

gDLS*: Generalized Pose-and-Scale Estimation Given Scale and Gravity Priors

Supplemental Material

Victor Fragoso
Microsoft
victor.fragoso@microsoft.com

Joseph DeGol
Microsoft
joseph.degol@microsoft.com

Gang Hua
Wormpex AI
ganghua@gmail.com

I. Introduction

In this document, we present the full mathematical derivation of gDLS* shown in the main submission. This derivation provides closed form equations that allow a user to implement the proposed method.

II. Incorporating Priors via Regularizers

Section 3 of the main submission describes the proposed formulation to include scale and rotation priors using regularizers. This formulation is:

$$J' = J(R, \mathbf{t}, s, \boldsymbol{\alpha}) + \underbrace{\lambda_s (s_0 - s)^2}_{J_s} + \underbrace{\lambda_g \|\mathbf{g}_Q \times R\mathbf{g}_W\|^2}_{J_g}, \quad (\text{i})$$

where

$$J(R, \mathbf{t}, s, \boldsymbol{\alpha}) = \sum_{i=1}^n \|\alpha_i \mathbf{r}_i - (R\mathbf{p}_i + \mathbf{t} - s\mathbf{c}_i)\|^2 \quad (\text{ii})$$

is the gDLS [2] pose-and-scale re-projection error cost function from the n 2D-3D correspondences, J_s is the scale prior regularizer, and J_g is the gravity direction constraint imposing a rotation prior.

II.i. Cost Function Depending Only on Rotation

As mentioned in Section 3, the first step to obtain a cost function that we can minimize in a single shot is to rewrite Eq. (i) as a function of the rotation matrix R . To do so, we define

$$\mathbf{x} = [\alpha_1 \quad \dots \quad \alpha_n \quad s \quad \mathbf{t}^\top]^\top, \quad (\text{iii})$$

a vector holding the depths for each i -th point α_i , the scale s , and translation vector \mathbf{t} . We know that the optimal depths, translation, and scale \mathbf{x}^* vanish the gradient $\nabla_{\mathbf{x}} J'$ of the cost function J' , *i.e.*,

$$\nabla_{\mathbf{x}} J' |_{\mathbf{x}=\mathbf{x}^*} = [\nabla_{\mathbf{x}} J + \nabla_{\mathbf{x}} J_s]_{\mathbf{x}=\mathbf{x}^*} = 0. \quad (\text{iv})$$

To satisfy this constraint, we calculate each of the gradients $\nabla_{\mathbf{x}} J$ and $\nabla_{\mathbf{x}} J_s$:

$$\begin{aligned} \nabla_{\mathbf{x}} J &= 2A^\top A \mathbf{x} - 2A W \mathbf{b} \\ \nabla_{\mathbf{x}} J_s &= 2P \mathbf{x} - 2P \mathbf{x}_0, \end{aligned} \quad (\text{v})$$

where

$$\begin{aligned} A &= \begin{bmatrix} \mathbf{r}_1 & & \mathbf{c}_1 & -I \\ & \ddots & \vdots & \vdots \\ & & \mathbf{r}_n & \mathbf{c}_n & -I \end{bmatrix}, \mathbf{b} = \begin{bmatrix} \mathbf{p}_1 \\ \vdots \\ \mathbf{p}_n \end{bmatrix} \\ P &= \begin{bmatrix} 0_{n \times n} & & \\ & \lambda_s & \\ & & 0_{3 \times 3} \end{bmatrix}, W = \begin{bmatrix} R & & \\ & \ddots & \\ & & R \end{bmatrix}, \end{aligned} \quad (\text{vi})$$

and $\mathbf{x}_0 = [0_n^\top \ s_0 \ 0_3^\top]^\top$. Note that

$$J_s = \lambda_s (s_0 - s)^2 = (\mathbf{x} - \mathbf{x}_0)^\top P (\mathbf{x} - \mathbf{x}_0). \quad (\text{vii})$$

Thus, we can rewrite Eq. (iv) into

$$\nabla_{\mathbf{x}} J' = A^\top A \mathbf{x} - A W \mathbf{b} + P \mathbf{x} - P \mathbf{x}_0 = 0, \quad (\text{viii})$$

by combining Equations (iv) and (v). Rearranging terms of Eq. (viii) yields Eq. (5) in the main submission, which is

$$\mathbf{x} = (A^\top A + P)^{-1} A^\top W \mathbf{b} + (A^\top A + P)^{-1} P \mathbf{x}_0. \quad (\text{ix})$$

In order to obtain the simplified version of Eq. (ix) shown in Eq. (5) of the main submission and inspired by [2], we rewrite

$$\begin{aligned} A^\top A + P &= \begin{bmatrix} I_{n \times n} & B'_{n \times 4} \\ B'^\top_{n \times 4} & D'_{4 \times 4} \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} I_{n \times n} & B_{n \times 4} \\ B^\top_{n \times 4} & D_{4 \times 4} \end{bmatrix}}_{A^\top A} + \underbrace{\begin{bmatrix} 0_{n \times n} & & \\ & \lambda_s & \\ & & 0_{3 \times 3} \end{bmatrix}}_P, \end{aligned} \quad (\text{x})$$

where

$$B = \begin{bmatrix} \mathbf{r}_1^\top \mathbf{c}_1 & -\mathbf{r}_1^\top \\ \vdots & \vdots \\ \mathbf{r}_n^\top \mathbf{c}_n & -\mathbf{r}_n^\top \end{bmatrix} \quad (\text{xi})$$

$$D = \begin{bmatrix} \sum_{i=1}^n \mathbf{c}_i^\top \mathbf{c}_i & \sum_{i=1}^n -\mathbf{c}_i^\top \\ \sum_{i=1}^n -\mathbf{c}_i & nI \end{bmatrix},$$

and I is the identity matrix. It is important to mention that $A^\top A$ is exactly the same as that of gDLS [2], and thus

$$A^\top A + P = \begin{bmatrix} I_{n \times n} & B_{n \times 4} \\ B_{n \times 4}^\top & D'_{4 \times 4} \end{bmatrix} \quad (\text{xii})$$

$$D'_{4 \times 4} = \begin{bmatrix} \lambda_s + \sum_{i=1}^n \mathbf{c}_i^\top \mathbf{c}_i & \sum_{i=1}^n -\mathbf{c}_i^\top \\ \sum_{i=1}^n -\mathbf{c}_i & nI \end{bmatrix}.$$

Eq. (ix) requires the inverse of $(A^\top A + P)^{-1}$. To compute a closed form relationship, we use the following block matrix expression

$$(A^\top A + P)^{-1} = \begin{bmatrix} E_{n \times n} & F_{n \times 4} \\ G_{4 \times n} & H_{4 \times 4} \end{bmatrix}. \quad (\text{xiii})$$

Through block matrix inversion, we obtain the following closed-form block matrices:

$$\begin{aligned} E &= I + BHB^\top \\ F &= -BH \\ G &= -HB^\top \\ H &= (D' - Y)^{-1} \\ Y &= \begin{bmatrix} \sum_{i=1}^n \mathbf{c}_i^\top \mathbf{r}_i \mathbf{r}_i^\top \mathbf{c}_i & \sum_{i=1}^n -\mathbf{c}_i^\top \mathbf{r}_i \mathbf{r}_i^\top \\ \sum_{i=1}^n -\mathbf{r}_i \mathbf{r}_i^\top \mathbf{c}_i & \sum_{i=1}^n \mathbf{r}_i \mathbf{r}_i^\top \end{bmatrix} \end{aligned} \quad (\text{xiv})$$

Like in gDLS [2], we use matrices U , S , and V to simplify Eq. (ix), *i.e.*,

$$(A^\top A + P)^{-1} A^\top = \begin{bmatrix} U \\ S \\ V \end{bmatrix}, \quad (\text{xv})$$

where

$$U = \begin{bmatrix} \mathbf{r}_1^\top & & \\ & \ddots & \\ & & \mathbf{r}_n^\top \end{bmatrix} + B \begin{bmatrix} S \\ V \end{bmatrix}$$

$$\begin{bmatrix} S \\ V \end{bmatrix} = -HB^\top \begin{bmatrix} \mathbf{r}_1^\top & & \\ & \ddots & \\ & & \mathbf{r}_n^\top \end{bmatrix} + H \begin{bmatrix} \mathbf{c}_1^\top & \cdots & \mathbf{c}_n^\top \\ -I & \cdots & -I \end{bmatrix}$$

$$= H \begin{bmatrix} \mathbf{c}_1^\top - \mathbf{c}_1 \mathbf{r}_1 \mathbf{r}_1^\top & \cdots & \mathbf{c}_n^\top - \mathbf{c}_n \mathbf{r}_n \mathbf{r}_n^\top \\ \mathbf{c}_1 \mathbf{c}_1^\top - I & \cdots & \mathbf{c}_n \mathbf{c}_n^\top - I \end{bmatrix}. \quad (\text{xvi})$$

We can simplify Eq. (ix) further. To do this, we focus on simplifying the term encoding the scale prior, yielding

$$(A^\top A + P)^{-1} P \mathbf{x}_0 = \begin{bmatrix} E & F \\ G & H \end{bmatrix} \underbrace{\begin{bmatrix} 0 & \\ & \lambda_s \\ & & 0 \end{bmatrix}}_P \underbrace{\begin{bmatrix} 0 \\ s_0 \\ 0 \end{bmatrix}}_{\mathbf{x}_0},$$

$$= \lambda_s s_0 \underbrace{\begin{bmatrix} F_1 \\ B_1 \end{bmatrix}}_1 \quad (\text{xvii})$$

where F_1 and B_1 are the first column of the matrix F and B , respectively. Combining Equations (xvii) and (xv) allows us to rewrite Eq. (ix) as follows:

$$\mathbf{x} = \begin{bmatrix} U \\ S \\ V \end{bmatrix} W \mathbf{b} + \lambda_s s_0 \mathbf{l}, \quad (\text{xviii})$$

which is the bottom part of Eq. (5) in the main submission. Eq. (xviii) provides a linear relationship between depths, scale, and translation and the rotation matrix. The explicit relationships are the following

$$\begin{aligned} \alpha_i(R) &= \mathbf{u}_i^\top W \mathbf{b} + \lambda_s s_0 \mathbf{l}_i \\ s(R) &= S W \mathbf{b} + \lambda_s s_0 \mathbf{l}_{n+1} \\ \mathbf{t}(R) &= V W \mathbf{b} + \lambda_s s_0 \mathbf{l}_t, \end{aligned} \quad (\text{xix})$$

where \mathbf{u}_i^\top is the i -th row of matrix U , \mathbf{l}_j is the j -th entry of the vector \mathbf{l} , and \mathbf{l}_t corresponds to the last three entries of the vector \mathbf{l} . Specifically, the entries of vector \mathbf{l} are

$$\mathbf{l} = \begin{bmatrix} \mathbf{l}_1 \\ \vdots \\ \mathbf{l}_n \\ \mathbf{l}_{n+1} \\ \mathbf{l}_t \end{bmatrix} = \begin{bmatrix} F_{1,1} \\ \vdots \\ F_{n,1} \\ H_{1,1} \\ H_{2:4,1} \end{bmatrix}, \quad (\text{xx})$$

where $H_{2:4,1}$ represent the last three entries of the first column of H . We can use these explicit relationships (*i.e.*, Eq. (i) and Eq. (xx)) to rewrite the main cost function as one depending only on rotation parameters. To do so as clearly as possible, we define

$$\begin{aligned}
\mathbf{e}_i &= \alpha_i(R)\mathbf{r}_i - (R\mathbf{p}_i + \mathbf{t}(R) - s(R)\mathbf{c}_i) \\
&= (\mathbf{u}_i^\top W\mathbf{b} + \lambda_{s_0} F_{i,1})\mathbf{r}_i - R\mathbf{p}_i \\
&\quad - (VW\mathbf{b} + \lambda_{s_0} H_{2:4,1}) \\
&\quad + (SW\mathbf{b} + \lambda_{s_0} H_{1,1})\mathbf{c}_i \\
&= \underbrace{\mathbf{u}_i^\top W\mathbf{b}\mathbf{r}_i - R\mathbf{p}_i - VW\mathbf{b} + SW\mathbf{b}\mathbf{c}_i}_{\boldsymbol{\eta}_i} \quad (\text{xxi}) \\
&\quad + \underbrace{\lambda_{s_0} (F_{i,1}\mathbf{r}_i - H_{2:4,1} + H_{1,1}\mathbf{q}_i)}_{\mathbf{k}_i} \\
&= \boldsymbol{\eta}_i + \mathbf{k}_i
\end{aligned}$$

As noted in gDLS [2] paper, $\boldsymbol{\eta}_i$ can be factored out as follows:

$$\begin{aligned}
\boldsymbol{\eta}_i &= (\mathbf{r}_i\mathbf{r}_i^\top - I)(R\mathbf{p}_i - SW\mathbf{b}\mathbf{c}_i + VW\mathbf{b}) \\
&= \underbrace{(\mathbf{r}_i\mathbf{r}_i^\top - I) [L(\mathbf{p}_i) \quad -\mathbf{c}_i SL(\mathbf{b}) \quad VL(\mathbf{b})]}_{M_i} \text{vec}(R), \quad (\text{xxii})
\end{aligned}$$

where $\text{vec}(R)$ vectorizes a rotation matrix R , and $L(\mathbf{z})$ is a function that computes a matrix such that $R\mathbf{z} = L(\mathbf{z})\text{vec}(R)$. Since we use the rotation representation of Upnp [1], *i.e.*,

$$\text{vec}(R) = [q_1^2 \ q_2^2 \ q_3^2 \ q_4^2 \ q_1q_2 \ q_1q_3 \ q_1q_4 \ q_2q_3 \ q_2q_4 \ q_3q_4]^\top, \quad (\text{xxiii})$$

then the function $L(\cdot)$ is

$$L(\mathbf{z})^\top = \begin{bmatrix} \mathbf{z}_1 & \mathbf{z}_2 & \mathbf{z}_3 \\ \mathbf{z}_1 & -\mathbf{z}_2 & -\mathbf{z}_3 \\ -\mathbf{z}_1 & \mathbf{z}_2 & -\mathbf{z}_3 \\ -\mathbf{z}_1 & -\mathbf{z}_2 & \mathbf{z}_3 \\ 0 & -2\mathbf{z}_3 & 2\mathbf{z}_2 \\ 2\mathbf{z}_3 & 0 & -2\mathbf{z}_1 \\ -2\mathbf{z}_2 & 2\mathbf{z}_1 & 0 \\ 2\mathbf{z}_2 & 2\mathbf{z}_1 & 0 \\ 2\mathbf{z}_3 & 0 & 2\mathbf{z}_1 \\ 0 & 2\mathbf{z}_3 & 2\mathbf{z}_2 \end{bmatrix}. \quad (\text{xxiv})$$

By substituting the relationships shown in Eq. (xix) and the factorizations shown in Eq. (xxii) into Eq. (i), we obtain

the following relationships:

$$\begin{aligned}
J'_{\text{gDLS}} &= \sum_{i=1}^n \mathbf{e}_i^\top \mathbf{e}_i = \sum_{i=1}^n \boldsymbol{\eta}_i^\top \boldsymbol{\eta}_i + 2\mathbf{k}_i^\top \boldsymbol{\eta}_i + \mathbf{k}_i^\top \mathbf{k}_i \\
&= \sum_{i=1}^n \text{vec}(R)^\top M_i^\top M_i \text{vec}(R) + 2\mathbf{k}_i^\top M_i \text{vec}(R) + \mathbf{k}_i^\top \mathbf{k}_i \\
&= \text{vec}(R)^\top \underbrace{\left(\sum_{i=1}^n M_i^\top M_i \right)}_{M_{\text{gDLS}}} \text{vec}(R) + \\
&\quad 2 \underbrace{\left(\sum_{i=1}^n \mathbf{k}_i^\top M_i \right)}_{\mathbf{d}_{\text{gDLS}}^\top} \text{vec}(R) + \underbrace{\sum_{i=1}^n \mathbf{k}_i^\top \mathbf{k}_i}_{k_{\text{gDLS}}} \\
&= \text{vec}(R)^\top M_{\text{gDLS}} \text{vec}(R) + 2\mathbf{d}_{\text{gDLS}}^\top \text{vec}(R) + k_{\text{gDLS}} \quad (\text{xxv})
\end{aligned}$$

$$\begin{aligned}
J'_s &= \lambda_s (s_0 - s(R))^2 \\
&= \lambda_s (SL(\mathbf{b})\text{vec}(R) + \lambda_s s_0 H_{1,1} - s_0)^2 \\
&= \text{vec}(R)^\top \underbrace{(\lambda_s L(\mathbf{b})^\top S^\top SL(\mathbf{b}))}_{M_s} \text{vec}(R) + \\
&\quad 2 \underbrace{\lambda_s (s_0 - \lambda_s s_0 H_{1,1})}_{\mathbf{d}_s^\top} SL(\mathbf{b}) \text{vec}(R) + \text{and} \quad (\text{xxvi}) \\
&\quad \underbrace{\lambda_s (\lambda_s s_0 H_{1,1} - s_0)^2}_{k_s} \\
&= \text{vec}(R)^\top M_s \text{vec}(R) + 2\mathbf{d}_s^\top \text{vec}(R) + k_s
\end{aligned}$$

$$\begin{aligned}
J'_g &= \lambda_g \|\mathbf{g}_Q \times R\mathbf{g}_W\|^2 \\
&= \text{vec}(R)^\top \underbrace{(\lambda_g L(\mathbf{g}_W)^\top [\mathbf{g}_Q]_\times [\mathbf{g}_Q]_\times L(\mathbf{g}_W))}_{M_g} \text{vec}(R) \\
&= \text{vec}(R)^\top M_g \text{vec}(R) \quad (\text{xxvii})
\end{aligned}$$

The symbol $[\cdot]_\times$ indicates the skew symmetric matrix. By putting together the components of the cost, we end up with the final cost function

$$J' = J'_{\text{gDLS}} + J'_s + J'_g, \quad (\text{xxviii})$$

which is Eq. (10) in the main submission.

References

- [1] L. Kneip, H. Li, and Y. Seo. Upnp: An optimal o(n) solution to the absolute pose problem with universal applicability. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2014. 3
- [2] C. Sweeney, V. Fragoso, T. Höllerer, and M. Turk. gDLS: A scalable solution to the generalized pose and scale problem. In *Proc. of the European Conf. on Computer Vision (ECCV)*, 2014. 1, 2, 3