Riemannian Normal Coordinates from Distance Functions on Triangle Meshes

Anders Brun[1]
Centre for Image Analysis, Swedish University of Agricultural Sciences, Sweden
and
Ola Nilsson[2]
Department of Science and Technology, Linköping University, Sweden
and
Martin Reimers[3]
Department of Informatics and Centre of Mathematics for Applications, University of Oslo, Norway
and
Ken Museth[4]
Department of Science and Technology, Linköping University, Sweden
and
Hans Knutsson[5]
Department of Biomedical Engineering, Linköping University, Sweden

---

1. INTRODUCTION

of

With Antarctica being the exception, most of the world is visible in the map projection used in the emblem of the United Nations. This map, painted in white on light blue background and centered at the North Pole, is perhaps the most well known example of the so called azimuthal equidistant map projection. This \emph projection technique preserves angles and geodesic distances, from the base point on the North Pole to points in all directions up to a certain radius, making it useful to e.g. radio amateurs and airline travelers. same paragraph
This mapping is also a special case of the logarithm function or logarithm map, log map for short, defined on a surface embedded in space, or more generally a N-dimensional manifold. Given a basis for the surface tangent plane, at the base point for which the map is centered, the log map describes a coordinate system. These coordinates can directly be used to parametrize the surface, e.g. for decal compositing, or *local* texture mapping, and they are natural in the sense that geodesics emanating from the base point are mapped to straight lines.

The exponential map, as shown in figure 1, maps vectors $\boldsymbol{x}$ in the tangent plane of the base point $\boldsymbol{p}$, denoted T_pS, to points $\boldsymbol{x}$ on the surface S along geodesics. The Hopf-Rinow theorem states that if the closed and bounded subsets of S are compact, then $\exp_p(\boldsymbol{x})$ is defined for any $\boldsymbol{x} \in T_pS$, i.e. it is geodesically complete [do Carmo, 1992]. The log map, denoted $\log_p(\boldsymbol{x})$, is the inverse of the exponential map and takes points from the surface S to the tangent plane T_pS. It is always defined for geodesically complete surfaces and is furthermore uniquely determined within a certain radius of injectivity from p. The relations between these two maps ~~can be seen~~ in figure 1. Note in particular that the log function can be used to generate a local coordinate system for points on the surface by equipping T_pS with a suitable basis. Choosing an orthonormal

is illustrated

you still havent motivated the terminology exp/log map!

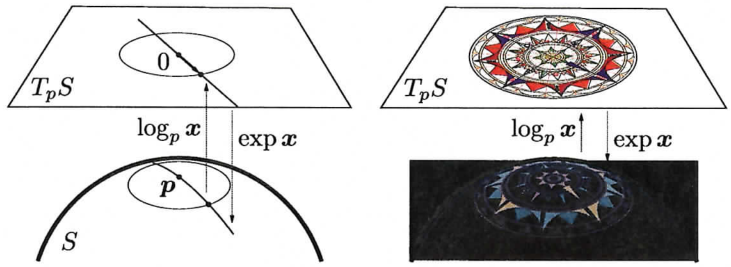

Fig. 1. Left: The log function is the inverse of the exponential map. Right: The log map can directly be used for local parametrization exemplified here with decal compositing.

basis yields the Riemannian Normal Coordinates (RNC).

map

Parameterizations of data by numerically estimating the log function was, to the best of our knowledge, first proposed ~~for use in~~ manifold learning [Brun et al., 2005]. It was a method to estimate $\log_p(\boldsymbol{x})$ from a set of data points sampled from a manifold. The method – named LogMap – was based on numerical estimation of gradients to an estimated geodesic distance function. In this paper we present a straightforward and modular way ~~of using~~ the LogMap method, together with accurate distance estimates, to compute local texture coordinates for a triangle mesh, as shown in figure 1.

in the context of

dubbed

to apply

where in fig 1 - right?

1.1 Contributions

contribution of

to

(and accurate?)

The main ~~result of~~ this paper is ~~that we~~ demonstrate a fast algorithm for computing local parametrizations, with decal compositing as the main application.

2. PREVIOUS WORK

A lot of research has been devoted to parametrization and finding suitable texture coordinates on ~~a~~ triangulated meshes. Optimization for minimal area/angle distortion have been key instruments to evaluate the generated parametrizations. See for example recent surveys [Floater and Hormann, 2005; Sheffer et al., 2006]. These approaches are in a sense global, and typically parametrize the whole surface. In this work we take on a different approach and look at a *local* parametrization that is "geometry aware".

Recent work in manifold learning has influenced computer graphics. Especially techniques for finding parametrizations of a set of data points, which are assumed to lie on a low-dimensional manifold embedded in a possibly high-dimensional Euclidean space. Well known methods, for instance Locally Linear Embedding (LLE) [Roweis and Saul, 2000] and Isomap [Tenenbaum et al., 2000], have been used in texture mapping applications [Peyré and Cohen, 2005]. One of the earliest examples of manifold learning, ~~which the reader might be familiar with,~~ ~~is~~ are the Self Organizing Maps [Kohonen, 1982]. It is an iterative and relatively slow procedure for adapting a low-dimensional mesh to a set of data points. Nowadays the term manifold learning is almost synonymous to *efficient* algorithms. They usually achieve computational efficiency by solving a large and sparse eigenvalue problem, or they guarantee, by other means, that the solution is unique and can be ~~obtained~~ quickly. computed

In this paper we focus on a technique that is related to both differential geometry and to manifold learndubbed

ing. In [Brun et al., 2005] the LogMap method was originally presented as a means to estimate RNC in a manifold known only from a discrete set of samples, possibly embedded in a high-dimensional Euclidean space. An approach, called ExpMap [Schmidt et al., 2006], with a similar goal has also been suggested for texture parametrization of 2D surfaces. Both methods are ~~similarly~~ based on geodesics on the surface, and as such the resulting parametrizations are approximations of $\log_p(x)$. The main differences between the LogMap and ExpMap is the fact that the latter is specialized for 2D manifolds and ~~builds upon reasoning about how~~ the 2D surface is curved and embedded in 3D. Furthermore ExpMap is not proven to be convergent and it introduces approximations ~~in the computations~~ assuming that the surface is flat or developable, at least locally. The ExpMap method also shares many similarities with another method ~~for~~ from manifold learning called Riemannian Manifold Learning [Lin et al., 2006], which also aims at estimating RNC. The LogMap on the other hand is a convergent method that largely focuses on intrinsic characteristics of the manifold and inherits much of its power from the choice of algorithm to calculate geodesic distances.

that

derived from the embedding of 2D surface in 3D space

attempts proposed by

Another method that focuses on geodesics distances is [Shapira and Shamir, 2009]. This method finds a local disc (with seems if necessary) and tries to preserve geodesic angle and distance at the perimeter and subsequently use Floaters mean-value coordinates to find a parametrization. Although local, the method still puts more emphasis on the boundary than the center of the disc, which is inverse to both LogMap and ExpMap, and consequently this parametrization is not an approximation of RNC. contrary

2.1 Computing Geodesic Distance

To mention just a few;

The eikonal equation governing geodesic distance has important applications in many fields. ~~It is used in~~ seismology, optics, computer vision, and medical imaging and analysis. A popular algorithm for solving the eikonal equation is Sethian's fast marching method [Sethian, 2001] (FMM). When used on triangular meshes this method, ~~although first order accurate,~~ FMM (and siblings) assume linear interpolation of distance values along edges resulting in linear wavefronts. This introduces a triangulation dependent relative error as large as 20% for reasonable meshes. Where do you have this number from? Reference!

lets hope Osher doesn't review this paper!!

it can also be 2nd order

To amend for this Reimers [Reimers, 2004] introduced a method with Euclidian precision. Its nonlinear updates makes a strict one-pass algorithm impossible since new values are not guaranteed to only depend on smaller values. Thus, the run-time complexity is not bounded. In practice Reimers' method performs comparably to FMM with respect to running times. The convergence is first order and independent of triangulation. Reimers' method is furthermore faster than the exact version of [Surazhsky et al., 2005] which is $O(n^2 \log n)$, and shows similar accuracy to their approximate version. Reimers' method also uses much less memory and is easier to implement.

what do you mean?

never use the term "exact" in the context of numerical schemes!

3. THEORY

3.1 The Jacobian and the metric

a a (don't use "the" if you haven't defined it yet)

The relation between the Jacobian of the mapping, and the metric expressed in texture coordinates is ~~easy to derive~~. Let $\mathbf{A}$ be a 2×3 matrix and $\mathbf{b}$ a 2-vector, mapping a surface triangle to texture coordinates.

readily derived

$$\mathbf{u} = \mathbf{A}\mathbf{x} + \mathbf{b}$$

you should also define u

The scalar product, inherited from the 3-D Euclidean space, between two vectors $\mathbf{u}$ and $\mathbf{v}$ in texture coordinates is then

$$\langle \mathbf{u}, \mathbf{v} \rangle = \langle \mathbf{A}^{-1}\mathbf{u}, \mathbf{A}^{-1}\mathbf{v} \rangle = \mathbf{u}^{\mathsf{T}}(\mathbf{A}\mathbf{A}^{\mathsf{T}})^{-1}\mathbf{v}.$$

Hence

~~It follows that~~ the metric can be expressed as

~~And for this reason, we have the following expression for the metric,~~

$$\mathbf{G} = (\mathbf{A}\mathbf{A}^{\mathsf{T}})^{-1}. \tag{1}$$

3.2 Local properties of Riemannian normal coordinates

Riemannian Normal Coordinates have certain properties that make them suitable for local calculations in differential geometry, ~~which we will see later in this paper~~. In 2-D, some of the most important are (adapted from [Lee, 1997]):

properties

(1) The coordinates of p are $(0, 0)$.

(2) A geodesic starting at p with velocity $\mathbf{v}$ is described in RNC by

$$\gamma_{\mathbf{v}}(t) = (tv_1, tv_2), \tag{2}$$

where t is the time variable.

$t = 0$

(3) The metric is the unit matrix in the origin,

$G(t)$ it's not obvious what the argument refers to

$$\mathbf{G}(\mathbf{0}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{3}$$

(4) The first partial derivatives of the metric vanishes at p. (well that's pretty obvious given 3; :-)

$$\left.\frac{\partial \mathbf{G}}{\partial u^i}\right|_{t=0} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}. \tag{4}$$

define U

(5) Any Euclidean ball in U is a geodesic ball in M. define M (manifold?)

boldface — is this the directional derivative?

(6) At any point q in U-p, d/dr is the velocity vector of the unit speed geodesic from p to q, and therefore has unit length with respect to g. (reformulate) don't you mean G ?

From Eq. 3 and Eq. 4 we can derive a Taylor approximation of the metric tensor in RNC at p, boldface

$$\mathbf{G}(\mathbf{u}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + O(||\mathbf{u}||^2), \text{when } \mathbf{u} \to \mathbf{0}. \tag{5}$$

"big O notation"

this is the proper name

The ~~expression~~ $O(||\mathbf{u}||^2)$ is matrix-valued, meaning that the components of the metric are constant close to (0,0), up to a second order variation. However, these properties are not unique to the ~~LOGMAP~~ LogMap. ~~and~~ In fact defining a mapping that makes a perpendicular mapping from the surface to the tangent plane, close to p, will have the same asymptotic behavior. Next we will therefore investigate the second order term of the metric in RNC.

LogMap

3.3 Local behavior of ~~LOGMAP~~

approximation

A second-order Taylor ~~expansion~~ of the metric in RNC is given by

$$\mathbf{G}(\mathbf{u}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \frac{1}{2}\mathbf{u}^{\mathsf{T}}\frac{\partial^2 \mathbf{G}}{\partial u \partial v}\mathbf{u} + O(||\mathbf{u}||^3), \text{when } \mathbf{u} \to \mathbf{0}. \tag{6}$$

this is somewhat confusing - you talk about 2'nd order but you still include the truncation term ($O(||u||^3)$) - so it is not actually an approximation

Due to the special properties of RNC, the second order term can also be expressed using the Riemannian curvature tensor R_{iajb} [Brewin, 1996],

$$\frac{1}{2}\frac{\partial^2 \mathbf{G}}{\partial u \partial v} = -\frac{1}{3} R_{iajb}. \tag{7}$$

In general the Riemannian curvature tensor has $n^2(n^2-1)/12$ degrees of freedom, which for surfaces ($n = 2$) amounts to only one. Expressed using the Gauss curvature, K, its components are

do you mean Gaussian curvature?

$$R_{abcd} = K(g_{ac}g_{db} - g_{ad}g_{cb}) \tag{8}$$

which for the particular case of RNC, where the metric is the unit matrix, yields $R_{2121} = R_{1212} = K$, $R_{2112} = R_{1221} = -K$ and all other components are zero. Inserted into Eq. 7 and Eq. 6 ~~it~~ this finally gives ~~the~~ formula

$$\mathbf{G}(\mathbf{u}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \mathbf{u}^{\mathsf{T}} \begin{bmatrix} -\frac{Ku_2u_2}{3} & \frac{Ku_2u_1}{3} \\ \frac{Ku_1u_2}{3} & -\frac{Ku_1u_1}{3} \end{bmatrix} \mathbf{u} + O(||\mathbf{u}||^3), \text{ when } \mathbf{u} \to \mathbf{0}. \tag{9}$$

same comment as for eq. 6

In global texture mapping, various measures of local texture distortion have been defined based on the Jacobian J_f of the function $f : \mathbf{R}^2 \to \mathbf{R}^3$ mapping texture coordinates to 3-D coordinates. This Jacobian is related to the metric in the texture coordinates by $\mathbf{G} = J_f^{\mathsf{T}} J_f$. A deviation of the metric from the unit matrix ~~imply~~ implies that the mapping is distorted in relatio to the metric inherited from the 3-D space, and since these local distortion should be rotation invariant these measures can be expressed in the eigenvalues ~~or singular values~~ of the metric. Now that we have ~~an expression~~ a good approximation for the metric ~~that is valid~~ close to p, we can easily derive Taylor expressions for these eigenvalues and singular values by solving the characteristic equation,

? you mean SVD?

nal

$$\mathbf{G}\mathbf{u} = \lambda \mathbf{u}, \tag{10}$$

which implies that

$$\left.\begin{array}{rcl} \lambda_1 & = & 1 \\ \lambda_2 & = & 1 - \frac{K}{3}||\mathbf{u}||^2 + O(||\mathbf{u}||^3) \end{array}\right\} \text{if} K \geq 0 \tag{11}$$

and

$$\left.\begin{array}{rcl} \lambda_1 & = & 1 - \frac{K}{3}||\mathbf{u}||^2 + O(||\mathbf{u}||^3) \\ \lambda_2 & = & 1 \end{array}\right\} \text{if} K \leq 0 \tag{12}$$

The ~~Corresponding~~ corresponding Taylor expansion for the singular values, $\sigma_i = \sqrt{\lambda_i}$, is simply

$$\left.\begin{array}{rcl} \sigma_1 & = & 1 \\ \sigma_2 & = & 1 - \frac{K}{6}||\mathbf{u}||^2 + O(||\mathbf{u}||^3) \end{array}\right\} \text{if} K \geq 0 \tag{13}$$

why not $\sqrt{K}$? You state it's simple but I don't follow :-)

and

$$\left.\begin{array}{rcl} \sigma_1 & = & 1 - \frac{K}{6}||\mathbf{u}||^2 + O(||\mathbf{u}||^3) \\ \sigma_2 & = & 1 \end{array}\right\} \text{if} K \leq 0. \tag{14}$$

We will now derive Taylor expressions for a number of previously defined distortion measures. In general, they all attain their minimum when the metric is the unit matrix and thus they are all minimal in RNC ~~close to 0~~. for $p \to 0$ (or $u \to 0$)!

Hormanns MIPS metrics [Hormann and Greiner, 2000], K_2 and K_F: as

defines the

$$K_2 = \frac{\sigma_1}{\sigma_2} = 1 + ||\mathbf{u}||^2 \frac{|K|}{6} + O(||\mathbf{u}||^3). \tag{15}$$

you need some kind of intro to the following list of metrics – as it stands the reader so no clue ~~why~~ whats coming...

$$K_2 = \frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1} = 2 + ||\mathbf{u}||^4 \frac{|K^2|}{6^2} + O(||\mathbf{u}||^5). \tag{16}$$

Conformal energy [Pinkall and Polthier, 1993]: is given by

$$K_C = (\sigma_1 - \sigma_2)^2/2 = \frac{K^2}{72}||\mathbf{u}||^4 + O(||\mathbf{u}||^5). \tag{17}$$

Green – Lagrange deformation tensor [Maillot et al., 1993]:

$$K_G = (\sigma_1 - 1)^2 + (\sigma_2 - 1)^2 = \frac{K^2}{36}||\mathbf{u}||^4 + O(||\mathbf{u}||^5). \tag{18}$$

Combined Energy [Degener et al., 2003]:

$$K_\theta = \left(\frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1}\right)\left(\sigma_1\sigma_2 + \frac{1}{\sigma_1\sigma_2}\right)^\theta = 2^{\theta+1} + \frac{2^{\theta+1}(\theta+1)K^2}{72}||\mathbf{u}||^4 + O(||\mathbf{u}||^5). \tag{19}$$

Dirichlet energy [Pinkall and Polthier, 1993]:

$$E_D = \frac{1}{2}(\sigma_1^2 + \sigma_2^2) = 1 - \frac{K}{6}||\mathbf{u}||^2 + O(||\mathbf{u}||^3). \tag{20}$$

Stretch energies [Sorkine et al., 2002; Sander et al., 2001]:

$$E_2 = \sqrt{E_D} = 1 - \frac{K}{12}||\mathbf{u}||^2 + O(||\mathbf{u}||^3). \tag{21}$$

$$E_\infty = \sigma_1 = \max(1, 1 - \frac{K}{6}||\mathbf{u}||^2 + O(||\mathbf{u}||^3). \tag{22}$$

$$E_S = \max(\sigma_1, \sigma_2^{-1}) = 1 + \frac{|K|}{6}||\mathbf{u}||^2 + O(||\mathbf{u}||^3). \tag{23}$$

this is not a real sentence?

The Riemannian metric between the unit matrix and the metric tensor. This quantity has not been used previously in the texture mapping literature, but is commonly used in Diffusion Tensor MRI to measure the distance between two tensors. It amounts to the geodesic length between two positive definite matrices, when a certain metric is assumed in the space of all positive definite matrices. (Add citation, check calculation!!):

$$D_R = \left[\log^2 \lambda_1 + \log^2 \lambda_2\right]^{1/2} = \frac{|K|}{6}||\mathbf{u}||^2 + O(||\mathbf{u}||^4). \tag{24}$$

define λ_1 and λ_2 - is it the two lowest or heighest eigen values?

not a good description

In summary, all of the above distortion measures are radially dependent ~~in~~ on RNC close to p. They are all at least second-order flat close to p. Finally they all depend on the Gaussian curvature K, having smaller distortion the less curvature we have on the surface. However, They are not at all dependent on the mean curvature, i.e. they do not depend on the particular embedding of the surface in space. These characteristics are not unique to RNC. However, a simple orthogonal projection from the surface to the tangent plane will not have these properties, e.g. the distortion will generically not be radially dependent.

Finally they all lead to reduced distortion as Gusion curvature decreases.

This section really belongs in an appendix

incosistant - in the beginning of this paper you use "LogMap"

3.4 On global optimality of LOGMAP

(Reformulate, this is exactly like in my thesis...) One may ask why the log map is useful to perform dimension reduction and visualization for data points. Suppose we are looking for some mapping $f: M \rightarrow \mathbf{R}^m$ such that

define: p, d and f

(1) $f(p) = \mathbf{0}$,

(2) $d(p, x) = ||f(x)||$,

(3) $d(y, x) \approx ||f(y) - f(x)||$ when $y \approx p$.

In short, this would be a mapping that preserves all distances exactly between points $x \in M$ and the base point $p \in M$. For points $y \in M$ that are close to p, distances are approximately preserved. It turns out that this mapping is the $\log_p(x)$ mapping and it is expressed in the following theorem.

Theorem: Suppose $f: M \rightarrow \mathbf{R}^n$ is a continuous mapping and $f(p) = \mathbf{0}$. Then

$$d(x, y) = ||f(x) - f(y)|| + ||f(y)||^2 B(x, y), \tag{25}$$

for some bounded function B, if an only if

$$f(x) = \mathbf{A} \log_p(x), \ \mathbf{A} \in O(n), \tag{26}$$

where $O(n)$ is the group of orthogonal transformations, $\mathbf{A}^T\mathbf{A} = \mathbf{I}$.

Proof:

$$\exists B : d(x, y) = ||f(x) - f(y)|| + ||f(y)||^2 B(x, y) \Leftrightarrow \tag{27}$$

via taylor approximation on both sides

$$\exists B_1 : d(x, p) + \langle \nabla_p d(x, p), \log_p(y) \rangle = \tag{28}$$

$$||f(x)|| - \frac{\langle f(x), f(y) \rangle}{||f(x)||} + B_1(x, y)||f(y)||^2 \Leftrightarrow \tag{29}$$

$$\exists B_2 : \frac{\langle \log_p(x), \log_p(y) \rangle}{d(x, p)} = \frac{\langle f(x), f(y) \rangle}{||f(x)||} + B_2(x, y)||f(y)||^2 \Leftrightarrow \tag{30}$$

$$\exists B_3 : \frac{\langle \log_p(x), \log_p(y) \rangle}{d(x, p) d(y, p)} = \frac{\langle f(x), f(y) \rangle}{||f(x)|| ||f(y)||} + B_3(x, y)||f(y)|| \Leftrightarrow \tag{31}$$

$$f(x) = \mathbf{A} \log_p(x), \ \mathbf{A} \in O(n), \tag{32}$$

The $\Leftarrow$ in the last step is obvious, while the $\Rightarrow$ follows from the fact that the expression should be valid for y arbitrary close to the base point p.

From this result we can state that LogMaps, or rather the true $\log_p(x)$, is the optimal mapping in the above sense, i.e. it is the most linear mapping centered at the base point p.

3.5 Noise

Several kinds of noise will affect the quality of the logmap. To mention a few:

Geometric noise. , ~~the~~ due to the quality of the mesh model. The mesh may exhibit a granular structrue ~~due to~~ caused by, for instance, noise during the digitization of an object.

Discretization error (geometry). , an insufficient number of triangles in the model make the surface non-smooth. This error applies both to the distances computed and the discretization (stencil) of the gradient operator.

~~truncation~~ ok

Interpolation or model error (distances). , from the choice of implementation of geodesic distance computation. This error can range from significant, for instance, when the Dijkstra algorithm is used, to low when the proposed method by Reimers is used. There also exist algorithms for triangular meshes that deliver exact distances, up to numerical precision. – references

Truncation error. , if the iterative procedure for calculating the distances is terminated before convergence, a truncation error occurs. – well most distance computations (e.g. FMM) have inherent truncation errors!

I would leave this out

Round-off errors. , from the implementation of floating point numbers. For this application, the LOGMAP, this noise ~~almost~~ always orders of magnitude lower than the other kinds of noise presented above.

There are different ways to ~~combat~~ address noise. For the LOGMAP algorithm, it is up to the user to decide what to consider as geometric noise. Discretization and model errors are more objective in nature. The single most sensitive step in the LOGMAP algorithm is the gradient calculation, which is known to be an operator prone to noise in signal and image processing. Ways to combat noise include: increasing mesh resolution, smoothing the mesh to remove geometric noise and remove sharp edges and design robust ways to estimate the gradient. The latter include smoothing calculated distance maps, choosing a larger stencil for the gradient estimation and possibly also increase the number of points in the stencil. Further improvements include to adaptively increase resolution of the mesh close to the point p, to improve the angular resolution of the gradient estimation, which is particularly important for distance computation schemes such as Dijkstras algorithm and Fast Marching where the error close to a point source can be significant.

how are they different?

4. METHOD

We derive a concise formula for how $\log_p(\boldsymbol{x})$ can be computed by considering some results related to how the so called intrinsic mean is computed [Karcher, 1977; Fletcher et al., 2004]. Let $\{\boldsymbol{x}_i\}$ be N data points in a surface S and seek the minimizer to the function

$$f(\boldsymbol{p}) = \frac{1}{2N}\sum_{i=1}^{N} d^2(\boldsymbol{p}, \boldsymbol{x}_i), \tag{33}$$

where $d^2(\boldsymbol{p}, \boldsymbol{x}_i)$ is the squared geodesic distance between points $\boldsymbol{p}$ and $\boldsymbol{x}_i$. It is then shown in [Karcher, 1977] that the gradient of f is

$$\nabla f(\boldsymbol{p}) = -\frac{1}{N}\sum_{i=1}^{N} \log_{\boldsymbol{p}}(\boldsymbol{x}_i). \tag{34}$$

Setting $N = 1$ and $\boldsymbol{x}_1 = \boldsymbol{x}$ gives the following formula,

$$\log_p(\boldsymbol{x}) = -\frac{1}{2}\nabla_{\boldsymbol{p}} d^2(\boldsymbol{p}, \boldsymbol{x}). \tag{35}$$

Here the subscript on the del operator is used to note that the gradient of the squared distance function is evaluated with respect to the point $\boldsymbol{p}$ on S.