a_{m,k}\, T_k(x_n, y_n, H_n), \qquad \phi(x, y, H) = \phi_0 + \phi_s \sum_{m,k} b_{m,k}\, T_k(x_n, y_n, H_n).$$

Correct Jacobian shape (scalar parameter). Define the residual vector $\mathbf{r}(H) = [r_\lambda \;\; r_\phi]^\top \in \mathbb{R}^2$ and the Jacobian column vector

$$\mathbf{J}_H(H) = \frac{\partial \mathbf{r}}{\partial H} = \begin{bmatrix} \frac{\partial \lambda}{\partial H} \\ \frac{\partial \phi}{\partial H} \end{bmatrix} \in \mathbb{R}^{2\times 1}.$$

Then the Gauss–Newton step is scalar:

$$\Delta H = -\left(\mathbf{J}_H^\top \mathbf{J}_H\right)^{-1} \mathbf{J}_H^\top \mathbf{r}(H), \qquad H \leftarrow H + \Delta H.$$

The chain rule through normalization is

$$\frac{\partial H_n}{\partial H} = \frac{1}{H_s}, \qquad \frac{\partial}{\partial H} T_k(x_n, y_n, H_n) = \frac{\partial T_k}{\partial H_n} \frac{1}{H_s}.$$

The channel derivatives are

$$\frac{\partial \lambda}{\partial H} = \lambda_s \sum_{m,k} a_{m,k} \frac{\partial T_k}{\partial H_n} \frac{1}{H_s}, \qquad \frac{\partial \phi}{\partial H} = \phi_s \sum_{m,k} b_{m,k} \frac{\partial T_k}{\partial H_n} \frac{1}{H_s}.$$

Since T_k are monomials in (x_n, y_n, H_n), their derivatives are explicit, e.g.,

$$\frac{\partial}{\partial H_n}(H_n) = 1, \quad \frac{\partial}{\partial H_n}(x_n H_n) = x_n, \quad \frac{\partial}{\partial H_n}(H_n^3) = 3H_n^2, \quad \frac{\partial}{\partial H_n}(x_n y_n H_n) = x_n y_n, \text{ etc.}$$

Full derivative vector for the 20-term basis (implementation literal). Using the basis ordering in the original text, the derivatives $\partial T_k / \partial H_n$ are:

k	T_k	$\frac{\partial T_k}{\partial H_n}$
0	1	0
1	x_n	0
2	y_n	0
3	H_n	1
4	$x_n y_n$	0
5	$x_n H_n$	x_n
6	$y_n H_n$	y_n
7	x_n^2	0
8	y_n^2	0

9	H_n^2	$2H_n$
10	$x_n y_n H_n$	$x_n y_n$
11	x_n^3	0
12	$x_n^2 y_n$	0
13	$x_n^2 H_n$	x_n^2
14	$x_n y_n^2$	0
15	$x_n H_n^2$	$2x_n H_n$
16	y_n^3	0
17	$y_n^2 H_n$	y_n^2
18	$y_n H_n^2$	$2y_n H_n$
19	H_n^3	$3H_n^2$

Damping for field robustness. When $\mathbf{J}_H^\top \mathbf{J}_H$ is small (height weakly observable) or when nonlinearity is severe, use Levenberg–Marquardt damping:

$$\Delta H = -\left(\mathbf{J}_H^\top \mathbf{J}_H + \mu\right)^{-1} \mathbf{J}_H^\top \mathbf{r}(H), \qquad \mu > 0,$$

where μ is adjusted per iteration [24,25]. This is operationally relevant for near-nadir imagery or flat-terrain scenarios.

If H^* is also observed (e.g., from a DEM or LiDAR tie), a third residual $r_H = \hat{H}(x, y, H) - H^*$ can be appended to the objective with an appropriate weight to stabilize the solution.

Height inversion fails deterministically when $\frac{\partial \lambda}{\partial H} \approx 0$ and $\frac{\partial \phi}{\partial H} \approx 0$ over the feasible H range, because the residual surface becomes nearly flat and updates become numerically dominated by noise. It also fails when the surrogate mapping is non-monotone in H at fixed (x, y), creating multiple local minima; in that case bracketed search over H bounds or multi-start initialization is required.

Before the section summary table, note that the true competitor methods are terrain-ray intersection using rigorous sensor geometry and multi-view stereo/triangulation, which can make height observable without relying on a surrogate's height sensitivity.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Gauss–Newton $\Delta H = -(J^\top J)^{-1} J^\top r$	Newton/Gauss lineage formalized in nonlinear LS methods [24,25]	Local convergence; requires good initialization; observability issues	Levenberg–Marquardt damping [24,25] rigorous ray intersection	Enables ISR elevation inference when only single-image surrogate is available
Explicit $\partial T_k / \partial H_n$ list	Deterministic calculus; basis-specific	None; failure is numeric not analytic	Automatic differentiation engines	Enables stable, auditable solver implementation under constrained ISR compute

11. Solver with All Image Coordinates Unknown (2-D Inversion)

This section provides back-projection needed for chip-to-map tasks and georegistration refinement in ISR exploitation; historically, Newton-type methods for solving nonlinear systems are classical, and controversies center on multiple solutions and singular Jacobians when the mapping is locally non-invertible.

If (λ^*, ϕ^*, H^*) are known and (x, y) are unknown, update $\mathbf{u} = [x\ y]^\top$ using the 2×2 Jacobian

$$\mathbf{J}_{xy} = \begin{bmatrix} \frac{\partial \lambda}{\partial x} & \frac{\partial \lambda}{\partial y} \\ \frac{\partial \phi}{\partial x} & \frac{\partial \phi}{\partial y} \end{bmatrix}, \quad \Delta \mathbf{u} = -(\mathbf{J}_{xy}^\top \mathbf{J}_{xy})^{-1} \mathbf{J}_{xy}^\top \begin{bmatrix} \lambda(x, y, H^*) - \lambda^* \\ \phi(x, y, H^*) - \phi^* \end{bmatrix},$$

with the partials obtained by chain rule through x_n, y_n and the basis terms T_k.

Explicit chain rule form. With $x_n = (x - x_0) / x_s$ and $y_n = (y - y_0) / y_s$,

$$\frac{\partial x_n}{\partial x} = \frac{1}{x_s}, \qquad \frac{\partial y_n}{\partial y} = \frac{1}{y_s}, \qquad \frac{\partial x_n}{\partial y} = 0, \qquad \frac{\partial y_n}{\partial x} = 0.$$

Then for the longitude channel,

$$\frac{\partial \lambda}{\partial x} = \lambda_s \sum_{m,k} a_{m,k} \left(\frac{\partial T_k}{\partial x_n} \frac{1}{x_s} \right), \qquad \frac{\partial \lambda}{\partial y} = \lambda_s \sum_{m,k} a_{m,k} \left(\frac{\partial T_k}{\partial y_n} \frac{1}{y_s} \right),$$

and analogously for ϕ.

Back-projection fails when $\mathbf{J}_{xy}$ is singular or ill-conditioned, which occurs when the mapping locally collapses or when the surrogate contains flat regions in (x, y). Operationally, this corresponds to poorly constrained geometry, weak texture/tie points, or out-of-box evaluations. Damping (Levenberg–Marquardt) and bounded search in (x, y) can reduce divergence but do not create invertibility.

Before the section summary table, note that competitor methods include direct use of RPC inverse models where available, and geometric back-projection using rigorous models.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Newton/Gauss–Newton update in (x, y)	Classical nonlinear solve; modern photogrammetry uses analogous iterations	Multiple solutions; Jacobian singularities	Closed-form inverse for specific models; rigorous sensor back-projection	Supports ISR chip-to-map alignment and coordinate-to-pixel queries

12. Summary Expression (Large-Matrix Impression)

This section collects the complete forward mapping in a compact tensor form that is directly implementable and auditable; historically, stacking coefficients as structured arrays mirrors how RPC coefficients are stored and applied in exploitation systems (NIMA 2000), and the controversy is that compact forms can hide validity assumptions unless accompanied by explicit variable definitions and limits.

Collecting everything, the final expanded mapping from (x, y, H) to $(\lambda, \phi, \hat{H})$ under M recursive cubic refinements on the shared basis is

$$\begin{bmatrix} \lambda \\ \phi \\ \hat{H} \end{bmatrix} = \begin{bmatrix} \lambda_0 \\ \phi_0 \\ H_0 \end{bmatrix} + \begin{bmatrix} \lambda_s & 0 & 0 \\ 0 & \phi_s & 0 \\ 0 & 0 & H_s \end{bmatrix} \left(\sum_{m=0}^{M} \underbrace{\begin{bmatrix} \mathbf{a}_m^\top \\ \mathbf{b}_m^\top \\ \mathbf{d}_m^\top \end{bmatrix}}_{3\times 20} \underbrace{\mathbf{T}(x_n, y_n, H_n)}_{20\times 1} \right), \qquad \mathbf{a}_m, \mathbf{b}_m, \mathbf{d}_m \in \mathbb{R}^{20}.$$

Coefficient determination proceeds by repeated least-squares on successively smaller residuals,

$$(\mathbf{T}^\top\mathbf{T} + \lambda_{\text{reg}}\mathbf{I})\,\mathbf{a}_m = \mathbf{T}^\top\mathbf{r}_m^{(\lambda)}, \quad (\mathbf{T}^\top\mathbf{T} + \lambda_{\text{reg}}\mathbf{I})\,\mathbf{b}_m = \mathbf{T}^\top\mathbf{r}_m^{(\phi)}, \quad (\mathbf{T}^\top\mathbf{T} + \lambda_{\text{reg}}\mathbf{I})\,\mathbf{d}_m = \mathbf{T}^\top\mathbf{r}_m^{(H)},$$

with $\mathbf{r}_{m+1}^{(q)} = \mathbf{r}_m^{(q)} - \mathbf{T}\,\mathbf{c}_m^{(q)}$ and $\mathbf{r}_0^{(q)} \equiv \mathbf{q}_n$. This stacked formulation emphasizes the large, block-structured linear algebra underlying the recursive polynomial refinement for longitude, latitude, and elevation.

Although each layer is a cubic polynomial, the final mapping is still a cubic polynomial in (x_n, y_n, H_n) because all layers share the same basis; the "extra capacity" is not in function class but in the estimator path (iterative residual fitting) and its implicit filtering. This matters for claims of expressivity: the synthesis improves fit by changing estimation dynamics rather than by adding new basis functions.

Before the section summary table, note that the main competitor at this level is to directly solve a single ridge regression and stop; the controversy is whether recursive residual fitting provides better generalization under ISR noise and systematic error distributions.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Vector master map with tensor coefficients	RPC-inspired structured coefficient storage (NIMA 2000)	Recursive fitting changes estimator path; risk of overfitting	Single-pass ridge regression (Hoerl and Kennard 1970); boosting with shrinkage (Friedman 2001)	Provides auditable, batched forward evaluation for ISR pipelines

13. Fully Expanded Seven-Layer Recursive Polynomial (Longitude Channel)

This section explicitly uses expansion to make coefficient accounting and evaluation transparent for ISR implementation and audit; historically, explicit monomial expansions are standard for RPC coefficient application (NIMA 2000), and the controversy is parameter count versus the available control information, especially in denied environments with sparse GCPs.

If our convergence occurs at seven levels of polynomials, the breakout consists of 140 coefficients to derive the final answer. Since this is for one channel, the final x, y , z then requires 420 coefficients.

$$\begin{aligned}
\lambda = \lambda_0 + \lambda_s\Big[&(a_{0,0} + a_{0,1}x_n + a_{0,2}y_n + a_{0,3}H_n + a_{0,4}x_ny_n + a_{0,5}x_nH_n + a_{0,6}y_nH_n + a_{0,7}x_n^2 + a_{0,8}y_n^2 + a_{0,9}H_n^2 + a_{0,10}x_ny_nH_n + a_{0,11}x_n^3 + a_{0,12}x_n^2y_n + a_{0,13}x_n^2H_n + a_{0,14}x_ny_n^2 + a_{0,15}x_nH_n^2 + a_{0,16}y_n^3 + a_{0,17}y_n^2H_n + a_{0,18}y_nH_n^2 + a_{0,19}H_n^3)\\
&+(a_{1,0} + a_{1,1}x_n + a_{1,2}y_n + a_{1,3}H_n + a_{1,4}x_ny_n + a_{1,5}x_nH_n + a_{1,6}y_nH_n + a_{1,7}x_n^2 + a_{1,8}y_n^2 + a_{1,9}H_n^2 + a_{1,10}x_ny_nH_n + a_{1,11}x_n^3 + a_{1,12}x_n^2y_n + a_{1,13}x_n^2H_n + a_{1,14}x_ny_n^2 + a_{1,15}x_nH_n^2 + a_{1,16}y_n^3 + a_{1,17}y_n^2H_n + a_{1,18}y_nH_n^2 + a_{1,19}H_n^3)\\
&+(a_{2,0} + a_{2,1}x_n + a_{2,2}y_n + a_{2,3}H_n + a_{2,4}x_ny_n + a_{2,5}x_nH_n + a_{2,6}y_nH_n + a_{2,7}x_n^2 + a_{2,8}y_n^2 + a_{2,9}H_n^2 + a_{2,10}x_ny_nH_n + a_{2,11}x_n^3 + a_{2,12}x_n^2y_n + a_{2,13}x_n^2H_n + a_{2,14}x_ny_n^2 + a_{2,15}x_nH_n^2 + a_{2,16}y_n^3 + a_{2,17}y_n^2H_n + a_{2,18}y_nH_n^2 + a_{2,19}H_n^3)\\
&+(a_{3,0} + a_{3,1}x_n + a_{3,2}y_n + a_{3,3}H_n + a_{3,4}x_ny_n + a_{3,5}x_nH_n + a_{3,6}y_nH_n + a_{3,7}x_n^2 + a_{3,8}y_n^2 + a_{3,9}H_n^2 + a_{3,10}x_ny_nH_n + a_{3,11}x_n^3 + a_{3,12}x_n^2y_n + a_{3,13}x_n^2H_n + a_{3,14}x_ny_n^2 + a_{3,15}x_nH_n^2 + a_{3,16}y_n^3 + a_{3,17}y_n^2H_n + a_{3,18}y_nH_n^2 + a_{3,19}H_n^3)\\
&+(a_{4,0} + a_{4,1}x_n + a_{4,2}y_n + a_{4,3}H_n + a_{4,4}x_ny_n + a_{4,5}x_nH_n + a_{4,6}y_nH_n + a_{4,7}x_n^2 + a_{4,8}y_n^2 + a_{4,9}H_n^2 + a_{4,10}x_ny_nH_n + a_{4,11}x_n^3 + a_{4,12}x_n^2y_n + a_{4,13}x_n^2H_n + a_{4,14}x_ny_n^2 + a_{4,15}x_nH_n^2 + a_{4,16}y_n^3 + a_{4,17}y_n^2H_n + a_{4,18}y_nH_n^2 + a_{4,19}H_n^3)\\
&+(a_{5,0} + a_{5,1}x_n + a_{5,2}y_n + a_{5,3}H_n + a_{5,4}x_ny_n + a_{5,5}x_nH_n + a_{5,6}y_nH_n + a_{5,7}x_n^2 + a_{5,8}y_n^2 + a_{5,9}H_n^2 + a_{5,10}x_ny_nH_n + a_{5,11}x_n^3 + a_{5,12}x_n^2y_n + a_{5,13}x_n^2H_n + a_{5,14}x_ny_n^2 + a_{5,15}x_nH_n^2 + a_{5,16}y_n^3 + a_{5,17}y_n^2H_n + a_{5,18}y_nH_n^2 + a_{5,19}H_n^3)\\
&+(a_{6,0} + a_{6,1}x_n + a_{6,2}y_n + a_{6,3}H_n + a_{6,4}x_ny_n + a_{6,5}x_nH_n + a_{6,6}y_nH_n + a_{6,7}x_n^2 + a_{6,8}y_n^2 + a_{6,9}H_n^2 + a_{6,10}x_ny_nH_n + a_{6,11}x_n^3 + a_{6,12}x_n^2y_n + a_{6,13}x_n^2H_n + a_{6,14}x_ny_n^2 + a_{6,15}x_nH_n^2 + a_{6,16}y_n^3 + a_{6,17}y_n^2H_n + a_{6,18}y_nH_n^2 + a_{6,19}H_n^3)\Big].
\end{aligned}$$

Each parenthetical term represents a full cubic polynomial layer with its own coefficient set $(a_{m,k})$.
Summing across all seven layers explicitly shows the recursive accumulation of polynomial residuals that refine the mapping until convergence.

Before the section summary table, note that explicit expansions are primarily for audit and for fixed-function compiled implementations; competitor methods avoid this by storing coefficients and evaluating via basis-vector dot products.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Fully expanded seven-layer λ polynomial	Expansion style consistent with RPC polynomial evaluation [8]	Parameter count vs control information; risk of overfitting	Dot-product basis evaluation; bias correction models [12]	Enables deterministic ISR implementation and coefficient auditability

14. Master Equation Fully Expanded

Let's close with a single master equation that is fully expanded (seven layers, all channels) and is directly implementable; its historical context is that the cubic basis mirrors RPC/RFM polynomial structure while the multi-layer residual-estimation pathway borrows from stagewise residual fitting, and the primary controversy is that the final function remains cubic in (x_n, y_n, H_n) despite the recursive estimation, so the gain is estimator dynamics rather than function-class enlargement [8,11,4].

Define normalization (repeated here so the section is self-contained):

$$x_n = \frac{x - x_0}{x_s}, \qquad y_n = \frac{y - y_0}{y_s}, \qquad H_n = \frac{H - H_0}{H_s}.$$

Define the 20 monomials:

$$\begin{aligned}
&T_0 = 1,\ T_1 = x_n,\ T_2 = y_n,\ T_3 = H_n,\ T_4 = x_n y_n,\ T_5 = x_n H_n,\ T_6 = y_n H_n,\\
&T_7 = x_n^2,\ T_8 = y_n^2,\ T_9 = H_n^2,\ T_{10} = x_n y_n H_n,\\
&T_{11} = x_n^3,\ T_{12} = x_n^2 y_n,\ T_{13} = x_n^2 H_n,\ T_{14} = x_n y_n^2,\ T_{15} = x_n H_n^2,\\
&T_{16} = y_n^3,\ T_{17} = y_n^2 H_n,\ T_{18} = y_n H_n^2,\ T_{19} = H_n^3.
\end{aligned}$$

For seven layers (m = 0,…, 6), the fully expanded three-channel mapping is:

$$\lambda = \lambda_0 + \lambda_s \sum_{m=0}^{6} \left(\sum_{k=0}^{19} a_{m,k} T_k \right), \qquad \phi = \phi_0 + \phi_s \sum_{m=0}^{6} \left(\sum_{k=0}^{19} b_{m,k} T_k \right), \qquad \hat{H} = H_0 + H_s \sum_{m=0}^{6} \left(\sum_{k=0}^{19} d_{m,k} T_k \right).$$

Writing each channel explicitly as seven cubic layers:

$$\lambda = \lambda_0 + \lambda_s \Big[\Pi_0^{(\lambda)}(x_n, y_n, H_n) + \Pi_1^{(\lambda)}(x_n, y_n, H_n) + \Pi_2^{(\lambda)}(x_n, y_n, H_n) + \Pi_3^{(\lambda)}(x_n, y_n, H_n) + \Pi_4^{(\lambda)}(x_n, y_n, H_n) + \Pi_5^{(\lambda)}(x_n, y_n, H_n) + \Pi_6^{(\lambda)}(x_n, y_n, H_n) \Big],$$

$$\phi = \phi_0 + \phi_s \Big[\Pi_0^{(\phi)}(x_n, y_n, H_n) + \Pi_1^{(\phi)}(x_n, y_n, H_n) + \Pi_2^{(\phi)}(x_n, y_n, H_n) + \Pi_3^{(\phi)}(x_n, y_n, H_n) + \Pi_4^{(\phi)}(x_n, y_n, H_n) + \Pi_5^{(\phi)}(x_n, y_n, H_n) + \Pi_6^{(\phi)}(x_n, y_n, H_n) \Big],$$

$$\hat{H} = H_0 + H_s \Big[\Pi_0^{(H)}(x_n, y_n, H_n) + \Pi_1^{(H)}(x_n, y_n, H_n) + \Pi_2^{(H)}(x_n, y_n, H_n) + \Pi_3^{(H)}(x_n, y_n, H_n) + \Pi_4^{(H)}(x_n, y_n, H_n) + \Pi_5^{(H)}(x_n, y_n, H_n) + \Pi_6^{(H)}(x_n, y_n, H_n) \Big],$$

where each layer polynomial is fully expanded as

$$\begin{aligned}
\Pi_m^{(\lambda)} = {} & a_{m,0} + a_{m,1} x_n + a_{m,2} y_n + a_{m,3} H_n + a_{m,4} x_n y_n + a_{m,5} x_n H_n + a_{m,6} y_n H_n + a_{m,7} \\
& x_n^2 + a_{m,8} y_n^2 + a_{m,9} H_n^2 + a_{m,10} x_n y_n H_n + a_{m,11} x_n^3 + a_{m,12} x_n^2 y_n + a_{m,13} x_n^2 H_n + a_{m,14} \\
& x_n y_n^2 + a_{m,15} x_n H_n^2 + a_{m,16} y_n^3 + a_{m,17} y_n^2 H_n + a_{m,18} y_n H_n^2 + a_{m,19} H_n^3,
\end{aligned}$$

$$\begin{aligned}
\Pi_m^{(\phi)} = {} & b_{m,0} + b_{m,1} x_n + b_{m,2} y_n + b_{m,3} H_n + b_{m,4} x_n y_n + b_{m,5} x_n H_n + b_{m,6} y_n H_n + b_{m,7} \\
& x_n^2 + b_{m,8} y_n^2 + b_{m,9} H_n^2 + b_{m,10} x_n y_n H_n + b_{m,11} x_n^3 + b_{m,12} x_n^2 y_n + b_{m,13} x_n^2 H_n + b_{m,14} \\
& x_n y_n^2 + b_{m,15} x_n H_n^2 + b_{m,16} y_n^3 + b_{m,17} y_n^2 H_n + b_{m,18} y_n H_n^2 + b_{m,19} H_n^3,
\end{aligned}$$

$$\begin{aligned}
\Pi_m^{(H)} = {} & d_{m,0} + d_{m,1} x_n + d_{m,2} y_n + d_{m,3} H_n + d_{m,4} x_n y_n + d_{m,5} x_n H_n + d_{m,6} y_n H_n + d_{m,7} \\
& x_n^2 + d_{m,8} y_n^2 + d_{m,9} H_n^2 + d_{m,10} x_n y_n H_n + d_{m,11} x_n^3 + d_{m,12} x_n^2 y_n + d_{m,13} x_n^2 H_n + d_{m,14} \\
& x_n y_n^2 + d_{m,15} x_n H_n^2 + d_{m,16} y_n^3 + d_{m,17} y_n^2 H_n + d_{m,18} y_n H_n^2 + d_{m,19} H_n^3.
\end{aligned}$$

The forward evaluation cost is dominated by computing the 20 monomials once per input and then performing 7 dot products per channel (or one dot product with pre-summed coefficients if layers are summed offline). Summing coefficients offline produces a single 20-term coefficient vector per channel but discards layer-wise diagnostics; if layerwise behavior is operationally meaningful (e.g., to detect overfit), keep the layer tensor.

Before the section summary table, note that the master equation competes with RPC rational functions and rigorous models; the controversy is not that it cannot be implemented, but whether its empirical nature is acceptable for the required ISR error budgets.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods with dates and references	Key operational relevance
Fully expanded 3-channel, 7-layer cubic stack	Cubic basis aligned with RPC/RFM [8]; residual stacking lineage [4]	Function class remains cubic; gains come from estimator dynamics; risk of overfitting	RPC rational functions [8,11]; rigorous sensor models [20]	Provides deployable ISR geolocation surrogate with explicit audit trail

15. Variable Table (Consolidated)

This section ends with a consolidated variable table for direct implementation and to prevent ISR failure from symbol ambiguity; historically, RPC standards insist on explicit offset/scale parameters (NIMA 2000) and DoD policy insists on explicit coordinate/height reference conventions (Chairman of the Joint Chiefs of Staff 2023) [13].

Variable	Definition / Meaning	Units / Domain	Required?	Failure / ambiguity conditions
x, y	Image-space coordinates	pixels (typ.)	Yes	Must match training coordinate convention (pixel origin, axis directions)
H	Height variable used in basis	meters	Yes (forward); Unknown in height inversion	Height-type ambiguity (ellipsoid vs MSL) must be resolved
x_0, y_0, H_0	Normalization offsets	same as x, y, H	Yes	Wrong offsets shift domain; can cause extrapolation
x_s, y_s, H_s	Normalization scales (half-ranges)	same as x, y, H	Yes	Hard failure if any scale is 0
x_n, y_n, H_n	Normalized inputs	dimensionless	Derived	Extrapolation if $\lvert x_n \rvert > 1$ etc increases instability
λ, ϕ	Longitude, latitude	degrees	Output	Must be tied to declared datum (WGS84 assumed in most ISR systems)
λ_0, ϕ_0	Output offsets	degrees	Yes	Wrong offsets cause systematic shifts
λ_s, ϕ_s	Output scales	degrees	Yes	Wrong scales distort geolocation
$\hat{H}$	Predicted height output	meters	Optional output	Must be defined as same height type as H unless stated otherwise

$\mathbf{T}(x_n, y_n, H_n)$	20-term basis vector	dimensionless	Derived	Ordering must be fixed and consistent
T_k	k-th monomial term	dimensionless	Derived	Basis degenerates if H_n or other variables constant
m	Layer index	$\{0, \ldots, 6\}$	Design choice	Too large risks overfit; too small leaves systematic error
k	Basis-term index	$\{0, \ldots, 19\}$	Fixed	Ordering mismatch breaks coefficient interpretation
$a_{m,k}, b_{m,k}, d_{m,k}$	Coefficients for $\lambda, \phi, \hat{H}$ layers	dimensionless	Learned	Ill-conditioned if training design matrix is rank-deficient
λ_{reg}	Ridge/Tikhonov parameter	scalar ≥ 0	Training hyperparameter	Too small: instability/ overfit; too large: underfit
N	Number of correspondences	integer	Training	Must be sufficiently large and distributed; height variation required
$\mathbf{r}(H)$	Residual vector for inversion	$\mathbb{R}^2$	In inversion	Becomes flat if height unobservable
$\mathbf{J}_H(H)$	Jacobian of residual w.r.t H	$\mathbb{R}^{2\times 1}$	In inversion	Near-zero causes divergence; require damping or constraints
$N(\lambda, \phi)$	Geoid undulation	meters	If converting MSL↔ellipsoid	Must specify EGM model; mismatch injects spatial bias

16. Empirical Test Results and Comparative Performance

This section records comparative accuracy outcomes for ISR geolocation and tracking, with emphasis on the operational trade between non-causal post-processing and causal real-time maneuver tracking rooted in recursive state estimation and multiple-model maneuver handling, noting the long-standing controversy that offline smoothers and batch interpolants are not operationally equivalent to real-time trackers under latency and maneuver constraints [26-29].

Rudolf E. Kalman (1930–2016) introduced the Kalman filter as a recursive optimal estimator for linear systems under Gaussian noise, establishing the modern state-space view used in navigation and tracking [26]. Henk A. P. Blom is a Dutch control and tracking researcher associated with multiple-model estimation for switching systems, and Yaakov Bar-Shalom is a central figure in target tracking and data association whose work underpins widely deployed clutter/maneuver trackers in defense systems [27,30]. Claude E. Shannon (1916–2001) formalized information entropy; “mode entropy” is a direct application of Shannon entropy to the IMM mode probability vector [31].

17. Conclusion

For maximum accuracy in post-processing scenarios, the Fourier-Space Spline with Cubic Polynomial Recursion delivers sub-meter performance unmatched by any other method tested. For real-time tracking with maneuvers, IMM-based filters, especially with Mode Entropy or Model Transition augmentation, provide the best balance of accuracy and computational feasibility. The analysis demonstrates that combining frequency-domain interpolation with systematic residual compensation can achieve accuracy levels (0.16m) that approach typical GPS measurement noise floors, suggesting this represents a near-optimal solution for the given sensor precision [32-35].

Rank	Method	Overall Error (m)	Category	vs Kalman
1	★ Fourier-Space Spline + Cubic Poly Recursion (8 levels)	0.16	Post-proc + Recursive	98.40%
2	Fourier-Space Spline	1.2	Post-proc	88%
3	Cubic Spline	2.2	Post-proc	78%
4	Piecewise Linear	2.8	Post-proc	71%
5	IMM + Mode Entropy	4.4	Augmented	55%
6	IMM + Model Transition	4.6	Augmented	53%
7	IMM + Model Probability	4.6	Augmented	53%
8	IMM Filter	4.9	Real-time	50%
9	Multi-Bernoulli Filter	5.1	RFS	48%
10	PF + ESS Resample	5.2	Augmented	47%
11	Bernoulli Filter	5.4	RFS	45%
12	PF + Weight Degeneracy	5.4	Augmented	45%
13	Kalman + Jerk Constraint	5.8	Augmented	41%
14	UKF + Posterior Mode	5.8	Augmented	41%
15	Particle Filter	5.8	Real-time	41%
16	Kalman + Kinematic Feasibility	5.8	Augmented	41%
17	Kalman + Composite Quality	6.2	ML-Aug	37%
18	Kalman + Accel Maneuver	6.3	Augmented	36%
19	UKF	6.5	Real-time	34%
20	MHT + Assoc Entropy	6.5	Augmented	34%
21	Kalman + Bayesian Neural	6.6	ML-Aug	33%
22	MHT + LR Pruning	6.7	Augmented	32%
23	Kalman + MC Dropout	6.7	ML-Aug	32%
24	Kalman + Predictive Entropy	6.8	ML-Aug	31%
25	Kalman + Mutual Information	6.8	Fusion	31%
26	Cross-Sensor Innovation	6.9	Fusion	30%
27	Dempster-Shafer Fusion	6.9	ML-Aug	30%
28	MHT	7	Real-time	29%
29	Kalman + Feature Inlier	7	Vision	29%
30	Kalman + Tracklet Confidence	7	ML-Aug	29%
31	Kalman + Sensor Agreement	7.1	Fusion	28%
32	Kalman + Optical Flow	7.3	Vision	26%
33	Kalman + Softmax Assign	7.3	ML-Aug	26%
34	Kalman + Adaptive R	7.4	Sensor	24%

35	Kalman + NIS Adaptive	7.5	Augmented	23%
36	Kalman + OOD Gating	7.5	ML-Aug	23%
37	Kalman + Sensor Credibility	7.6	Sensor	22%
38	Kalman + Re-ID Score	7.6	Vision	22%
39	Kalman + Likelihood Wt	7.7	Augmented	21%
40	JPDA	7.7	Real-time	21%
41	Kalman + Adaptive Gating	7.8	Augmented	20%
42	Kalman + CFAR Score	7.9	Sensor	19%
43	Kalman + Redundancy Count	7.9	Fusion	19%
44	EKF	8	Real-time	18%
45	Kalman + SNR Weighting	8.1	Sensor	17%
46	Kalman + Track Score	8.1	Augmented	17%
47	Kalman + Detector Conf	8.1	Sensor	17%
48	Kalman + NCC Weighting	8.1	Vision	17%
49	PDA	8.4	Real-time	14%
50	Information Filter	9.7	Real-time	1%
51	Kalman Filter	9.8	Real-time	Baseline
52	Dead Reckoning	12.7	Real-time	-30%

The ranking is dominated by a causality divide. The spline-based methods are post-processing constructs: they can exploit the full trajectory, enforce global smoothness, and absorb low-frequency bias components that are not representable in a strictly causal filter without delay or smoothing. The IMM family is the strongest real-time performer in this test set; this is consistent with IMM's design objective of approximating optimal Bayesian filtering for Markov-switching maneuver models at tractable cost [27]. The RFS filters (Bernoulli and multi-Bernoulli) originate in Mahler's Finite Set Statistics (FISST) formulation; they are structurally targeted at multi-object uncertainty and data association rather than single-track smoothing, so a single-track scalar "overall error" comparison can be methodologically mismatched unless the evaluation explicitly accounts for track continuity and cardinality [36,37].

The post-processing delta between "Fourier-Space Spline" (1.2 m) and "Fourier-Space Spline + Cubic Poly Recursion" (0.16 m) corresponds to a factor reduction of **1.2/0.16 = 7.5** in the reported scalar error. If the error metric is RMSE over Euclidean position error, this implies a large reduction in residual energy, consistent with a two-stage bias model in which a smooth low-frequency map is followed by structured residual correction. Our claim that 0.16 m "approaches typical GPS measurement noise floors" is only meaningful when the GNSS measurement type is known (code vs carrier, differential vs standalone) and when "noise floor" is tied to the same statistic (measurement noise vs final position error), since decimeter-scale can be either noise-limited or still well above carrier-phase-limited regimes depending on system and environment [32,33,38].

A reproducible comparison requires a single operational definition of "Overall Error (m)." A defensible default in ISR geolocation testing is the trajectory RMSE in a common Cartesian frame:

$$e_t = \|\hat{\mathbf{p}}_t - \mathbf{p}_t\|_2, \qquad E_{\mathrm{RMSE}} = \sqrt{\frac{1}{T}\sum_{t=1}^{T} e_t^2}.$$

If the evaluation instead uses mean absolute error $E_{\mathrm{MAE}} = \frac{1}{T}\sum_{t=1}^{T} e_t$, or uses a circular error probable metric (CEP) commonly used in weapon/targeting performance reporting, the numerical ranking can change for heavy-tailed error distributions and for intermittent outliers; this is a known controversy in ISR performance reporting because RMSE weights outliers quadratically while CEP explicitly targets quantiles [28].

The “vs Kalman” percentage is consistent with a scalar improvement ratio of the form

$$\Delta_{\%} = 100\left(1 - \frac{E}{E_{\mathrm{KF}}}\right), \qquad E_{\mathrm{KF}} = 9.8\ \mathrm{m},$$

with sign reversal for degradations (negative values) such as dead reckoning. Minor rounding inconsistencies at the < 0.1% level are expected if the table reports rounded errors while percentages are computed on unrounded errors.

I am well aware, that method family comparisons are not intrinsically “fair” across categories. Post-processing methods implicitly approximate fixed-interval smoothing, for which the canonical linear-Gaussian reference is the Rauch–Tung–Striebel (RTS) smoother; an ISR-relevant competitor set would therefore include RTS smoothing as a post-processing baseline alongside spline interpolation, since RTS is a strict Bayesian smoother under linear-Gaussian assumptions [28]. Real-time maneuver tracking comparisons should similarly pin down the maneuver model class (constant velocity, constant acceleration, jerk, Singer acceleration model) because performance differences can be dominated by model mismatch rather than filter form [39,27]. Without these definitions, “near-optimal” should be treated as a hypothesis rather than a proven bound statement; if near-optimality is desired as a claim, the standard path is to report a Cramér–Rao lower bound (CRLB) for the chosen measurement model and compare achieved RMSE to the bound [34,35,40]. As such, rigorous test work in standardized units remains to be done.

A small set of explicit failure conditions follows directly from the method structure. The post-processing spline plus residual recursion can fail by extrapolation if mission geometry leaves the training domain (e.g., targets outside the convex hull of the fitted track), by aliasing if the fitted bandwidth is too low for a maneuver-rich trajectory, and by overfitting if recursion depth is increased without cross-validated regularization. Real-time IMM variants can fail under incorrect mode-transition priors (transition matrix mismatch), under near-singular observability when measurement geometry collapses, and under ambiguity bursts when clutter or dropouts drive mode probabilities toward uniformity; “mode entropy” is diagnostically useful here because high Shannon entropy indicates no mode is decisively supported [31,27].

Before the summary table, the central basis is that the reported 0.16 m regime is consistent with a bias-limited error profile in which low-frequency distortion plus structured residuals dominate; the real-time regime is dominated by model-switching fidelity and measurement association robustness rather than by global interpolation smoothness. There is no intent to confuse these two.

Key equation(s)	Date and origin (researcher, institution)	Major controversies or critiques	True competitor methods (dates, references)	Key operational relevance
$E_{\mathrm{RMSE}} = \sqrt{\frac{1}{T}\sum_{t=1}^{T} \lVert \hat{\mathbf{p}}_t - \mathbf{p}_t \rVert_2^2}$	RMSE as a standard quadratic risk statistic; formal statistical treatment in early 20th-century estimation theory [40,34]	RMSE is outlier-sensitive; CEP/ quantile metrics can be more operationally aligned for ISR/ targeting error distributions with heavy tails or intermittent dropouts [28]	CEP-style quantile metrics used in weapons/targeting performance reporting; robust losses [44] are a statistical alternative to quadratic loss	Provides a single scalar to rank geolocation/tracking methods in a mission-relevant unit (meters)
$\Delta_{\%} = 100\left(1 - \frac{E}{E_{\mathrm{KF}}}\right)$	Relative improvement reporting is a standard normalization practice in estimator benchmarking	Misleading if baselines differ in causality class (post-proc vs real-time) or if E is not the same statistic across methods	Category-separated baselines: RTS smoother for post-proc [28]; IMM as maneuver real-time baseline [27]	Enables quick ISR trade decisions when a Kalman-class baseline is mandated for compute/latency reasons

IMM filtering with mode probability vector $\boldsymbol{\pi}$ and mode entropy $H_{\text{mode}} = -\sum_j \pi_j \log \pi_j$	IMM: Blom and Bar-Shalom, IEEE TAC, multiple-model switching for maneuvering targets [27]; entropy: Shannon, Bell System Technical Journal [31]	IMM is prior-sensitive: transition matrix and model set dominate performance; entropy augmentation is diagnostic unless explicitly coupled to adaptation logic	Multiple-model variable-structure filters particle filters for nonlinear/non-Gaussian regimes [43]	Best real-time balance in maneuver regimes under constrained compute, consistent with ISR tracking pipelines
Bernoulli / multi-Bernoulli RFS filtering	RFS/FISST: Mahler, Artech House [36] ;CBMeMBer: Vo, Vo, and Cantoni, IEEE TSP [37]	RFS filters target multi-object state and cardinality; single-track scalar error can understate their operational advantage in track continuity, association robustness, and false track control	MHT for multi-hypothesis association [46]; JPDA/PDA for probabilistic association [30,29]	Relevant when ISR scenes contain multiple closely spaced movers with ambiguous measurement origin
Post-processing spline/interpolation + residual recursion (FSS-RPR)	Spline approximation theory: Schoenberg [2]; computational splines: de Boor [3]; signal-processing view: Unser [6]; stagewise residual fitting: Friedman [4]; FFT computation: Cooley and Tukey [5]	Not causal; performance can be dominated by domain coverage and regularization; recursion depth increases overfitting risk without cross-validated controls	Fixed-interval Bayesian smoothing (RTS) as the linear-Gaussian post-proc competitor [28]	Maximizes post-mission georegistration accuracy for ISR exploitation and mensuration when offline compute is available

References

1. baron de Fourier, J. B. J. (1822). *Théorie analytique de la chaleur*. Firmin Didot.
2. Schoenberg, I. J. (1946). Contributions to the problem of approximation of equidistant data by analytic functions. Part A. On the problem of smoothing or graduation. A first class of analytic approximation formulae. *Quarterly of applied mathematics, 4*(1), 45-99.
3. De Boor, C., & De Boor, C. (1978). *A practical guide to splines* (Vol. 27, p. 325). New York: Springer.
4. Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.
5. Cooley, J. W., & Tukey, J. W. (1965). An algorithm for the machine calculation of complex Fourier series. Mathematics of computation, 19(90), 297-301.
6. Unser, M. (1999). Splines: A perfect fit for signal and image processing. *IEEE Signal Processing Magazine, 16*(6), 22-38
7. Wahba, G. (1990). *Spline models for observational data*. Society for industrial and applied mathematics.
8. National Imagery and Mapping Agency. 2000. STDI-0002 Version 2.1: Compendium of Controlled Extensions for the National Imagery Transmission Format Standard (NITFS). Washington, DC: NIMA.
9. Legendre, A. M. (1806). Nouvelles méthodes pour la détermination des orbites des comètes; par AM Legendre... chez Firmin Didot, libraire pour lew mathematiques, la marine, l'architecture, et les editions stereotypes, rue de Thionville.
10. Gauss, C. F. (1809). Theoria motus corporum coelestum. Werke.
11. Tao, C. V., & Hu, Y. (2001). A comprehensive study of the rational function model for photogrammetric processing. Photogrammetric engineering and remote sensing, 67(12), 1347-1358.

12. Grodecki, J., & Dial, G. (2003). Block adjustment of high-resolution satellite images described by rational polynomials. *Photogrammetric Engineering & Remote Sensing, 69*(1), 59-68.
13. Chairman of the Joint Chiefs of Staff. 2023. CJCSI 3900.01E: Position (Point and Area) Reference Procedures. Washington, DC: Joint Chiefs of Staff.
14. Joint Chiefs of Staff. 2018. Joint Publication 3-60: Joint Targeting. Washington, DC: Joint Chiefs of Staff.
15. National Geospatial-Intelligence Agency. 2014. NGA.STND.0036_1.0.0: Department of Defense World Geodetic System 1984—Its Definition and Relationships with Local Geodetic Systems. Springfield, VA: NGA Office of Geomatics.
16. Pavlis, N. K., Holmes, S. A., Kenyon, S. C., & Factor, J. K. (2012). The development and evaluation of the Earth Gravitational Model 2008 (EGM2008). *Journal of geophysical research: solid earth, 117*(B4).
17. Tikhonov, A. N. (1963). Solution of incorrectly formulated problems and the regularization method. Sov Dok, 4, 1035-1038.
18. Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55-67.
19. Hartley, R. I., & Saxena, T. (1997, May). The cubic rational polynomial camera model. In *Image understanding workshop* (Vol. 649, p. 653).
20. Brown, D. C. (1971). Close-range camera calibration. Photogrammetric engineering, 37(8), 855-866.
21. Triggs, B., McLauchlan, P. F., Hartley, R. I., & Fitzgibbon, A. W. (1999, September). Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms* (pp. 298-372). Berlin, Heidelberg: Springer Berlin Heidelberg.
22. Besl, P. J., & McKay, N. D. (1992, April). Method for registration of 3-D shapes. In *Sensor fusion IV: control paradigms and data structures* (Vol. 1611, pp. 586-606). Spie.
23. Bookstein, F. L. (2002). Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on pattern analysis and machine intelligence, 11*(6), 567-585.
24. Levenberg, K. (1944). A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics, 2*(2), 164-168.
25. Marquardt, D. W. (1963). An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics, 11*(2), 431-441.
26. Kalman, R. E. (1960). A new approach to linear filtering and prediction problems.
27. Blom, H. A., & Bar-Shalom, Y. (1988). The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE transactions on Automatic Control, 33*(8), 780-783.
28. Rauch, H. E., Tung, F., & Striebel, C. T. (1965). Maximum likelihood estimates of linear dynamic systems. *AIAA journal, 3*(8), 1445-1450.
29. Bar-Shalom, Y., Fortmann, T. E., & Tracking and Data Association. (1988). Academic Press: Boston. MA, USA, 179, 1988.
30. Bar-Shalom, Y., & Tse, E. (1975). Tracking in a cluttered environment with probabilistic data association. *Automatica, 11*(5), 451-460.
31. Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal, 27*(3), 379-423.
32. Misra, P., & Enge, P. (2002). Global positioning system: Signals, measurements, and Performance. *IEEE aerospace and electronic systems magazine, 17*(10), 36-37.
33. Kaplan, E. D., & Hegarty, C. (Eds.). (2017). Understanding GPS/GNSS: principles and applications. Artech house.
34. Cramer, H. (1946). Mathematical methods of statistics, Princeton, 1946. Math Rev (Math-SciNet) MR16588 Zentralblatt MATH, 63, 300.
35. Rao, C. R. (1945). Information and the accuracy attainable in the estimation of statistical parameters. Bull. *Calcutta Math. Soc, 37*(3), 81-91.
36. Mahler, R. (2007). Statistical multisource-multitarget information fusion. Artech.
37. Vo, B. T., Vo, B. N., & Cantoni, A. (2008). The cardinality balanced multi-target multi-Bernoulli filter and its implementations. *IEEE transactions on signal processing, 57*(2), 409-423.
38. Parkinson, B. W. (1996). Global positioning system: Theory and applications i. Progress in Astronautics and aeronautics, 163, 3-55.
39. Singer, R. A. (2007). Estimating optimal tracking filter performance for manned maneuvering targets. *IEEE Transactions on Aerospace and electronic systems,* (4), 473-483.
40. Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. Philosophical transactions of the Royal Society of London. *Series A, containing papers of a mathematical or physical character, 222*(594-604), 309-368.
41. Abdel-Aziz, Y. I., & Karara, H. M. (1971). Direct linear transformation from comparator coordinates into object space coordinates. In Proceedings of the ASP/UI Symposium on Close-Range Photogrammetry. Urbana, IL. American Society of Photogrammetry.
42. Golub, G. H., & Van Loan, C. F. (2013). *Matrix computations*. JHU press.
43. Gordon, N. J., Salmond, D. J., & Smith, A. F. (1993, April). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *IEE proceedings F (radar and signal processing)* (Vol. 140, No. 2, pp. 107-113). IEE.
44. Huber, P. J. (1992). Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution* (pp.

492-518). New York, NY: Springer New York.

45. Julier, S. J., & Uhlmann, J. K. (1997, July). New extension of the Kalman filter to nonlinear systems. In *Signal processing, sensor fusion, and target recognition VI* (Vol. 3068, pp. 182-193). Spie.
46. Reid, D. (2003). An algorithm for tracking multiple targets. *IEEE transactions on Automatic Control, 24*(6), 843-854.