GCorP: Global Correlation Pooling

Anonymous authors
Paper under double-blind review

Abstract

Global covariance pooling provides expressive second-order representations, but existing designs primarily operate on covariance matrices and associated matrix functions. We introduce *Global Correlation Pooling* (GCorP), which turns covariance into a correlation matrix encoding pairwise channel dependencies and maps this matrix to geometry-specific flat or hyperbolic spaces. For flat geometries, we prove an exact Euclidean realization of non-trivialized correlation-manifold multinomial logistic regression: its intrinsic logits become an FC layer, and Euclidean SGD on the induced flat parameters reproduces the corresponding Riemannian SGD trajectory. We evaluate GCorP on image classification, EEG foundation-model classification, and generative-model post-training. Across these tasks, GCorP offers a favorable accuracy–efficiency trade-off and performs well .

1 Introduction

Global pooling aggregates the feature vectors at all spatial locations into a fixed-dimensional representation for downstream prediction. Global average pooling is simple and efficient, but it replaces each channel's spatial responses by their mean and therefore retains only first-order channel statistics. Bilinear and global covariance pooling additionally summarize how channel responses vary together, yielding expressive symmetric positive-definite representations for recognition (Lin et al., 2015; Ionescu et al., 2015; Wang et al., 2021). Their non-Euclidean geometry is commonly handled by a matrix function before vectorization and classification (Arsigny et al., 2006; Pennec et al., 2006; Bhatia, 2007).

Existing covariance-pooling methods mainly improve this matrix-function step and its computation. Matrix square-root and power transformations improve covariance representations (Lin & Maji, 2017; Wang et al., 2021); iSQRT-COV, MPA, SVD-Padé, and Halley-SVD reduce the cost of the associated matrix functions (Li et al., 2018; Song et al., 2022a; 2021; Gu et al., 2025). These approaches retain a covariance-derived representation. ISICE instead forms partial correlation from a sparse inverse covariance, a different statistic that measures the relationship between two channels after accounting for the others (Rahman et al., 2023).

Full-rank correlation matrices provide an alternative to covariance matrices (Chen et al., 2026): they standardize marginal variances while retaining pairwise channel dependencies. Existing correlation networks develop geometric layers for correlation-valued inputs, but correlation matrices remain underexplored as global pooling representations produced directly from backbone features. Moreover, they are not ordinary Euclidean vectors, so their use in comparison, classification, or distance-based learning requires suitable geometric operations. Recent correlation geometries provide tractable Euclidean or hyperbolic spaces for these operations (Thanwerdas & Pennec, 2022; Chen et al., 2026), opening a path from correlation-valued inputs to end-to-end global pooling.

We therefore introduce *Global Correlation Pooling* (GCorP), which forms C from backbone features and maps it to a prototype space associated with a correlation geometry. ECM and LECM use lower-triangular Cholesky diffeomorphisms, OLM and LSM provide alternative flat diffeomorphisms, and PHCM uses Poincaré or Lorentz representations. These alternatives expose the same correlation matrix to Euclidean or hyperbolic classifiers and distance-based objectives. For the four flat maps, we formulate CorMLR through class-dependent base points and tangent vectors, then prove an exact Euclidean realization: intrinsic CorMLR logits reduce to an FC layer, and Euclidean SGD on the induced flat parameters reproduces the corresponding Riemannian SGD trajectory. For the logarithmic maps, we use the SVD-free orthogonal-polynomial matrix-logarithm approximation of Rahman et al.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(2026). We evaluate the same pooling interface in image classification, generative-model post-training, and EEG foundation-model evaluation.

Our contributions are threefold. **(i)** We formulate global correlation pooling as a second-order representation of pairwise channel dependencies, bringing full-rank correlation geometry into end-to-end pooling. **(ii)** We instantiate this representation with ECM, LECM, OLM, LSM, and PHCM diffeomorphisms and show that the four flat heads are exact Euclidean realizations of non-trivialized CorMLR: their FC logits match intrinsic CorMLR logits, and Euclidean SGD on the induced flat parameters reproduces the corresponding Riemannian SGD. **(iii)** We validate the effectiveness of GCorP through experiments on image recognition, EEG classification, and generative-model post-training.

2 Related Work

Covariance and high-order pooling. Bilinear and covariance pooling retain second-order channel statistics, while high-order pooling extends them to richer feature interactions (Lin et al., 2015; Gao et al., 2019; Xie et al., 2021). Matrix functions and their efficient approximations operate on covariance representations (Lin & Maji, 2017; Li et al., 2018; Song et al., 2021; Gu et al., 2025). Their effectiveness has also been studied through spectral, optimization, and Riemannian interpretations (Wang et al., 2021; 2023; Chen et al., 2025). These methods primarily transform the covariance descriptor itself. GCorP instead first converts channel covariance into a full-rank correlation matrix and then applies a map selected for its correlation geometry, separating marginal channel scale from the modeled cross-channel dependence.

Correlation geometries and Riemannian networks. Full-rank correlation matrices admit tractable metrics with closed-form geodesics and logarithm maps (Thanwerdas & Pennec, 2022). The off-log and log-scaled diffeomorphisms used by GCorP build on correlation-matrix parametrization and diagonal rescaling (Archakov & Hansen, 2021; Thanwerdas, 2024). Riemannian networks over correlation matrices extend MLR and fully connected layers to five geometries (Chen et al., 2026). Whereas these networks assume correlation-valued inputs and develop layers acting on them, GCorP constructs correlations directly from backbone feature maps as a global pooling representation. It then exposes the resulting geometry-specific representations to either classification heads or distance-based objectives, and relates its flat heads to CorMLR with class-specific base points rather than a common tangent-space origin.

Hyperbolic and biosignal representations. Poincaré and Lorentz classifiers provide the hyperbolic operations used by PHCM (Ganea et al., 2018; Shimizu et al., 2021; Bdeir et al., 2024). Covariance-based Riemannian methods also support EEG classification and recent foundation-model settings (Barachant et al., 2013; Jiang et al., 2024; Hu et al., 2026). GCorP uses one correlation-pooling interface across these classifiers and distance-based objectives. Appendix F provides an extended discussion.

3 Preliminaries

3.1 Full-Rank Correlation Geometries

Notation. For Euclidean vectors and matrices, $\langle \cdot, \cdot \rangle$ denotes the standard inner product, Frobenius for matrices, and $\| \cdot \|$ its induced norm. On a Riemannian manifold $\mathcal{M}$, Log_P, Exp_P, and $\langle \cdot, \cdot \rangle_P$ denote the logarithm map, exponential map, and metric inner product at $P \in \mathcal{M}$.

Given $\Sigma \in \mathrm{SPD}(n)$, its correlation matrix is $C = \mathrm{Cor}(\Sigma) = \mathrm{Diag}(\Sigma)^{-1/2} \Sigma \, \mathrm{Diag}(\Sigma)^{-1/2}$. The full-rank correlation matrices form $\mathrm{Cor}^{+}(n) = \{C \in \mathrm{SPD}(n) \mid \mathrm{diag}(C) = \mathbf{1}\}$, whose tangent space is identified with $\mathrm{Hol}(n)$, the symmetric hollow matrices. Following CoRNet (Chen et al., 2026), GCorP uses four flat pullback geometries and the hyperbolic PHCM geometry on $\mathrm{Cor}^{+}(n)$.

Prototype spaces and inner products. The flat prototype spaces are $\mathrm{LT}_0(n)$, the lower-triangular matrices with zero diagonal, $\mathrm{Hol}(n)$, and $\mathrm{Row}_0(n)$, the symmetric matrices with zero row sum. We also write $\mathrm{LT}_1(n) = I + \mathrm{LT}_0(n)$ for the lower-triangular matrices with unit diagonal. OLM and LSM were originally induced by permutation-invariant inner products on $\mathrm{Hol}(n)$ and $\mathrm{Row}_0(n)$.

Following the main-paper convention of CoRNet (Chen et al., 2026, Remark 2.2), GCorP instead equips all four flat prototype spaces with their standard Frobenius inner products.

For a flat head g, a diffeomorphism $\phi_g : \mathrm{Cor}^+(n) \to \mathcal{V}_g$ maps a correlation matrix to its prototype space. It induces the pullback metric $g_C^{\phi_g}(V,W) = \langle \mathrm{d}\phi_{g,C}[V], \mathrm{d}\phi_{g,C}[W] \rangle_{\mathcal{V}_g}$ for $V, W \in T_C \mathrm{Cor}^+(n)$. The subscript $\mathcal{V}_g$ identifies the prototype space in which the inner product is evaluated.

ECM and LECM. For $L = \mathrm{Chol}(C)$ and $D_L = \mathrm{Diag}(\mathrm{diag}(L))$, the diffeomorphism $\Theta(C) = D_L^{-1} L$ maps $\mathrm{Cor}^+(n)$ to $\mathrm{LT}_1(n)$, with $\Theta^{-1} = \mathrm{Cor} \circ \mathrm{Chol}^{-1}$. ECM identifies $\mathrm{LT}_1(n)$ with $\mathcal{V}_{\mathrm{ECM}} = \mathrm{LT}_0(n)$ through $\phi_{\mathrm{ECM}}(C) = \mathrm{lt}(\Theta(C))$. LECM further applies the matrix logarithm, giving $\phi_{\mathrm{LECM}} = \log \circ \Theta : \mathrm{Cor}^+(n) \to \mathrm{LT}_0(n)$ with inverse $\Theta^{-1} \circ \exp$ (Thanwerdas & Pennec, 2022).

OLM and LSM. OLM is the diffeomorphism $\phi_{\mathrm{OLM}} = \mathrm{Log}^\circ = \mathrm{off} \circ \log : \mathrm{Cor}^+(n) \to \mathrm{Hol}(n)$ with inverse Exp°. LSM is the diffeomorphism $\phi_{\mathrm{LSM}} = \mathrm{Log}^\star : \mathrm{Cor}^+(n) \to \mathrm{Row}_0(n)$ with inverse $\mathrm{Exp}^\star = \mathrm{Cor} \circ \exp$, where $\mathrm{Log}^\star(C) = \log(\mathcal{D}^\star(C) C \mathcal{D}^\star(C))$ and $\mathcal{D}^\star(C)$ is the unique positive diagonal scaling that makes the output have zero row sum (Thanwerdas, 2024).

PHCM. For $L = \mathrm{Chol}(C)$, its k-th nontrivial row belongs to the open hemisphere HS^{k-1}, and the rows form the product $\prod_{i=1}^{n-1} \mathrm{HS}^i$ (Thanwerdas & Pennec, 2022). GCorP uses its Poincaré and Lorentz realizations, denoted $\mathrm{GCorP\text{-}PHCM}_p$ and $\mathrm{GCorP\text{-}PHCM}_l$, as detailed in Section 5.1.

The explicit inverse maps, numerical routines, and induced Riemannian operators are given in Section B.

4 Global Correlation Pooling

4.1 From Covariance to Correlation

Let $X = [x_1, \ldots, x_N] \in \mathbb{R}^{d \times N}$ denote the last-layer feature map flattened over its N spatial positions. Vanilla global average pooling returns the first-order representation $\bar{x} = X \mathbf{1}_N / N$.

Global covariance pooling. GCP captures second-order channel statistics by centering the spatial samples and forming a regularized covariance $\Sigma(X) \in \mathrm{SPD}(d)$. This covariance records both how channel responses vary together and each channel's variance. A subsequent matrix function produces the pooled matrix representation used by the task-specific head.

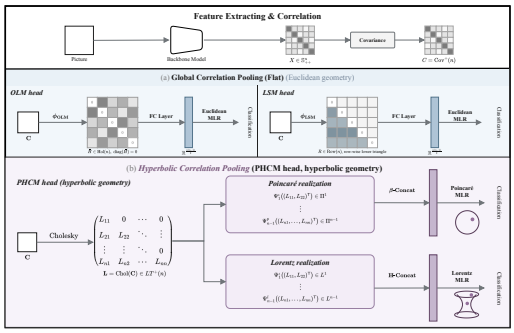


Figure 1: GCorP with flat pooling heads and hyperbolic pooling heads.

Global correlation pooling. GCorP replaces the covariance matrix by $C(X) = \mathrm{Cor}\,(\Sigma(X)) \in \mathrm{Cor}^+(d)$, using the diagonal standardization in Section 3.1. The resulting unit-diagonal matrix retains pairwise channel dependencies on the full-rank correlation manifold. The following subsections instantiate $C(X)$ with six geometry-specific pooling heads. Throughout the paper, GCorP-OLM, GCorP-LSM, GCorP-ECM, GCorP-LECM, $\mathrm{GCorP\text{-}PHCM}_p$, and $\mathrm{GCorP\text{-}PHCM}_l$ denote these heads directly.

4.2 Flat GCorP Heads

GCorP first constructs the correlation matrix described in the previous subsection. A flat pooling head selects $g \in \{\mathrm{ECM}, \mathrm{LECM}, \mathrm{OLM}, \mathrm{LSM}\}$ and applies the diffeomorphism ϕ_g defined in the preliminaries. The resulting element of the prototype space is the pooled representation. It may be retained there for a distance-based objective or vectorized when a downstream module requires a Euclidean vector.

As defined in Section 3.1, ECM and LECM map to $\mathrm{LT}_0(d)$, OLM maps to $\mathrm{Hol}(d)$, and LSM maps to $\mathrm{Row}_0(d)$. We denote the corresponding prototype space by $\mathcal{V}_g$ and set $m = d(d-1)/2$. When a Euclidean vector is needed, we apply a fixed linear bijection $f_{\mathrm{vec},g} : \mathcal{V}_g \to \mathbb{R}^m$. Let vecl stack the strictly lower-triangular entries row by row. For all four flat heads, we use the same extraction

$$f_{\mathrm{vec},g}(U) = \mathrm{vecl}(U) = (U_{ij})_{1 \le j < i \le d}, \qquad g \in \{\mathrm{ECM}, \mathrm{LECM}, \mathrm{OLM}, \mathrm{LSM}\}.$$

For ECM and LECM, these entries are exactly the free entries of $\mathrm{LT}_0(d)$. For OLM, symmetry and the zero diagonal recover the full matrix from its strict lower triangle. For LSM, symmetry first recovers the off-diagonal entries, and the zero-row-sum condition uniquely gives $U_{ii} = -\sum_{j \neq i} U_{ij}$. Thus, the strict lower triangle also uniquely determines every matrix in $\mathrm{Row}_0(d)$. The common map is only a fixed linear identification with $\mathbb{R}^m$ and need not preserve the Frobenius inner product. The prototype-space metric remains the Frobenius metric, while the adjoint $f^*_{\mathrm{vec},g}$ in Proposition 4.1 accounts for the chosen vectorization. Explicit inverses and adjoints are given in Section C.1.

The flat pooling interface maps X to $\Sigma(X)$ by Cov, then to $C(X)$ by Cor, then to $\phi_g(C(X))$ by ϕ_g, and finally to $z_g = f_{\mathrm{vec},g}(\phi_g(C(X)))$. The pooling construction is shared across all four flat geometries; only the diffeomorphism and the resulting geometry-specific representation change.

4.3 Euclidean Realization of Flat CorMLR

For a flat head $g \in \{\mathrm{ECM}, \mathrm{LECM}, \mathrm{OLM}, \mathrm{LSM}\}$, let $\phi_g : \mathrm{Cor}^+(d) \to \mathcal{V}_g$ and $f_{\mathrm{vec},g} : \mathcal{V}_g \to \mathbb{R}^m$ be the diffeomorphism and vectorization specified in Sections 3.1 and 4.2. We equip $\mathcal{V}_g$ with the standard Frobenius inner product. For an input correlation matrix C, write $z_g(C) = \phi_g(C)$ and $x_g(C) = f_{\mathrm{vec},g}(z_g(C))$.

For class k, let $P_k \in \mathrm{Cor}^+(d)$ and $A_k \in T_{P_k}\,\mathrm{Cor}^+(d)$ be the class-specific base point and tangent vector, and set $q_k = \phi_g(P_k)$ and $a_k = \mathrm{d}\phi_{g,P_k}[A_k]$. The flat CorMLR logit is

$$s_{g,k}(C) = \langle z_g(C) - q_k, a_k \rangle_{\mathcal{V}_g}.$$

Let $f^*_{\mathrm{vec},g} : \mathbb{R}^m \to \mathcal{V}_g$ denote the adjoint defined by

$$r^\top f_{\mathrm{vec},g}(u) = \langle f^*_{\mathrm{vec},g}(r), u \rangle_{\mathcal{V}_g}.$$

Proposition 4.1 (Euclidean realization of flat CorMLR). [↓] *For every flat head g, choose the FC parameters so that*

$$f^*_{\mathrm{vec},g}(w_{g,k}) = a_k, \qquad b_{g,k} = -\langle q_k, a_k \rangle_{\mathcal{V}_g}.$$

Then the CorMLR logit is exactly the FC logit,

$$s_{g,k}(C) = w_{g,k}^\top x_g(C) + b_{g,k}.$$

Moreover, optimizing the geometric parameters with the ϕ_g-induced RSGD defined by the pullback of the Euclidean metric under $(P_k, A_k) \mapsto (q_k, a_k)$ produces exactly the same iterates as Euclidean SGD on the flat pairs (q_k, a_k).

Consequently, for every flat head g,

$$\mathrm{CorMLR}_g + \phi_g\text{-induced RSGD} \equiv \phi_g + f_{\mathrm{vec},g} + \mathrm{FC}_{(q,a)} + \text{Euclidean SGD}.$$

Here $\mathrm{FC}_{(q,a)}$ is the reparameterized FC layer with $w_{g,k} = (f^*_{\mathrm{vec},g})^{-1}(a_k)$ and $b_{g,k} = -\langle q_k, a_k \rangle_{\mathcal{V}_g}$. Thus the optimized Euclidean variables are (q_k, a_k), rather than independent affine parameters $(w_{g,k}, b_{g,k})$.

4.4 Fast Matrix Logarithm

Both ϕ_{OLM} and ϕ_{LSM} require a matrix logarithm. Write $B = C$ for OLM and $B = \mathcal{D}^\star C \mathcal{D}^\star$ for LSM, where $\mathcal{D}^\star = \mathcal{D}^\star(C)$. Following Rahman et al. (2026), each method applies its approximation to $\widetilde{B} = B/\operatorname{tr}(B)$ and restores the scale using $\log(B) = \log\left(\widetilde{B}\right) + \log(\operatorname{tr}(B))\, I$. We compare five alternatives in Table 1. Here, $X = \widetilde{B} - I$, $Z = 2\widetilde{B} - I$, K is the truncation order, and $\sigma(B)$ is the spectrum of B. The symbols P_k, T_k, and $L_k^{(\alpha)}$ denote the corresponding polynomial bases, c_k their coefficients, and p_m, q_n the Padé polynomials, with $\alpha > -1$. We use the Legendre expansion in all OLM and LSM experiments; the comparison across families is reported in Section 6.1.3.

Table 1: Fast matrix-log approximations.

Method	Approximation	Range	Method	Approximation	Range
Taylor	$\sum_{k=1}^{K} \frac{(-1)^{k+1}}{k} X^k$	X near 0	Laguerre	$\sum_{k=0}^{K} c_k L_k^{(\alpha)}\left(\widetilde{B}\right)$	$\sigma\left(\widetilde{B}\right) \subset (0,\infty)$
Legendre	$\sum_{k=0}^{K} c_k \mathsf{P}_k(Z)$	$\sigma(Z) \subset (-1,1)$	Padé	$p_m(X) q_n(X)^{-1}$	X near 0
Chebyshev	$\sum_{k=0}^{K} c_k T_k(Z)$	$\sigma(Z) \subset (-1,1)$			

5 PHCM: Hyperbolic Correlation Pooling

5.1 Poincaré and Lorentz PHCM Pooling

We fix curvature $K = -1$. Following the Cholesky-row construction of CorNet (Chen et al., 2026), let $L = \mathrm{Chol}(C)$, write $u_i = L_{i,1:i-1}$ and $\rho_i = L_{ii}$, and define $h_i = \left(u_i^\top, \rho_i\right)^\top$ for $i = 2, \ldots, d$. Since $C = LL^\top$ and $\mathrm{Diag}(C) = I$, each nontrivial row satisfies

$$\|u_i\|^2 + \rho_i^2 = 1, \qquad \rho_i > 0, \tag{1}$$

so h_i lies on the open hemisphere HS^{i-1}. PHCM maps these row factors to a single hyperbolic representation before classification. The Poincaré and Lorentz variants use different aggregation operators:

$$C \xrightarrow{\mathrm{Chol}} \{h_i\}_{i=2}^d \to \begin{cases} \{p_i\}_{i=2}^d \xrightarrow{\beta\text{-Concat}} \phi_{\mathrm{PH},p}(C) \in \mathbb{P}^M, & \mathrm{GCorP\text{-}PHCM}_p, \\ \{\ell_i\}_{i=2}^d \xrightarrow{\text{H-Concat}} \phi_{\mathrm{PH},\ell}(C) \in \mathbb{L}^M, & \mathrm{GCorP\text{-}PHCM}_l, \end{cases} \qquad M = \frac{d(d-1)}{2}. \tag{2}$$

Poincaré aggregation. Stereographic projection maps each Cholesky row to

$$p_i(C) = \frac{u_i}{1+\rho_i} \in \mathbb{P}^{i-1}. \tag{3}$$

Let $\beta_m = B\left(m/2, 1/2\right)$. The standard β-concatenation of Shimizu et al. (2021) gives

$$\phi_{\mathrm{PH},p}(C) = \mathrm{Exp}_0^{\mathbb{P}^M}\left(\begin{bmatrix} \frac{\beta_M}{\beta_1} \mathrm{Log}_0^{\mathbb{P}^1}\left(p_2(C)\right) \\ \vdots \\ \frac{\beta_M}{\beta_{d-1}} \mathrm{Log}_0^{\mathbb{P}^{d-1}}\left(p_d(C)\right) \end{bmatrix}\right). \tag{4}$$

Lorentz aggregation. The corresponding row factor in the Lorentz model is

$$\ell_i(C) = \left(\frac{1}{\rho_i}, \frac{u_i}{\rho_i}\right) \in \mathbb{L}^{i-1}. \tag{5}$$

Writing

$$s(C) = \begin{bmatrix} u_2/\rho_2 \\ \vdots \\ u_d/\rho_d \end{bmatrix} \in \mathbb{R}^M, \qquad \phi_{\mathrm{PH},\ell}(C) = \begin{bmatrix} \sqrt{1 + \|s(C)\|^2} \\ s(C) \end{bmatrix} \in \mathbb{L}^M, \tag{6}$$

directly concatenates the spatial components and restores the Lorentz time component. Both β-Concat and H-Concat are numerical aggregation operators that convert the Cholesky-row factors into one classifier input.

Implementation. $\mathrm{GCorP\text{-}PHCM}_p$ feeds $\phi_{\mathrm{PH},p}(C)$ to the Poincaré MLR of HNN++ (Shimizu et al., 2021), whereas $\mathrm{GCorP\text{-}PHCM}_l$ feeds $\phi_{\mathrm{PH},\ell}(C)$ to Lorentz MLR (Bdeir et al., 2024). Their closed forms and numerical details are given in Section D.3.

Table 2: ImageNet-1K results. Bold and underline mark the best and second-best values for each metric and backbone, respectively.

Method	ViT-B/16			Swin-T			ResNet-50			ResNet-101		
	Top-1 ↑	Top-5 ↑	s/batch ↓	Top-1 ↑	Top-5 ↑	s/batch ↓	Top-1 ↑	Top-5 ↑	s/batch ↓	Top-1 ↑	Top-5 ↑	s/batch ↓
iSQRT-COV (Li et al., 2018)	79.58	93.92	0.417	78.18	93.46	0.231	76.28	92.19	0.132	79.16	94.26	0.188
MPA (Song et al., 2022a)	79.22	93.95	0.405	78.84	94.29	0.219	78.61	94.06	0.119	**80.38**	95.11	0.172
SVD-Padé (Song et al., 2021)	79.63	94.20	3.950	76.43	93.11	2.532	76.17	92.08	0.535	77.54	93.03	0.715
Halley-SVD (Gu et al., 2025)	76.01	92.11	0.472	75.12	91.90	0.259	74.92	90.81	0.160	77.39	93.57	0.213
ISICE (Rahman et al., 2023)	**81.28**	95.59	0.505	**80.25**	**95.25**	0.303	**79.02**	94.22	0.204	80.14	95.21	0.248
GCorP-OLM	81.26	**95.62**	0.396	80.24	95.12	0.210	78.99	**94.35**	0.114	80.31	**95.26**	0.166
GCorP-LSM	80.87	95.23	0.417	80.00	95.08	0.216	78.51	94.10	0.128	79.89	94.88	0.181
GCorP-ECM	70.25	89.49	**0.378**	77.95	93.62	**0.205**	76.55	93.14	**0.106**	78.62	94.28	**0.158**
GCorP-LECM	67.26	87.72	0.397	77.72	93.66	0.213	76.35	93.17	0.114	77.95	93.85	0.168
GCorP-PHCM$_p$	79.71	93.59	0.395	76.35	93.01	0.213	77.02	92.93	0.114	78.17	93.01	0.167
GCorP-PHCM$_l$	80.33	94.94	0.390	78.56	94.34	0.212	75.92	92.51	0.113	77.50	93.31	0.166

6 Experiments

6.1 Classification Analysis

6.1.1 Visual Classification

We evaluate GCorP on ImageNet-1K with ViT-B/16, Swin-T, ResNet-50, and ResNet-101 backbones against covariance- and partial-correlation-based pooling methods. Full training, timing, and implementation details are given in Section E.1.

Accuracy and efficiency. GCorP-OLM records the highest GCorP Top-1 on every reported backbone. It is within 0.03 Top-1 points of ISICE on ViT-B/16, Swin-T, and ResNet-50 and 0.17 points above it on ResNet-101, while its measured per-batch time is approximately 1.3–1.8× lower. The Legendre implementation evaluates the logarithmic map with matrix products, which accounts for this cost profile relative to the decomposition- and solve-based baselines.

Geometry pattern. Across all four ImageNet backbones the non-Cholesky OLM and LSM heads reach higher Top-1 and Top-5 accuracy than the Cholesky-based ECM, LECM, and PHCM variants. The gap is backbone-dependent: GCorP-OLM exceeds GCorP-ECM by 11.0 points on ViT-B/16 and by 2.3, 2.4, and 1.7 points on Swin-T, ResNet-50, and ResNet-101.

Covariance versus correlation. We hold the MPA pipeline, ViT-B/16 backbone, and experimental protocol fixed and only change the statistic entering the matrix-function step. Replacing covariance by correlation raises Top-1 from 79.22% to 80.15% and Top-5 from 93.95% to 94.79%, directly supporting correlation as a second-order representation. The complete comparison is reported in Table 8.

6.1.2 EEG Foundation-Model Classification

Pipeline. Given an EEG segment, the pretrained LaBraM-Base model produces a global token y_0 and non-global spatiotemporal tokens (Jiang et al., 2024). Following RHOP (Hu et al., 2026), the latter are summarized by a Fréchet mean Y_m and tangent-space covariance Y_c, which are embedded into the high-order SPD representation $G = \mathrm{embed}(Y_m, Y_c)$. RHOP and GCorP share this construction through G and differ only in the downstream head:

$$z_g = \begin{cases} \mathrm{vec}\left(\mathrm{iSICE}(G)\right), & g = \mathrm{RHOP}, \\ f_{\mathrm{vec},g}\left(\phi_g\left(\mathrm{Cor}(G)\right)\right), & g \in \{\mathrm{ECM}, \mathrm{LECM}, \mathrm{OLM}, \mathrm{LSM}\}, \\ \phi_g\left(\mathrm{Cor}(G)\right), & g \in \{\mathrm{PHCM}_p, \mathrm{PHCM}_\ell\}. \end{cases} \tag{7}$$

Thus, RHOP applies iSICE directly to G, whereas GCorP first standardizes G into the correlation matrix $C = \mathrm{Cor}(G)$ and then applies a geometry-specific map ϕ_g. For the four flat heads, z_g is concatenated with y_0 and passed to a Euclidean linear classifier; for the GCorP-PHCM heads, z_g and a projected y_0 are fused in the origin tangent space before the corresponding hyperbolic MLR. The token statistics and exact fusion equations are given in Section E.3.

Table 3: EEG test performance (%, mean ± standard deviation over three seeds) and runtime (seconds per iteration). Bold and underline mark the best and second-best values per metric, respectively.

Method	TUAB				TUEV			
	BAcc.	PR-AUC	AUROC	s/iter.	BAcc.	Kappa	W. F1	s/iter.
iSQRT-COV (Li et al., 2018)	82.053 ± 0.157	90.622 ± 0.467	90.370 ± 0.709	0.1777	**63.99 ± 0.59**	60.13 ± 1.50	80.55 ± 0.72	0.1130
SVD-Padé (Song et al., 2021)	82.061 ± 0.150	90.811 ± 0.241	90.530 ± 0.101	0.2694	51.13 ± 1.19	47.23 ± 2.03	71.21 ± 3.69	0.1997
Halley-SVD (Gu et al., 2025)	81.532 ± 0.444	90.055 ± 0.589	89.652 ± 0.481	0.2930	59.45 ± 1.26	61.49 ± 1.19	81.14 ± 0.77	0.2440
ISICE (Rahman et al., 2023)	**82.150 ± 0.071**	90.784 ± 0.279	90.479 ± 0.393	0.2039	60.55 ± 3.03	57.91 ± 3.06	79.72 ± 1.32	0.1455
RHOP (Hu et al., 2026)	81.902 ± 0.511	90.779 ± 0.161	90.560 ± 0.100	0.2019	59.13 ± 1.49	56.39 ± 5.93	79.19 ± 3.42	0.1150
GCorP-LSM	81.610 ± 0.621	90.101 ± 0.310	90.113 ± 0.280	0.1547	62.34 ± 0.58	58.41 ± 2.52	79.60 ± 1.16	0.0843
GCorP-OLM	81.957 ± 0.285	**90.818 ± 0.330**	**90.601 ± 0.437**	**0.1415**	60.51 ± 4.78	59.64 ± 2.10	80.16 ± 0.83	**0.0732**
GCorP-PHCM$_p$	82.080 ± 0.361	89.726 ± 0.510	90.203 ± 0.205	0.1527	63.82 ± 0.18	**61.83 ± 1.37**	**81.53 ± 1.67**	0.0772
GCorP-PHCM$_l$	81.550 ± 0.490	89.940 ± 0.239	89.733 ± 0.284	0.1517	61.25 ± 2.96	60.34 ± 1.70	80.78 ± 1.84	0.0768
GCorP-ECM	81.903 ± 0.300	90.470 ± 0.469	90.361 ± 0.481	0.1450	60.45 ± 5.82	59.50 ± 1.99	80.59 ± 1.37	0.0749
GCorP-LECM	81.737 ± 0.260	90.251 ± 0.302	90.167 ± 0.342	0.1453	59.61 ± 2.40	57.33 ± 2.25	79.46 ± 1.10	0.0764

Configurations are selected using validation data only, with validation weighted F1 on TUEV and validation balanced accuracy on TUAB. The full experimental setup, checkpoint-selection protocol, and validation sweeps are given in Section E.3.

Table 4: pMF-B/16 post-training results under Non-SIM and SIM objectives. Bold and underline mark the best and second-best values per metric. *MPA** undergoes numerical collapse.

Objective	Method	Non-SIM			SIM		
		FID ↓	FDr$_6$ ↓	s/step ↓	FID ↓	FDr$_6$ ↓	s/step ↓
pMF-B/16	Base (Lu et al., 2026)	3.31	13.70	–	3.31	13.70	–
	+ *FD-loss* (Yang et al., 2026)	**0.81**	10.81	0.595	0.94	**4.20**	0.764
Approx. cov.	iSQRT-COV (Li et al., 2018)	10.20	19.46	0.620	9.38	10.90	0.805
	MPA (Song et al., 2022a)*						
	SVD-Padé (Song et al., 2021)	**0.81**	10.95	0.620	2.25	9.23	0.811
	Halley-SVD (Gu et al., 2025)	2.41	14.05	0.716	11.27	27.25	0.971
Correlation	GCorP-OLM	0.84	10.80	0.541	0.95	4.41	0.695
	GCorP-LSM	0.84	**10.63**	0.608	0.95	4.43	0.709
	GCorP-PHCM$_p$	0.87	10.72	0.564	0.98	4.32	0.708
	GCorP-PHCM$_l$	0.87	10.94	0.548	**0.92**	5.50	0.741
	GCorP-ECM	0.87	11.04	**0.522**	0.99	5.48	**0.641**
	GCorP-LECM	1.66	13.25	0.541	1.88	8.03	0.681

Results. This comparison preserves the LaBraM-Base backbone and the upstream token statistics within each dataset, with all configurations selected on validation data before test evaluation. On TUAB, the reported heads achieve balanced accuracies from 81.55% to 82.15%. On TUEV, GCorP-PHCM$_p$ records 61.83% Cohen's kappa, 81.53% weighted F1, and 63.82% balanced accuracy, while GCorP-OLM records the lowest iteration time at 0.0732 seconds versus 0.1150 for RHOP. GCorP can therefore be inserted without redesigning the upstream foundation model, with the preferred geometry depending on the dataset.

6.1.3 Ablation study

Logarithm approximation. We compare Legendre, Taylor, Chebyshev, Laguerre, and Padé approximations for the matrix logarithm. Legendre is selected for OLM and LSM because it gives the strongest classification result among the tested approximations without requiring the linear solve used by Padé. The complete sweep is reported in Table 9.

Expansion order. We sweep the approximation order on ImageNet classification and FD-loss post-training. The two settings use different default orders because the classification and distribution-matching metrics follow different order trends. The complete ImageNet, OLM, and LSM sweeps are given in Table 10.

Newton iterations. We vary the number of damped Newton iterations used by LSM to obtain $\mathcal{D}^\star(C)$. The sweep supports using a single iteration, which is the default in all LSM experiments. The complete results are reported in Table 11.

PHCM input scaling. We sweep the input scale for both Poincaré and Lorentz PHCM. The selected scales are used in the corresponding ImageNet experiments, with input scaling serving as the PHCM optimization control. The complete records are reported in Table 12.

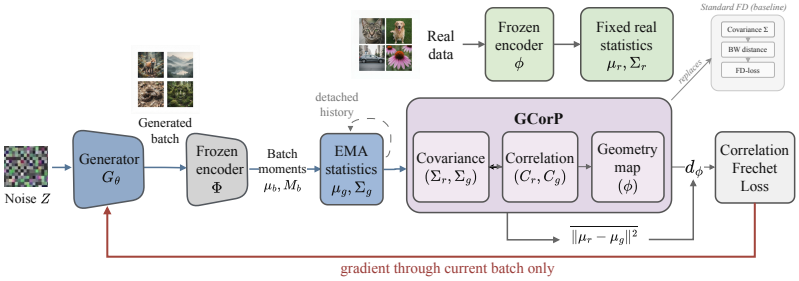


Figure 2: FD-loss post-training with GCorP and detached EMA statistics.

6.2 Generative Model Post-Training

We apply GCorP to FD-loss (Yang et al., 2026), which post-trains a generator by matching feature statistics of generated and real images. A queue or exponential moving average (EMA) accumulates large-population statistics, while gradients are propagated only through the current generated batch.

The original objective matches feature means and covariances through $\mathcal{L}_{\mathrm{FD},k} = \|\mu_{r,k} - \mu_{g,k}\|_2^2 + d_{\mathrm{BW}}^2(\Sigma_{r,k}, \Sigma_{g,k})$, where d_{BW} is the Bures–Wasserstein covariance distance (Han et al., 2021). GCorP keeps the mean term and population-estimation pipeline, standardizes each covariance as $C_{t,k} = \mathrm{Diag}(\Sigma_{t,k})^{-1/2}\,\Sigma_{t,k}\,\mathrm{Diag}(\Sigma_{t,k})^{-1/2}$ for $t \in \{r, g\}$, and replaces $d_{\mathrm{BW}}^2(\Sigma_{r,k}, \Sigma_{g,k})$ by $d_\phi^2(C_{r,k}, C_{g,k})$, where $d_\phi(C_r, C_g) = d_{\mathcal{Z}_\phi}(\phi(C_r), \phi(C_g))$. The resulting per-encoder objective is

$$\mathcal{L}_{\phi,k} = \|\mu_{r,k} - \mu_{g,k}\|_2^2 + d_\phi^2(C_{r,k}, C_{g,k})\,. \tag{8}$$

For ECM, LECM, OLM, and LSM, d_ϕ is the Euclidean distance between the corresponding elements in flat prototype spaces; for PHCM, it is the Poincaré or Lorentz geodesic distance. Thus, GCorP changes the geometry used for second-order matching while preserving the FD-loss population estimator and mean-matching term. The objective compares standardized feature dependencies rather than raw covariances; the complete training pipeline is shown in Fig. 2 and detailed in Section E.4.

We post-train pMF-B/16 (Lu et al., 2026) with Inception-only (Non-SIM) and SIM objectives, comparing exact covariance FD, approximate covariance variants, and GCorP. The complete training setup, metric definitions, and implementation details are given in Section E.4.

Results. Compared with exact covariance FD, the leading GCorP objectives preserve comparable distribution- matching quality under both Non-SIM and SIM while reducing optimization time. Under Non-SIM, OLM, LSM, and PHCMp stay close to exact FD, with OLM and PHCMp requiring less time per step. Under SIM, PHCMp remains close to exact FD in FDr_6, while OLM reduces the per-step cost. GCorP also avoids the large FDr_6 degradation of iSQRT-COV and Halley-SVD and the numerical failure of MPA, improving the quality–efficiency trade-off through geometry-specific distances on correlation representations.

7 Conclusion

We introduced GCorP, a second-order pooling representation that brings full-rank correlation geometry into end-to-end learning through flat or hyperbolic pooling heads under one interface. For the flat geometries, GCorP also realizes non-trivialized CorMLR exactly with Euclidean computation: FC logits give the intrinsic classifier scores, and Euclidean SGD on the induced flat parameters reproduces the corresponding Riemannian SGD trajectory. On image recognition, EEG classification, and generative-model post-training, GCorP achieves a favorable accuracy–efficiency trade-off.

REFERENCES

Ilya Archakov and Peter Reinhard Hansen. A new parametrization of correlation matrices. *Econometrica*, 89(4):1699–1715, 2021.

Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine*, 56(2):411–421, 2006.

Alexandre Barachant, Stéphane Bonnet, Marco Congedo, and Christian Jutten. Classification of covariance matrices using a Riemannian-based kernel for BCI applications. *Neurocomputing*, 112: 172–178, 2013.

Ahmad Bdeir, Kristian Schwethelm, and Niels Landwehr. Fully hyperbolic convolutional neural networks for computer vision. In *ICLR*, 2024.

Rajendra Bhatia. *Positive Definite Matrices*. Princeton University Press, 2007.

Daniel Brooks, Olivier Schwander, Frédéric Barbaresco, Jean-Yves Schneider, and Matthieu Cord. Riemannian batch normalization for SPD neural networks. In *NeurIPS*, 2019.

Ziheng Chen, Yue Song, Gaowen Liu, Ramana Rao Kompella, Xiao-Jun Wu, and Nicu Sebe. Riemannian multinomial logistics regression for SPD neural networks. In *CVPR*, 2024a.

Ziheng Chen, Yue Song, Rui Wang, Xiao-Jun Wu, and Nicu Sebe. RMLR: Extending multinomial logistic regression into general geometries. In *NeurIPS*, 2024b.

Ziheng Chen, Yue Song, Xiao-Jun Wu, Gaowen Liu, and Nicu Sebe. Understanding matrix function normalizations in covariance pooling through the lens of riemannian geometry. In *ICLR*, 2025.

Ziheng Chen, Xiao-Jun Wu, Bernhard Schölkopf, and Nicu Sebe. Riemannian networks over full-rank correlation matrices. In *ICML*, 2026.

Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. In *NeurIPS*, 2018.

Zilin Gao, Jiangtao Xie, Qilong Wang, and Peihua Li. Global second-order pooling convolutional networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3019–3028, 2019. doi: 10.1109/CVPR.2019.00314.

Zilin Gao, Qilong Wang, Bingbing Zhang, Qinghua Hu, and Peihua Li. Temporal-attentive covariance pooling networks for video recognition. In *NeurIPS*, 2021.

Jiawei Gu, Ziyue Qiao, Xinming Li, and Zechao Li. Revitalizing SVD for global covariance pooling: Halley's method to overcome over-flattening. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=fqpbXJ2QtC.

Andi Han, Bamdev Mishra, Pratik Jawanpuria, and Junbin Gao. Learning with symmetric positive definite matrices via generalized Bures-Wasserstein geometry. In *NeurIPS*, 2021.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, 2017.

Chen Hu, Ziheng Chen, Rui Wang, Yefeng Zheng, and Nicu Sebe. Riemannian high-order pooling for brain foundation models. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=66h1sCMm7F.

Zhiwu Huang and Luc Van Gool. A Riemannian network for SPD matrix learning. In *AAAI*, 2017.

Catalin Ionescu, Orestis Vantzos, and Cristian Sminchisescu. Matrix backpropagation for deep networks with structured layers. In *ICCV*, 2015.

Wei-Bang Jiang, Li-Ming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous EEG data in BCI. In *ICLR*, 2024.

Peihua Li, Jiangtao Xie, Qilong Wang, and Wangmeng Zuo. Is second-order information helpful for large-scale visual recognition? In *ICCV*, 2017.

Peihua Li, Jiangtao Xie, Qilong Wang, and Zilin Gao. Towards faster training of global covariance pooling networks by iterative matrix square root normalization. In *CVPR*, 2018.

Tsung-Yu Lin and Subhransu Maji. Improved bilinear pooling with cnns. In Gabriel Brostow Tae-Kyun Kim, Stefanos Zafeiriou and Krystian Mikolajczyk (eds.), *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 117.1–117.12. BMVA Press, September 2017. ISBN 1-901725-60-X. doi: 10.5244/C.31.117. URL https://dx.doi.org/10.5244/C.31.117.

Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1449–1457, 2015. doi: 10.1109/ICCV.2015.170.

Yiyang Lu, Susie Lu, Qiao Sun, Hanhong Zhao, Zhicheng Jiang, Xianbang Wang, Tianhong Li, Zhengyang Geng, and Kaiming He. One-step latent-free image generation with pixel mean flows, 2026.

Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in Neural Information Processing Systems*, 30, 2017.

Maximillian Nickel and Douwe Kiela. Learning continuous hierarchies in the lorentz model of hyperbolic geometry. In *International Conference on Machine Learning*, pp. 3779–3788. PMLR, 2018.

Xavier Pennec, Pierre Fillard, and Nicholas Ayache. A Riemannian framework for tensor computing. *International Journal of Computer Vision*, 66(1):41–66, 2006.

Md Rifat Ur Rahman, Md Raihan Khan, Md Sakib Hossain Shovon, Pietro Liò, and Mohammad Ali Moni. Orthogonal polynomial approximation for matrix log normalization in global covariance pooling, 2026. URL https://arxiv.org/abs/2608.19021.

Saimunur Rahman, Piotr Koniusz, Lei Wang, Luping Zhou, Peyman Moghadam, and Changming Sun. Learning partial correlation based deep visual representation for image classification. In *CVPR*, 2023.

Ryohei Shimizu, YUSUKE Mukuta, and Tatsuya Harada. Hyperbolic neural networks++. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=Ec85b0tUwbA.

Ondrej Skopek, Octavian-Eugen Ganea, and Gary Bérard. Mixed-curvature variational autoencoders. In *ICLR*, 2020.

Yue Song, Nicu Sebe, and Wei Wang. Why approximate matrix square root outperforms accurate SVD in global covariance pooling? In *ICCV*, 2021.

Yue Song, Nicu Sebe, and Wei Wang. Fast differentiable matrix square root. In *ICLR*, 2022a.

Yue Song, Nicu Sebe, and Wei Wang. Fast differentiable matrix square root and inverse square root. *IEEE TPAMI*, 2022b.

Yue Song, Nicu Sebe, and Wei Wang. On the eigenvalues of global covariance pooling for fine-grained visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (3):3554–3566, 2023. doi: 10.1109/TPAMI.2022.3178802.

Yann Thanwerdas. Permutation-invariant log-Euclidean geometries on full-rank correlation matrices. *SIAM Journal on Matrix Analysis and Applications*, 45(2):930–953, 2024. doi: 10.1137/22M1538144.

Yann Thanwerdas and Xavier Pennec. Theoretically and computationally convenient geometries on full-rank correlation matrices. *SIAM Journal on Matrix Analysis and Applications*, 2022.

Qilong Wang, Jiangtao Xie, Wangmeng Zuo, Lei Zhang, and Peihua Li. Deep cnns meet global covariance pooling: Better representation and generalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(8):2582–2597, 2021. doi: 10.1109/TPAMI.2020.2974833.

Qilong Wang, Zhaolin Zhang, Mingze Gao, Jiangtao Xie, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Towards a deeper understanding of global covariance pooling in deep learning: An optimization perspective. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2023.

Jiangtao Xie, Ruiren Zeng, Qilong Wang, Ziqi Zhou, and Peihua Li. Sot: Delving deeper into classification head for transformer, 2021. URL https://arxiv.org/abs/2104.10935.

Jiawei Yang, Zhengyang Geng, Xuan Ju, Yonglong Tian, and Yue Wang. Representation Fréchet loss for visual generation, 2026. URL https://arxiv.org/abs/2604.28190.

Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 1839–1848, 2017. doi: 10.1109/ICCV.2017.202.

Zhou Yu, Jun Yu, Chenchao Xiang, Jianping Fan, and Dacheng Tao. Beyond bilinear: Generalized multimodal factorized high-order pooling for visual question answering. *IEEE Transactions on Neural Networks and Learning Systems*, 29(12):5947–5959, December 2018. ISSN 2162-2388. doi: 10.1109/tnnls.2018.2817340. URL http://dx.doi.org/10.1109/TNNLS.2018.2817340.

APPENDIX CONTENTS

A Use of Large Language Models **13**

B Full-Rank Correlation Geometry Details **13**

B.1 Flat Pullback Geometries 13

B.1.1 ECM and LECM 13

B.1.2 OLM and LSM 13

B.2 PHCM Product Geometry 14

C Riemannian MLR Background **14**

C.1 Vectorizations for the Four Flat Heads 14

C.2 Proof of the Flat CorMLR Realization 15

D Implementation Details for PHCM **16**

D.1 Poincaré Aggregation 16

D.2 Lorentz Aggregation 17

D.3 Closed-Form Hyperbolic Logits 17

D.4 Numerical Stabilization 17

E Experimental Setup Details **18**

E.1 ImageNet-1K Classification 18

E.2 Ablation Details 18

E.3 EEG Evaluation 20

E.4 FD Fine-Tuning 22

F Extended Related Work **24**

F.1 Global Covariance Pooling 24

F.2 Geometry of Full-Rank Correlation Matrices 24

F.3 Hyperbolic Neural Networks 24

F.4 High-Order Global Pooling 25

A USE OF LARGE LANGUAGE MODELS

Large Language Models (LLMs) were used primarily for language polishing and minor text editing. In limited cases, they also assisted in translating certain mathematical formulations into PyTorch code. All generated outputs were carefully reviewed and, where necessary, corrected by the authors. The authors take full responsibility for the final content of this paper.

B FULL-RANK CORRELATION GEOMETRY DETAILS

B.1 FLAT PULLBACK GEOMETRIES

We give the inverse maps and induced Riemannian operators omitted from Section 3.1. Let $\phi : \mathrm{Cor}^+(n) \to \mathcal{V}$ be any of the four flat diffeomorphisms, where $\mathcal{V}$ has the standard Frobenius inner product. For $C, C' \in \mathrm{Cor}^+(n)$ and $V, W \in T_C \mathrm{Cor}^+(n)$, the pullback metric is

$$g_C^\phi(V, W) = \langle \mathrm{d}\phi_C[V], \mathrm{d}\phi_C[W] \rangle_{\mathcal{V}}. \tag{9}$$

The logarithm, exponential, geodesic, and distance are respectively

$$\begin{aligned} \mathrm{Log}_C^{g^\phi}(C') &= (\mathrm{d}\phi_C)^{-1}[\phi(C') - \phi(C)], \\ \mathrm{Exp}_C^{g^\phi}(V) &= \phi^{-1}(\phi(C) + \mathrm{d}\phi_C[V]), \\ \gamma_{C,C'}(t) &= \phi^{-1}((1-t)\phi(C) + t\phi(C')), \\ \mathrm{d}_{g^\phi}(C, C') &= \|\phi(C) - \phi(C')\|_{\mathcal{V}}. \end{aligned} \tag{10}$$

These identities follow because ϕ is an isometry from the pullback geometry to its Euclidean prototype space.

B.1.1 ECM AND LECM

For $C \in \mathrm{Cor}^+(n)$, write $L = \mathrm{Chol}(C)$, $D_L = \mathrm{Diag}(\mathrm{diag}(L))$, and $\Theta(C) = D_L^{-1} L \in \mathrm{LT}_1(n)$. For $T \in \mathrm{LT}_1(n)$, the inverse is

$$\Theta^{-1}(T) = \mathrm{Cor}(TT^\top). \tag{11}$$

Consequently, for $U \in \mathrm{LT}_0(n)$,

$$\phi_{\mathrm{ECM}}^{-1}(U) = \mathrm{Cor}((I+U)(I+U)^\top), \qquad \phi_{\mathrm{LECM}}^{-1}(U) = \mathrm{Cor}(\exp(U)\exp(U)^\top). \tag{12}$$

Since $U^n = 0$, the required matrix functions are finite polynomials:

$$\log(I+U) = \sum_{k=1}^{n-1} \frac{(-1)^{k-1}}{k} U^k, \qquad \exp(U) = \sum_{k=0}^{n-1} \frac{1}{k!} U^k. \tag{13}$$

Thus, ECM and LECM do not require an eigendecomposition for these triangular matrix functions.

B.1.2 OLM AND LSM

For $H \in \mathrm{Hol}(n)$, the OLM inverse is

$$\mathrm{Exp}^\circ(H) = \exp(\mathcal{D}(H) + H), \tag{14}$$

where $\mathcal{D}(H)$ is the unique diagonal matrix for which the result has unit diagonal. Starting from $D_0 = 0$, it can be computed by

$$D_{r+1} = D_r - \mathrm{Diag}\left(\log\left(\mathrm{diag}\left(\exp(D_r + H)\right)\right)\right), \tag{15}$$

which converges exponentially (Archakov & Hansen, 2021).

For $R \in \mathrm{Row}_0(n)$, the LSM inverse is $\mathrm{Exp}^\star(R) = \mathrm{Cor}(\exp(R))$. The forward map uses $\mathcal{D}^\star(C) = \mathrm{Diag}(x)$, where $x \in \mathbb{R}_+^n$ is the unique solution of

$$F_C(x) = Cx - x^{-1} = 0. \tag{16}$$

Here x^{-1} is taken componentwise. Since $J_C(x) = C + \mathrm{Diag}(x^{-2})$, a damped Newton step is

$$x_{r+1} = x_r - \alpha_r J_C(x_r)^{-1} F_C(x_r), \qquad \alpha_r \in (0, 1], \tag{17}$$

with α_r chosen to keep the iterate positive (Thanwerdas, 2024).

B.2 PHCM Product Geometry

For $C \in \mathrm{Cor}^+(n)$, let $L = \mathrm{Chol}(C)$. Because C has unit diagonal, the k-th row prefix $h_k = (L_{k1}, \ldots, L_{kk})$ satisfies $\|h_k\| = 1$ and $L_{kk} > 0$. Hence

$$h_k \in \mathrm{HS}^{k-1} = \left\{ u \in \mathbb{R}^k \mid \|u\| = 1 \text{ and } u_k > 0 \right\}. \tag{18}$$

The first row is fixed, so the Cholesky map identifies the correlation manifold with the product $\prod_{i=1}^{n-1} \mathrm{HS}^i$. PHCM is the pullback of the product hyperbolic metric through this map. GCorP uses the canonical product with unit component weights. Its Poincaré and Lorentz realizations and their aggregation operators are given in Sections D.1, D.2 and 5.1.

C Riemannian MLR Background

C.1 Vectorizations for the Four Flat Heads

Let $m = d(d-1)/2$, and let vecl stack the strictly lower-triangular entries row by row. For every flat head, we use the common linear map $f_{\mathrm{vec},g}(U) = \mathrm{vecl}(U)$. This map identifies each prototype space with $\mathbb{R}^m$, but it is not required to preserve the Frobenius inner product.

Table 5: Explicit vectorizations for the four flat GCorP heads.

Head	$\boldsymbol{\mathcal{V}_g}$	$\boldsymbol{f_{\mathrm{vec},g}(U) \in \mathbb{R}^m}$
ECM	$\mathrm{LT}_0(d)$	$\mathrm{vecl}(U)$
LECM	$\mathrm{LT}_0(d)$	$\mathrm{vecl}(U)$
OLM	$\mathrm{Hol}(d)$	$\mathrm{vecl}(U)$
LSM	$\mathrm{Row}_0(d)$	$\mathrm{vecl}(U)$

The common extraction does not make the four pooled representations identical: the matrix $U = \phi_g(C)$ is still produced by the geometry-specific map ϕ_g. We next give the inverse of the extraction for each prototype space. Write $x = (x_{ij})_{1 \le j < i \le d} \in \mathbb{R}^m$ in the same row-wise order as vecl.

For ECM and LECM, define $U_{ij} = x_{ij}$ when $i > j$ and $U_{ij} = 0$ when $i \le j$. This uniquely recovers an element of $\mathrm{LT}_0(d)$. For OLM, set $U_{ij} = U_{ji} = x_{ij}$ for $i > j$ and $U_{ii} = 0$. Symmetry and the zero diagonal then uniquely recover an element of $\mathrm{Hol}(d)$.

For LSM, first set $U_{ij} = U_{ji} = x_{ij}$ for every $i > j$, and then define

$$U_{ii} = -\sum_{j \ne i} U_{ij}, \qquad 1 \le i \le d. \tag{19}$$

The resulting matrix is symmetric, and each row satisfies $\sum_{j=1}^{d} U_{ij} = 0$, so $U \in \mathrm{Row}_0(d)$. Conversely, for any $U \in \mathrm{Row}_0(d)$, the off-diagonal entries determine every diagonal entry by Equation (19). The recovery is therefore unique. Moreover, $\dim(\mathrm{Row}_0(d)) = d(d+1)/2 - d = m$, which equals the number of strict lower-triangular entries. Thus, vecl is a linear bijection on all four prototype spaces.

For completeness, the adjoint $f^*_{\mathrm{vec},g} : \mathbb{R}^m \to \mathcal{V}_g$ is the unique linear map satisfying

$$r^\top f_{\mathrm{vec},g}(u) = \left\langle f^*_{\mathrm{vec},g}(r), u \right\rangle_{\mathcal{V}_g} \quad \text{for all } r \in \mathbb{R}^m,\ u \in \mathcal{V}_g. \tag{20}$$

Since $f_{\mathrm{vec},g}$ is bijective between spaces of the same finite dimension, $f^*_{\mathrm{vec},g}$ is also bijective. Thus the equation $f^*_{\mathrm{vec},g}(w) = a$ has the unique solution $w = \left(f^*_{\mathrm{vec},g}\right)^{-1}(a)$. Let w_{ij} denote the component of w paired with the entry U_{ij} for $i > j$. The explicit weight relation is

$$w_{ij} = \begin{cases} a_{ij}, & g \in \{\mathrm{ECM}, \mathrm{LECM}\}, \\ 2a_{ij}, & g = \mathrm{OLM}, \\ 2a_{ij} - a_{ii} - a_{jj}, & g = \mathrm{LSM}, \end{cases} \qquad 1 \le j < i \le d. \tag{21}$$

Indeed, the Frobenius pairing for ECM and LECM contains each retained entry once. For $a, U \in \mathrm{Hol}(d)$, symmetry gives $\langle a, U\rangle_F = 2\sum_{i>j} a_{ij}U_{ij}$. For $a, U \in \mathrm{Row}_0(d)$, substituting $U_{ii} = -\sum_{j\neq i} U_{ij}$ gives

$$\begin{aligned}\langle a, U\rangle_F &= \sum_{i=1}^{d} a_{ii}U_{ii} + 2\sum_{i>j} a_{ij}U_{ij} \\ &= \sum_{i>j} (2a_{ij} - a_{ii} - a_{jj})\, U_{ij}.\end{aligned} \tag{22}$$

These identities prove Equation (21) and show why the raw lower-triangular vectorization need not be a Frobenius isometry. The general adjoint relation, rather than an isometry assumption, is the weight relation used in Proposition 4.1.

C.2 Proof of the Flat CorMLR Realization

For $x \in \mathbb{R}^m$, Euclidean multinomial logistic regression uses affine class logits $\ell_k(x) = w_k^\top x + b_k$. If $w_k \neq 0$, choose a reference point p_k satisfying $\langle p_k, w_k\rangle = -b_k$. Then

$$\ell_k(x) = \langle x - p_k, w_k\rangle\,. \tag{23}$$

The Riemannian MLR construction replaces the Euclidean displacement by a logarithm map at a class-specific base point and replaces the Euclidean inner product by the metric at that point (Ganea et al., 2018; Chen et al., 2024a;b; 2026). For $P \in \mathcal{M}$ and a nonzero $A \in T_P\mathcal{M}$, define the Riemannian hyperplane

$$H_{A,P} = \{C \in \mathcal{M} \mid \langle \mathrm{Log}_P(C), A\rangle_P = 0\}\,. \tag{24}$$

The associated signed-margin logit is

$$v(C; P, A) = \mathrm{sgn}\left(\langle \mathrm{Log}_P(C), A\rangle_P\right) \|A\|_P\, \mathrm{d}\left(C, H_{A,P}\right). \tag{25}$$

We use this signed-distance form only to justify the prototype-space expression in the main text. The CorMLR parameters remain the base point and tangent vector (P_k, A_k); no shared base point is imposed on the class parameters.

Proof of Proposition 4.1. Fix $g \in \{\mathrm{ECM}, \mathrm{LECM}, \mathrm{OLM}, \mathrm{LSM}\}$ and abbreviate $\phi = \phi_g$, $\mathcal{V} = \mathcal{V}_g$, and $f = f_{\mathrm{vec},g}$. Let $\mathcal{M} = \mathrm{Cor}^+(d)$ and write $g^\phi = \phi^*\langle\cdot,\cdot\rangle_{\mathcal{V}}$. Since ϕ is an exact global diffeomorphism, it is an isometry from $\mathcal{M}$ onto $\mathcal{V}$.

Prototype-space reduction from the (P_k, A_k) prototype. For $C \in \mathcal{M}$, $P_k \in \mathcal{M}$, and $A_k \in T_{P_k}\mathcal{M}$, set $z = \phi(C)$, $q_k = \phi(P_k)$, and $a_k = \mathrm{d}\phi_{P_k}[A_k]$. Isometry of the logarithm map gives

$$\mathrm{d}\phi_{P_k}\left[\mathrm{Log}^{g^\phi}_{P_k}(C)\right] = z - q_k. \tag{26}$$

The isometry also preserves the norm and inner product, so $\|A_k\|_{g^\phi_{P_k}} = \|a_k\|_{\mathcal{V}}$ and $\left\langle \mathrm{Log}^{g^\phi}_{P_k}(C), A_k\right\rangle_{g^\phi_{P_k}} = \langle z - q_k, a_k\rangle_{\mathcal{V}}$. It maps the hyperplane in (24) to

$$\phi\left(H_{A_k,P_k}\right) = \{u \in \mathcal{V} \mid \langle u - q_k, a_k\rangle_{\mathcal{V}} = 0\}\,. \tag{27}$$

The Euclidean distance to this hyperplane is

$$\mathrm{d}\left(C, H_{A_k,P_k}\right) = \frac{|\langle z - q_k, a_k\rangle_{\mathcal{V}}|}{\|a_k\|_{\mathcal{V}}}, \tag{28}$$

Substitution into (25) therefore gives

$$\begin{aligned}v_{g,k}(C; P_k, A_k) &= \mathrm{sgn}\left(\langle z - q_k, a_k\rangle_{\mathcal{V}}\right) \frac{\|a_k\|_{\mathcal{V}}\, |\langle z - q_k, a_k\rangle_{\mathcal{V}}|}{\|a_k\|_{\mathcal{V}}} \\ &= \langle z - q_k, a_k\rangle_{\mathcal{V}}\,.\end{aligned} \tag{29}$$

This is the prototype-space expression used in the main text directly from (P_k, A_k), without selecting a shared base point for the class parameters.

Affine-softmax realization. By the adjoint definition in Equation (20) and the choice $f^*(w_{g,k}) = a_k$, for every $v \in \mathcal{V}$ we have

$$\begin{aligned} w_{g,k}^\top f(v) &= \langle f^*(w_{g,k}), v\rangle_{\mathcal{V}} \\ &= \langle a_k, v\rangle_{\mathcal{V}}\,. \end{aligned} \tag{30}$$

Together with $b_{g,k} = -\langle q_k, a_k\rangle_{\mathcal{V}}$, this proves equality of the FC and CorMLR logits. Applying softmax proves the stated prediction equality.

Optimization equivalence. The forward equivalence does not by itself identify independent Euclidean SGD on the affine parameters with optimization of (P_k, A_k). Such a statement is valid only after specifying the metric on the tangent-bundle parameters. For one class, define

$$\Psi_g : T\mathcal{M} \longrightarrow \mathcal{V}_g \times \mathcal{V}_g, \qquad \Psi_g(P, A) = (\phi_g(P), \mathrm{d}\phi_{g,P}[A])\,. \tag{31}$$

Because ϕ_g is a global diffeomorphism, Ψ_g is a diffeomorphism with inverse

$$\Psi_g^{-1}(q, a) = \left(\phi_g^{-1}(q), \left(\mathrm{d}\phi_{g,\phi_g^{-1}(q)}\right)^{-1}[a]\right). \tag{32}$$

Equip $T\mathcal{M}$ with the pullback metric $h_g = \Psi_g^*\langle\cdot,\cdot\rangle_{\mathcal{V}_g\times\mathcal{V}_g}$. If F is a differentiable CorMLR loss and $f = F \circ \Psi_g^{-1}$ is its expression in the variables (q, a), then

$$\mathrm{d}\Psi_{g,x}\left[\mathrm{grad}_{h_g} F(x)\right] = \nabla f\left(\Psi_g(x)\right). \tag{33}$$

To verify (33), let $x = (P, A)$, $y = \Psi_g(x)$, and let ξ be any tangent vector at x. By the definition of the pullback metric and the chain rule,

$$\begin{aligned} h_{g,x}\left(\mathrm{grad}_{h_g} F(x), \xi\right) &= \left\langle \mathrm{d}\Psi_{g,x}\left[\mathrm{grad}_{h_g} F(x)\right], \mathrm{d}\Psi_{g,x}[\xi]\right\rangle \\ &= \mathrm{d}F_x[\xi] \\ &= \mathrm{d}f_y\left[\mathrm{d}\Psi_{g,x}[\xi]\right] \\ &= \langle \nabla f(y), \mathrm{d}\Psi_{g,x}[\xi]\rangle\,. \end{aligned} \tag{34}$$

Since $\mathrm{d}\Psi_{g,x}$ is onto, the last identity for every ξ implies (33).

The same isometry gives the exponential-map relation

$$\Psi_g\left(\mathrm{Exp}_x^{h_g}(\xi)\right) = \Psi_g(x) + \mathrm{d}\Psi_{g,x}[\xi]. \tag{35}$$

Indeed, Euclidean straight lines in the variables (q, a) pull back through Ψ_g to geodesics for h_g. Therefore one update with the induced Riemannian metric satisfies

$$\Psi_g(x^+) = \Psi_g(x) - \eta\nabla f\left(\Psi_g(x)\right), \qquad x^+ = \mathrm{Exp}_x^{h_g}\left(-\eta\,\mathrm{grad}_{h_g} F(x)\right). \tag{36}$$

For K classes, apply Ψ_g componentwise and use the corresponding product metric, where the inner product is the sum over class pairs. The result is the same Euclidean update on all pairs (q_k, a_k). Thus, the CorMLR classifier is realized exactly through Euclidean variables: (q_k, a_k) are the optimization variables, and the FC layer implements their logits through $f^*_{\mathrm{vec},g}(w_{g,k}) = a_k$ and $b_{g,k} = -\langle q_k, a_k\rangle_{\mathcal{V}_g}$. □

D Implementation Details for PHCM

D.1 Poincaré Aggregation

Let $C \in \mathrm{Cor}^+(d)$ and $L = \mathrm{Chol}(C)$. For each $i = 2, \ldots, d$, write the nontrivial part of the i-th row as $h_i = \left(u_i^\top, \rho_i\right)^\top$, where $u_i = L_{i,1:i-1}$ and $\rho_i = L_{ii}$. The unit-diagonal constraint on C gives $\|u_i\|^2 + \rho_i^2 = 1$ with $\rho_i > 0$. Stereographic projection therefore produces

$$p_i = \frac{u_i}{1 + \rho_i} \in \mathbb{P}^{i-1}. \tag{37}$$

For $M = d(d-1)/2$ and $\beta_m = B\left(m/2, 1/2\right)$, Poincaré β-Concat is

$$\mathrm{Concat}_\beta(p_2, \ldots, p_d) = \mathrm{Exp}_0^{\mathbb{P}^M}\left(\begin{bmatrix} \frac{\beta_M}{\beta_1}\,\mathrm{Log}_0^{\mathbb{P}^1}(p_2) \\ \vdots \\ \frac{\beta_M}{\beta_{d-1}}\,\mathrm{Log}_0^{\mathbb{P}^{d-1}}(p_d) \end{bmatrix}\right). \tag{38}$$

The factors β_M/β_i are precomputed and the logarithmic vectors are concatenated before one exponential map in $\mathbb{P}^M$.

D.2 Lorentz Aggregation

The same Cholesky row gives a Lorentz factor

$$\ell_i = \left(\frac{1}{\rho_i}, \frac{u_i}{\rho_i}\right) \in \mathbb{L}^{i-1}, \tag{39}$$

because

$$-\frac{1}{\rho_i^2} + \frac{\|u_i\|^2}{\rho_i^2} = -1. \tag{40}$$

H-Concat collects all spatial components in

$$s = \begin{bmatrix} u_2/\rho_2 \\ \vdots \\ u_d/\rho_d \end{bmatrix} \in \mathbb{R}^M \tag{41}$$

and restores the time component:

$$\text{H-Concat}(\ell_2, \ldots, \ell_d) = \begin{bmatrix} \sqrt{1 + \|s\|^2} \\ s \end{bmatrix} \in \mathbb{L}^M. \tag{42}$$

The resulting vector satisfies the Lorentz constraint by construction and can be passed directly to a Lorentz layer.

D.3 Closed-Form Hyperbolic Logits

Poincaré implementation. Following HNN++ (Shimizu et al., 2021), $\mathrm{GCorP\text{-}PHCM}_p$ parameterizes the class hyperplane by $z_k \in \mathbb{R}^M$ and $\gamma_k \in \mathbb{R}$. Let $[z_k] = z_k/\|z_k\|$. For $x \in \mathbb{P}^M$, define $\lambda_x = 2/\left(1 - \|x\|^2\right)$. Under $K = -1$, the logit is

$$v_k^{\mathrm{HNN++}}(x) = 2\|z_k\| \,\mathrm{arcsinh}\left(\lambda_x \left\langle x, [z_k]\right\rangle \cosh(2\gamma_k) - (\lambda_x - 1)\sinh(2\gamma_k)\right). \tag{43}$$

Lorentz implementation. Let $x \in \mathbb{L}^M$ and let $\widetilde{a}_k$ be an ambient Lorentz normal satisfying $\langle \widetilde{q}_k, \widetilde{a}_k\rangle_{\mathcal{L}} = 0$ and $\langle \widetilde{a}_k, \widetilde{a}_k\rangle_{\mathcal{L}} > 0$. Lorentz MLR (Bdeir et al., 2024) uses the signed-distance logit

$$v_k^{\mathbb{L}}(x) = \|\widetilde{a}_k\|_{\mathcal{L}} \,\mathrm{arcsinh}\left(\frac{\langle x, \widetilde{a}_k\rangle_{\mathcal{L}}}{\|\widetilde{a}_k\|_{\mathcal{L}}}\right), \qquad \|\widetilde{a}_k\|_{\mathcal{L}} = \sqrt{\langle \widetilde{a}_k, \widetilde{a}_k\rangle_{\mathcal{L}}}. \tag{44}$$

D.4 Numerical Stabilization

Before Cholesky decomposition, we symmetrize the correlation matrix and add diagonal jitter when required. The PHCM input-scale factors used for large-scale training are applied before the final hyperbolic classifier and are reported with the experimental setup in Section E.1. These operations do not change the output dimensions: $\mathrm{GCorP\text{-}PHCM}_p$ returns a point in $\mathbb{P}^M$, whereas $\mathrm{GCorP\text{-}PHCM}_l$ returns a point in $\mathbb{L}^M$ represented by $M + 1$ ambient components.

E EXPERIMENTAL SETUP DETAILS

E.1 IMAGENET-1K CLASSIFICATION

We follow the pooling pipeline of Song et al. (2022b). A backbone feature map is projected to $d = 256$ channels by a 1×1 convolution, centered, and pooled into a covariance matrix, after which the geometry-specific head is applied. All methods use the same reduction dimension of 256. ResNet-50 and ResNet-101 use a global batch size of 256, while ViT-B/16 and Swin-T use 1024. All backbones are trained with AdamW at learning rate 10^{-3} and weight decay 0.05 for 110 epochs with a cosine schedule and a 10-epoch warmup starting at 10^{-5}. We use label smoothing 0.1 and SyncBatchNorm. ResNet backbones use RandAugment without Mixup or CutMix, while ViT-B/16 and Swin-T additionally use Mixup 0.8 and CutMix 1.0. ECM keeps the exact Cholesky representation and requires no matrix logarithm, while LECM uses the default matrix logarithm. OLM and LSM use the Legendre approximation at order 6, and LSM uses one Newton step for $\mathcal{D}^\star(C)$. For the GCorP-PHCM heads we use input-scale factors of 0.5 for GCorP-PHCM$_l$ and 0.25 for GCorP-PHCM$_p$. Baselines cover the standard matrix-function family, namely GAVP, iSQRT-COV (Li et al., 2018), MPA (Song et al., 2022a), SVD-Padé (Song et al., 2021), Halley-SVD (Gu et al., 2025), and the partial-correlation method ISICE (Rahman et al., 2023). ISICE uses its published configuration with 5 iterations, sparsity 0.01, and learning rate 5.0. Stable per-batch time is measured with batch size 128 for ViT-B/16, 64 for Swin-T and ResNet-50, and 64 for ResNet-101. ResNet-101 runtime is reported in seconds per batch using the mean values shown in Table 2. For ResNet-50, the reported ISICE accuracies are obtained from the final checkpoint at epoch 110. For ViT-B/16, the selected checkpoints for GCorP-ECM and GCorP-LECM are epochs 108 and 107–108, respectively; both use the learning rate 10^{-3} stated above.

Table 6: ResNet-101 validation results at the selected checkpoint. All runs completed 110 training epochs.

Method	**Best epoch**	**Top-1 (%)**	**Top-5 (%)**
MPA (Song et al., 2022a)	110	**80.380**	95.114
ISICE (Rahman et al., 2023)	110	80.144	<u>95.206</u>
iSQRT-COV (Li et al., 2018)	110	79.164	94.264
Halley-SVD (Gu et al., 2025)	109	77.388	93.568
GCorP-OLM	106	<u>80.314</u>	**95.264**
GCorP-LSM	109	79.892	94.882
GCorP-PHCM$_p$	109	78.170	93.012
GCorP-PHCM$_l$	110	77.502	93.308

Table 7: Additional ImageNet-1K validation results for GCorP-ECM and GCorP-LECM.

Backbone	**Method**	**Red. dim.**	**Epoch**	**Val Top-1 (%)**	**Val Top-5 (%)**
ResNet-50	GCorP-ECM	256	110	76.55	93.14
ResNet-50	GCorP-LECM	256	110	76.35	93.17
ResNet-101	GCorP-ECM	256	110	78.62	94.28
ResNet-101	GCorP-LECM	256	110	77.95	93.85

E.2 ABLATION DETAILS

Covariance versus correlation. We hold the MPA pipeline, ViT-B/16 backbone, schedule, and parameter count fixed and change only the statistic entering the matrix-function step.

Table 8: Covariance versus correlation under the fixed MPA pipeline on ViT-B/16.

MPA input	**Top-1**	**Top-5**
Covariance	79.22	93.95
Correlation	**80.15**	**94.79**

Replacing covariance by correlation raises Top-1 from 79.22% to 80.15% and Top-5 from 93.95% to 94.79%, which isolates the effect of the correlation statistic.

Approximation-family sweep.

Table 9: Logarithm approximation-family sweep on ImageNet-1K with ViT-B/16. All non-swept hyperparameters are fixed to their default values.

Log method	Top-1 (%)	Top-5 (%)	Best epoch
Legendre	**81.26**	**95.62**	109
Taylor	80.96	95.32	106
Chebyshev	80.68	95.23	109
Laguerre	80.32	94.98	107
Padé	79.06	93.75	110

Expansion-order sweep.

Table 10: Expansion-order sweeps on ImageNet-1K and FD-loss. ImageNet uses ViT-B/16; FD-loss uses the no-SIM pMF-B/16 setting. FID is computed against the guided-diffusion (ADM) reference statistics throughout, matching Table 4. All non-swept hyperparameters are fixed to their default values.

ImageNet-1K, OLM			
Order	Top-1 (%)	Top-5 (%)	Best epoch
2	80.56	94.96	109
4	81.01	95.53	108
5	81.19	95.60	109
6	**81.26**	**95.62**	109
7	80.91	95.41	104
8	80.67	95.38	106
10	80.35	95.15	109
12	80.12	94.91	107
14	79.71	94.63	106

FD-loss, OLM (no-SIM)			
Order	FID (ADM)	FDr_6	IS
4	0.9978	11.1243	302.5314
6	0.8856	10.7855	298.5446
8	**0.8426**	**10.8067**	**295.4589**
10	0.8312	10.9185	292.2905
12	0.8228	10.7539	288.4436
14	0.8222	10.8701	285.3784
16	0.8219	10.8121	283.0732

FD-loss, LSM (no-SIM)				
Order	MI	FID (ADM)	FDr_6	IS
6	1	0.8864	10.7409	299.6676
6	2	0.8936	10.7596	299.1840
6	3	0.8902	10.7473	297.6188
8	**1**	**0.8427**	**10.6344**	**289.0**
12	1	0.8279	10.7893	288.7789
12	2	0.8220	10.7979	287.5426
12	3	0.8235	10.8424	288.7501

FD-loss LSM measurements cover orders 6 and 12 with one to three Newton steps, plus the order-8 one-step configuration reported in Table 4.

Newton-step sweep.

Table 11: LSM Newton-step sweep on ImageNet-1K with ViT-B/16. All non-swept hyperparameters are fixed to their default values.

Max iterations	Top-1 (%)	Top-5 (%)	Best epoch
1	**80.87**	95.23	107
2	80.84	95.35	106
3	80.85	95.31	110
4	80.70	95.23	109
5	80.86	95.34	103
100	80.83	**95.37**	110

PHCM input-scale sweep.

Table 12: PHCM input-scale sweeps on ImageNet-1K with ViT-B/16. All non-swept hyperparameters are fixed to their default values.

Head	Input scale	Top-1 (%)	Top-5 (%)
GCorP-PHCMp	**0.25**	**79.71**	93.59
GCorP-PHCMp	0.50	79.28	**94.02**
GCorP-PHCMp	0.75	78.96	93.75
GCorP-PHCMp	None	78.87	93.97
GCorP-PHCMl	0.25	71.78	90.56
GCorP-PHCMl	0.325	80.32	**95.04**
GCorP-PHCMl	**0.50**	**80.33**	94.94
GCorP-PHCMl	0.625	80.11	94.72
GCorP-PHCMl	0.75	78.96	94.08
GCorP-PHCMl	None	61.06	83.69

E.3 EEG Evaluation

Pipeline. Given an EEG segment, the pretrained LaBraM-Base model produces a global token y_0 and non-global spatiotemporal tokens. RHOP maps the latter to token-wise quotient-Gaussian descriptors, aggregates them into a Fréchet mean Y_m and tangent-space covariance Y_c, and forms $G = \mathrm{embed}(Y_m, Y_c)$. We retain this construction through G but replace the geometry-specific downstream head. For $g \in \{\mathrm{OLM}, \mathrm{LSM}\}$, we use

$$z_g = f_{\mathrm{vec},g}\left(\phi_g\left(\mathrm{Cor}(G)\right)\right), \qquad \widehat{y}_g = \mathrm{softmax}\left(W_g\left[y_0; z_g\right] + b_g\right). \tag{45}$$

For PHCM, let $\mathcal{H}_p = \mathbb{P}^D$ and $\mathcal{H}_\ell = \mathbb{L}^D$ denote the Poincaré and Lorentz model spaces and set $z_{\mathrm{PH},h} = \phi_{\mathrm{PH},h}(\mathrm{Cor}(G))$. We fuse the representation with a projected CLS token in the origin tangent space,

$$\widetilde{z}_{\mathrm{PH},h} = \mathrm{Exp}_0^{\mathcal{H}_h}\left(\left[P_{\mathrm{cls},h} y_0; \mathrm{Log}_0^{\mathcal{H}_h}\left(z_{\mathrm{PH},h}\right)\right]\right), \qquad h \in \{p, \ell\}, \tag{46}$$

and pass $\widetilde{z}_{\mathrm{PH},h}$ to the corresponding hyperbolic MLR. No GCorP branch applies SICE to G.

Setup. We evaluate on TUAB for binary abnormality detection and TUEV for six-class EEG-event classification. All TUAB and TUEV methods use the same pretrained LaBraM-Base initialization. Within each dataset, the methods share preprocessing, token-wise statistical construction, global-token fusion, data splits, training budget, and checkpoint-selection protocol, differing only in the downstream pooling head. TUEV models are trained for 50 epochs with learning rate 5×10^{-4} and weight decay 0.075. For TUAB, the checkpoint of each seed is selected by validation balanced accuracy, and configurations are compared by the resulting validation balanced accuracy averaged over three seeds. The selected configurations are GCorP-OLM $(4, 4)$, GCorP-ECM $(6, 2)$, GCorP-LECM $(6, 4)$, GCorP-LSM $(6, 4)$, GCorP-PHCM$_p$ $(k = 4, k' = 1, \mathrm{scale} = 0.5)$, and GCorP-PHCM$_l$ $(k = 4, k' = 1, \mathrm{scale} = 0.75)$. For TUEV, the checkpoint of each seed is selected by validation weighted F1, and configurations are compared by the resulting validation weighted F1 averaged over three seeds. The selected configurations are GCorP-ECM $(6, 1)$, GCorP-LECM $(2, 1)$, GCorP-OLM $(1, 4)$, GCorP-LSM $(6, 2)$, GCorP-PHCM$_p$ $(2, 1)$, and GCorP-PHCM$_l$ $(4, 4)$. TUAB timing is reported in seconds per iteration: 0.2019 for RHOP, 0.2039 for ISICE, 0.1777 for iSQRT-COV, 0.1415 for GCorP-OLM, 0.1450 for GCorP-ECM, 0.1453 for GCorP-LECM, 0.1517 for GCorP-PHCM$_l$, and 0.1527 for GCorP-PHCM$_p$. TUEV

timing is reported in seconds per iteration: 0.1150 for RHOP, 0.1455 for ISICE, 0.1130 for iSQRT-COV, 0.0749 for GCorP-ECM, 0.0764 for GCorP-LECM, 0.0732 for GCorP-OLM, 0.0843 for GCorP-LSM, 0.0772 for GCorP-PHCM$_p$, and 0.0768 for GCorP-PHCM$_l$. Timing standard deviations are omitted from the table. All methods use the same three random seeds. Results are reported as mean $\pm$ standard deviation over three runs. TUAB is evaluated using balanced accuracy, PR-AUC, and AUROC, while TUEV is evaluated using balanced accuracy, Cohen's kappa, and weighted F1.

TUEV configuration selection. For each geometry, we sweep the discrete pair (k, k'). Within each seed, the checkpoint with the highest validation weighted F1 is selected; the final configuration is the one with the highest mean validation weighted F1 across the three selected seed checkpoints. This procedure uses no test metric. After selection, all configurations are frozen before evaluating the TUEV test set. The selected configurations are summarized in Table 13, and the complete validation sweeps are reported in Tables 14 to 16.

Table 13: TUEV validation statistics for the selected configurations (mean $\pm$ standard deviation over three seeds). Configurations are selected by the highest mean validation weighted F1.

Method	Setting	Val BAcc.	Val Kappa	Val W. F1
GCorP-ECM	$(6, 1)$	70.73 ± 1.24	82.89 ± 1.45	92.17 ± 0.78
GCorP-LECM	$(2, 1)$	72.27 ± 2.31	83.59 ± 1.91	92.36 ± 0.93
GCorP-OLM	$(1, 4)$	71.51 ± 2.15	84.12 ± 1.60	92.46 ± 0.84
GCorP-LSM	$(6, 2)$	73.58 ± 2.01	83.64 ± 0.71	92.33 ± 0.18
GCorP-PHCM$_p$	$(2, 1)$	72.81 ± 1.39	80.92 ± 0.73	90.93 ± 0.43
GCorP-PHCM$_l$	$(4, 4)$	73.31 ± 2.52	82.90 ± 3.15	92.06 ± 1.58

Table 14: TUEV validation sweep for GCorP-ECM and GCorP-LECM.

(k, k')	GCorP-ECM			GCorP-LECM		
	BAcc.	Kappa	W. F1	BAcc.	Kappa	W. F1
$(1, 1)$	72.67 ± 4.94	82.26 ± 0.75	91.68 ± 0.46	71.37 ± 2.33	82.03 ± 1.29	91.65 ± 0.73
$(1, 2)$	73.01 ± 2.05	83.23 ± 1.21	91.95 ± 0.59	71.69 ± 1.10	82.54 ± 1.66	91.66 ± 0.36
$(1, 4)$	71.54 ± 3.75	81.49 ± 0.94	91.16 ± 0.48	71.64 ± 6.03	81.47 ± 1.54	91.19 ± 1.08
$(2, 1)$	71.21 ± 2.08	81.80 ± 2.76	91.30 ± 1.36	72.27 ± 2.31	83.59 ± 1.91	92.36 ± 0.93
$(2, 2)$	73.05 ± 3.55	83.04 ± 0.85	91.98 ± 0.29	73.96 ± 1.96	82.88 ± 0.75	91.90 ± 0.36
$(2, 4)$	71.24 ± 2.28	81.73 ± 0.55	91.38 ± 0.38	71.64 ± 4.33	81.95 ± 3.40	91.55 ± 1.61
$(4, 1)$	72.89 ± 2.92	83.13 ± 0.93	91.85 ± 0.18	73.51 ± 2.85	83.69 ± 1.47	92.23 ± 0.57
$(4, 2)$	71.92 ± 1.74	81.55 ± 1.08	91.36 ± 0.07	71.01 ± 4.63	80.69 ± 0.76	90.75 ± 0.57
$(4, 4)$	72.03 ± 3.01	83.11 ± 1.32	92.10 ± 0.55	70.35 ± 2.03	81.08 ± 1.39	91.13 ± 0.96
$(6, 1)$	70.73 ± 1.24	82.89 ± 1.45	92.17 ± 0.78	71.18 ± 2.98	82.80 ± 0.72	91.78 ± 0.35
$(6, 2)$	71.65 ± 1.68	82.87 ± 2.05	91.59 ± 0.90	75.64 ± 4.17	82.05 ± 1.06	91.37 ± 0.25
$(6, 4)$	69.25 ± 4.16	82.43 ± 3.08	91.37 ± 1.38	70.39 ± 3.61	81.77 ± 1.71	91.31 ± 0.73

Table 15: TUEV validation sweep for GCorP-OLM and GCorP-LSM.

(k, k')	GCorP-OLM			GCorP-LSM		
	BAcc.	Kappa	W. F1	BAcc.	Kappa	W. F1
$(1, 1)$	71.69 ± 5.13	80.72 ± 0.50	90.83 ± 0.23	72.67 ± 2.19	81.39 ± 0.68	91.12 ± 0.46
$(1, 2)$	74.42 ± 4.55	82.49 ± 1.86	91.70 ± 0.94	76.41 ± 3.23	82.11 ± 2.51	91.54 ± 1.39
$(1, 4)$	71.51 ± 2.15	84.12 ± 1.60	92.46 ± 0.84	69.61 ± 0.82	81.59 ± 0.66	91.03 ± 0.36
$(2, 1)$	70.88 ± 0.95	82.72 ± 0.85	91.58 ± 0.52	72.69 ± 0.49	82.72 ± 1.80	92.02 ± 0.86
$(2, 2)$	72.86 ± 2.68	82.89 ± 0.84	91.74 ± 0.73	69.48 ± 1.09	82.09 ± 0.63	91.48 ± 0.43
$(2, 4)$	69.79 ± 0.98	82.08 ± 0.58	91.47 ± 0.43	72.31 ± 4.55	80.89 ± 2.39	90.93 ± 1.19
$(4, 1)$	72.45 ± 2.72	83.10 ± 2.04	92.07 ± 0.88	72.83 ± 4.51	81.30 ± 0.79	90.91 ± 1.06
$(4, 2)$	68.71 ± 2.18	80.05 ± 2.12	90.22 ± 1.20	70.78 ± 1.15	81.73 ± 2.90	91.11 ± 1.61
$(4, 4)$	70.24 ± 2.21	80.92 ± 0.45	90.73 ± 0.20	70.72 ± 1.14	80.61 ± 2.29	90.96 ± 1.27
$(6, 1)$	70.65 ± 2.05	81.44 ± 1.60	91.22 ± 0.97	73.51 ± 3.03	82.91 ± 2.05	91.95 ± 1.04
$(6, 2)$	71.63 ± 2.80	81.62 ± 1.69	91.27 ± 1.16	73.58 ± 2.01	83.64 ± 0.71	92.33 ± 0.18
$(6, 4)$	73.92 ± 2.91	82.84 ± 0.97	91.83 ± 0.67	68.85 ± 2.87	80.51 ± 1.88	90.85 ± 1.17

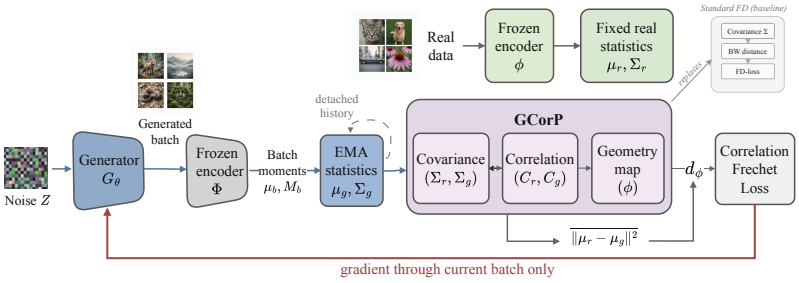


Figure 3: FD-loss post-training with GCorP and detached EMA statistics.

Table 16: TUEV validation sweep for GCorP-PHCM$_p$ and GCorP-PHCM$_l$.

(k, k')	GCorP-PHCM$_p$			GCorP-PHCM$_l$		
	BAcc.	Kappa	W. F1	BAcc.	Kappa	W. F1
$(1,1)$	69.46 ± 2.54	80.79 ± 3.26	90.81 ± 1.38	73.93 ± 2.90	81.02 ± 1.77	91.03 ± 0.74
$(1,2)$	67.81 ± 7.24	81.18 ± 0.68	90.37 ± 0.90	73.45 ± 2.00	80.37 ± 2.00	90.61 ± 1.17
$(1,4)$	70.33 ± 4.65	80.16 ± 1.54	90.72 ± 0.78	72.32 ± 3.28	82.28 ± 0.98	91.50 ± 0.45
$(2,1)$	72.81 ± 1.39	80.92 ± 0.73	90.93 ± 0.43	68.81 ± 3.03	80.26 ± 2.02	90.54 ± 0.81
$(2,2)$	74.23 ± 0.30	79.65 ± 0.89	90.50 ± 0.71	70.60 ± 5.37	81.13 ± 1.36	90.87 ± 0.83
$(2,4)$	69.22 ± 3.38	79.14 ± 0.78	90.18 ± 0.36	69.87 ± 0.95	80.75 ± 2.31	90.91 ± 1.24
$(4,1)$	70.28 ± 2.34	81.11 ± 2.83	90.88 ± 1.00	68.62 ± 4.85	81.86 ± 2.77	91.22 ± 1.49
$(4,2)$	70.32 ± 2.48	79.91 ± 1.31	90.29 ± 0.68	72.23 ± 4.75	82.18 ± 0.99	91.68 ± 0.67
$(4,4)$	74.24 ± 4.11	80.38 ± 2.15	90.51 ± 0.80	73.31 ± 2.52	82.90 ± 3.15	92.06 ± 1.58
$(6,1)$	70.08 ± 1.77	79.31 ± 2.54	90.18 ± 1.32	69.13 ± 9.48	82.61 ± 1.62	91.48 ± 1.33
$(6,2)$	71.08 ± 0.70	77.94 ± 1.87	89.85 ± 0.57	72.11 ± 0.87	82.32 ± 2.89	91.95 ± 1.09
$(6,4)$	67.47 ± 1.00	78.92 ± 0.22	89.97 ± 0.16	68.62 ± 2.99	81.80 ± 3.65	91.44 ± 2.01

E.4 FD Fine-Tuning

Differentiable feature statistics. We post-train a pretrained pMF-B/16 (Lu et al., 2026) and keep every feature-encoder parameter fixed. Following Yang et al. (2026), Inception-v3 produces a 2048-dimensional vector after its final global average pooling, whereas MAE and SigLIP2 use 1024- and 1152-dimensional CLS tokens from their final layers. Their input resolutions are 299, 224, and 224, respectively. For encoder k, image-level features $F_{b,k} = \{f_{i,k}\}_{i=1}^{B}$ are gathered across all devices, and the current-batch moments are

$$\mu_{b,k} = \frac{1}{B}\sum_{i=1}^{B} f_{i,k}, \qquad M_{b,k} = \frac{1}{B}\sum_{i=1}^{B} f_{i,k} f_{i,k}^{\top}. \tag{47}$$

The generated statistics at step s are

$$\begin{aligned} \mu_{g,k}^{(s)} &= \beta\,\mathrm{stopgrad}\left(\mu_{g,k}^{(s-1)}\right) + (1-\beta)\mu_{b,k}, \\ M_{g,k}^{(s)} &= \beta\,\mathrm{stopgrad}\left(M_{g,k}^{(s-1)}\right) + (1-\beta)M_{b,k}, \\ \Sigma_{g,k}^{(s)} &= M_{g,k}^{(s)} - \mu_{g,k}^{(s)}\left(\mu_{g,k}^{(s)}\right)^{\top}. \end{aligned} \tag{48}$$

We use $\beta = 0.999$ and initialize the detached history with statistics from 50,000 images produced by the base generator. The real statistics $(\mu_{r,k}, \Sigma_{r,k})$ are precomputed and fixed. The encoder weights are not optimized, but their forward operations remain differentiable with respect to the generated images. After evaluating (8), gradients flow through the current-batch moments and the encoder Jacobian to G_θ. The real statistics and historical EMA states remain outside this gradient path.

Representation distances. For the flat geometries, the second-order terms used in (8) are

$$\begin{aligned} d^2_{\mathrm{ECM}}(C_r, C_g) &= \|\phi_{\mathrm{ECM}}(C_r) - \phi_{\mathrm{ECM}}(C_g)\|_F^2, \\ d^2_{\mathrm{LECM}}(C_r, C_g) &= \|\phi_{\mathrm{LECM}}(C_r) - \phi_{\mathrm{LECM}}(C_g)\|_F^2, \\ d^2_{\mathrm{OLM}}(C_r, C_g) &= \|\mathrm{Log}^{\circ}(C_r) - \mathrm{Log}^{\circ}(C_g)\|_F^2, \\ d^2_{\mathrm{LSM}}(C_r, C_g) &= \|\mathrm{Log}^{\star}(C_r) - \mathrm{Log}^{\star}(C_g)\|_F^2. \end{aligned} \tag{49}$$

Table 17: Inception Score for the pMF-B/16 post-training experiments in Table 4. The SIM-MPA score is omitted because that run was numerically invalid.

Objective	Method	Non-SIM IS ↑	SIM IS ↑
Without FD-loss	pMF-B/16 (Lu et al., 2026)	254.6	254.6
With FD-loss	Exact covariance (Yang et al., 2026)	294.5	307.8
Approx. cov.	iSQRT-COV (Li et al., 2018)	372.1	364.2
	MPA (Song et al., 2022a)	–	–
	SVD-Padé (Song et al., 2021)	297.3	323.5
	Halley-SVD (Gu et al., 2025)	320.4	191.0
Correlation	GCorP-OLM	295.5	312.4
	GCorP-LSM	289.0	311.6
	GCorP-PHCM$_p$	284.3	301.8
	GCorP-PHCM$_l$	277.1	281.9
	GCorP-ECM	273.3	276.0
	GCorP-LECM	258.7	260.1

For PHCM, let $x_p = \phi_{\mathrm{PH},p}(C_r)$, $y_p = \phi_{\mathrm{PH},p}(C_g)$, $x_\ell = \phi_{\mathrm{PH},\ell}(C_r)$, and $y_\ell = \phi_{\mathrm{PH},\ell}(C_g)$. Under curvature -1,

$$d_{\mathbb{P}}(x_p, y_p) = \operatorname{arccosh}\left(1 + \frac{2\left\|x_p - y_p\right\|^2}{\left(1 - \left\|x_p\right\|^2\right)\left(1 - \left\|y_p\right\|^2\right)}\right), \qquad (50)$$
$$d_{\mathbb{L}}(x_\ell, y_\ell) = \operatorname{arccosh}\left(-\langle x_\ell, y_\ell\rangle_{\mathcal{L}}\right).$$

The two PHCM distances agree because the two model maps are related by the Lorentz–Poincaré isometry.

Evaluation metrics. FID (Heusel et al., 2017) and the multi-representation ratio FDr$_6$ are computed from 50,000 generated samples. Following Yang et al. (2026), for representation model i we define $\mathrm{FDr}_{\phi_i}(G) = \mathrm{FD}_{\phi_i}(G, T)/\mathrm{FD}_{\phi_i}(V, T)$ and $\mathrm{FDr}_6(G) = \frac{1}{6}\sum_{i=1}^{6}\mathrm{FDr}_{\phi_i}(G)$, where T and V denote the ImageNet training and validation sets. IS values are reported in Table 17.

Loss aggregation. Both settings balance the raw per-encoder losses using the stop-gradient rescaling of Yang et al. (2026):

$$\widehat{\mathcal{L}}_{\phi,k} = \frac{\mathcal{L}_{\phi,k}}{\mathrm{stopgrad}\left(\mathcal{L}_{\phi,k}\right) + \epsilon}, \qquad \mathcal{L} = \sum_k \widehat{\mathcal{L}}_{\phi,k}. \qquad (51)$$

We set $\epsilon = 0.01$. Inception-only contains one Inception-v3 term, while SIM sums the SigLIP2, Inception-v3, and MAE terms with equal weights.

Configuration. All loss variants post-train the complete pMF-B/16 generator for 50 epochs and use the same initialization, frozen encoders, real statistics, one-step sampling protocol, and evaluation procedure. We use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, learning rate 10^{-6}, zero weight decay, cosine decay, and five warmup epochs. Training uses bfloat16 precision, center cropping and horizontal flipping, no gradient clipping, and no model-weight EMA. Inception-only uses a global batch of 256, whereas SIM uses 128; both run on four A100-80G GPUs. OLM and LSM use an order-8 Legendre approximation, and LSM uses one Newton step. ImageNet classification uses order 6 as selected in Section 6.1.3. The PHCM variants use the Poincaré and Lorentz model maps defined in Section 5. The exact baseline evaluates the covariance term through a symmetric eigendecomposition, while the approximate covariance baselines use iSQRT-COV, MPA, or SVD-Padé. After post-training, we evaluate 50,000 independently generated images and report FID, FDr$_6$, and IS independently of the training loss.

iSQRT-COV iteration count. The default three iterations do not produce a converged SIM post-training run: online FID (ADM) remains 140.6722. With ten iterations the run only marginally converges, reaching FID (ADM) 9.4022, while sixteen iterations reduce it further to 3.6791. This improvement comes at substantial computational cost; the SIM iSQRT-COV configuration already requires 0.805 s/step in Table 4. The complete sweep is reported in Table 18.

Table 18: iSQRT-COV iteration-count ablation during pMF-B/16 SIM post-training. Entries are online FID (ADM) and FDr_6.

Iterations	FID (ADM) ↓	FDr_6 ↓
3	140.6722	146.7541
8	20.6029	20.5622
10	9.4022	10.9135
16	3.6791	9.5092

F Extended Related Work

F.1 Global Covariance Pooling

Second-order representations based on covariance matrices have become a standard tool for enhancing deep features by encoding pairwise channel statistics. Early work established the effectiveness of bilinear and outer-product pooling for fine-grained recognition (Lin et al., 2015) and enabled end-to-end learning of structured matrix layers via matrix backpropagation (Ionescu et al., 2015). Subsequent studies highlighted the importance of matrix functions for stabilizing training and improving recognition, including matrix square-root functions for bilinear features (Lin & Maji, 2017) and matrix power mappings for improved representation and generalization (Wang et al., 2021). The eigenvalue spectrum of covariance-pooled representations was later studied for fine-grained recognition (Song et al., 2023). Earlier analysis also investigated when second-order information is beneficial at large scale (Li et al., 2017).

Computational efficiency is another key consideration, since eigendecomposition or SVD can become a bottleneck at scale. To address this issue, Li et al. (2018) introduced an iterative matrix square-root computation based on the Newton–Schulz iteration that can be implemented using only matrix multiplications, enabling GPU-friendly training. Recent work revisited high-order iterations for SVD-based square-root computation to mitigate late-stage over-flattening (Gu et al., 2025). Beyond matrix functions, Riemannian neural networks learn layers directly on the SPD manifold (Huang & Van Gool, 2017; Brooks et al., 2019; Chen et al., 2024a), and MLR has been extended to general Riemannian geometries (Chen et al., 2024b). In parallel, several studies aim to interpret why GCP works beyond the numerical view. Wang et al. (2023) analyze GCP from an optimization perspective, and Chen et al. (2025) provide a unified explanation of matrix logarithm and matrix power mappings through the lens of Riemannian geometry.

F.2 Geometry of Full-Rank Correlation Matrices

The set of full-rank correlation matrices is a manifold rather than a vector space, and equipping it with a tractable metric is less straightforward than for the SPD cone. Thanwerdas & Pennec (2022) classify several families of metrics on $\mathrm{Cor}^+(d)$ that admit closed-form geodesics and logarithm maps, obtained as pullbacks along diffeomorphisms onto simpler prototype spaces. Two of these are used here. The off-log construction relies on the parametrization of Archakov & Hansen (2021), which shows that a symmetric hollow matrix determines a unique correlation matrix through an exponentially converging diagonal iteration. The log-scaled construction relies on the diagonal rescaling studied by Thanwerdas (2024), obtained as the zero of a monotone map on the positive orthant. Closest to our setting, Chen et al. (2026) build Riemannian networks over $\mathrm{Cor}^+(d)$, extending MLR and fully connected layers to manifold-valued inputs under five correlation geometries. Their classifiers use a fixed tangent space at the identity, and their inputs are already correlation matrices. We instead form the correlation matrix from a backbone feature map and use class-dependent base points and tangent vectors to interpret the ordinary OLM and LSM classification heads as CorMLR.

F.3 Hyperbolic Neural Networks

The PHCM branch first represents Cholesky rows as hyperbolic factors and then aggregates them for classification. Ganea et al. (2018) introduce hyperbolic layers and a Poincaré MLR based on gyrovector operations, and Shimizu et al. (2021) give the closed-form logit and the β-concatenation used by $\mathrm{GCorP\text{-}PHCM}_p$. Bdeir et al. (2024) derive the Lorentz MLR used by $\mathrm{GCorP\text{-}PHCM}_l$, for

which we aggregate the spatial components by H-Concat and restore the Lorentz time component. Product-manifold latent spaces with mixed curvature have also been studied for generative models (Skopek et al., 2020), and hyperbolic embeddings are established for hierarchical data (Nickel & Kiela, 2017; 2018).

F.4 High-Order Global Pooling

Beyond second-order statistics, high-order pooling aims to capture richer interactions and non-linear feature dependencies. Gao et al. (2019) extend second-order modeling into intermediate layers via global second-order pooling blocks, improving feature recalibration throughout the network. In the temporal domain, temporal-attentive covariance pooling integrates attention with covariance pooling to encode both intra-frame correlations and inter-frame cross-correlations for video recognition (Gao et al., 2021). For multimodal learning, factorized high-order pooling efficiently models higher-order interactions between vision and language (Yu et al., 2017; 2018). Recent transformer models have also adopted second-order classification heads via cross-covariance pooling (Xie et al., 2021). In biosignal modeling, Riemannian geometry has long been a standard tool for covariance-based EEG classification (Barachant et al., 2013), and recent EEG foundation models learn transferable representations from large-scale recordings (Jiang et al., 2024). Riemannian high-order pooling provides an SPD-manifold classification head that aggregates mean and second-order information and improves robustness across EEG benchmarks (Hu et al., 2026).