Supplementary Material: 3D Edge Sketch from Multiview Images

Yilin Zheng[1] Chiang-Heng Chien[1] Ricardo Fabbri[2] Benjamin Kimia[1]

[1]Brown University [2]Rio de Janeiro State University

1. Proof of Proposition 1

Proposition 1. *Let an edge $e_1 = (\gamma_1,\ \theta_1)$ in image 1 correspond to an edge $e_2 = (\gamma_2,\ \theta_2)$ in image 2 where the relative pose of the camera is $(\mathcal{R}_{21},\ \mathcal{T}_{21})$. Then the corresponding edge in a third view $e_3 = (\gamma_3,\ \theta_3)$ with relative pose to camera 1 $(\mathcal{R}_{31},\ \mathcal{T}_{31})$ is given by*

$$\gamma_3 = \frac{\begin{array}{r}\left[\left(b_3^T T_{21}\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right) - \left(b_3^T \mathcal{R}_{21}^T T_{21}\right)\right]\mathcal{R}_{31}\gamma_1 \\ + \left[1 - \left(b_3^T \mathcal{R}_{21}\gamma_1\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right)\right] T_{31}\end{array}}{\begin{array}{r}\left[\left(b_3^T T_{21}\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right) - \left(b_3^T \mathcal{R}_{21}^T T_{21}\right)\right]\left(b_3^T \mathcal{R}_{31}\gamma_1\right) \\ + \left[1 - \left(b_3^T \mathcal{R}_{21}\gamma_1\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right)\right]\left(b_3^T T_{31}\right)\end{array}}. \tag{1}$$

$$t_3 = \frac{\begin{array}{r}\mathcal{R}_{31}\left[\left(\gamma_1 \times t_1\right) \times \mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right)\right] \\ - b_3^T\left[\left(\gamma_1 \times t_1\right) \times \mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right)\right]\gamma_3\end{array}}{\begin{array}{r}\left\|\mathcal{R}_{31}\left[\left(\gamma_1 \times t_1\right) \times \mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right)\right]\right. \\ \left. - b_3^T\left[\left(\gamma_1 \times t_1\right) \times \mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right)\right]\gamma_3\right\|\end{array}}, \tag{2}$$

where $b_3 = [0, 0, 1]^T$ and $t_i = [\cos(\theta_i),\ \sin(\theta_i),\ 0]^T$.

Proof. Consider a triplet pair of correspondences (γ_1, γ_2, γ_3) in the first, second, and third cameras, respectively, related by $(\mathcal{R}_{21}, T_{21})$ and $(\mathcal{R}_{31}, T_{31})$, *i.e.*,

$$\begin{cases}\rho_2\gamma_2 = \rho_1\mathcal{R}_{21}\gamma_1 + T_{21}, & (3)\\ \rho_3\gamma_3 = \rho_1\mathcal{R}_{31}\gamma_1 + T_{31}, & (4)\end{cases}$$

where ρ_1, ρ_2, and ρ_3 are the unknown depths at γ_1, γ_2, and γ_3, respectively. Isolating out ρ_1 and ρ_2 by dot product with b_3 on both sides of Equation 3, *i.e.*,

$$\begin{cases}\rho_2 = \rho_1\left(b_3^T\mathcal{R}_{21}\gamma_1\right) + \left(b_3^T T_{21}\right), & (5)\\ \rho_1 = \rho_2\left(b_3^T\mathcal{R}_{21}^T\gamma_2\right) - \left(b_3^T\mathcal{R}_{21}^T T_{21}\right), & (6)\end{cases}$$

can then solve for depths ρ_1 and ρ_2 in terms of $\mathcal{R}_{21}$, T_{21}, γ_1, and γ_2.

$$\begin{cases}\rho_1 = \dfrac{\left(b_3^T T_{21}\right)\left(b_3^T\mathcal{R}_{21}^T\gamma_2\right) - \left(b_3^T\mathcal{R}_{21}^T T_{21}\right)}{1 - \left(b_3^T\mathcal{R}_{21}\gamma_1\right)\left(b_3^T\mathcal{R}_{21}^T\gamma_2\right)}, \\ \rho_2 = \dfrac{-\left(b_3^T\mathcal{R}_{21}^T T_{21}\right)\left(b_3^T\mathcal{R}_{21}\gamma_1\right) + b_3^T T_{21}}{1 - \left(b_3^T\mathcal{R}_{21}\gamma_1\right)\left(b_3^T\mathcal{R}_{21}^T\gamma_2\right)}.\end{cases} \tag{7}$$

Similarly, the depth ρ_3 can be isolated from Equation (4) as

$$\rho_3 = \rho_1\left(b_3^T\mathcal{R}_{31}\gamma_1\right) + \left(b_3^T T_{31}\right), \tag{8}$$

which gives

$$\gamma_3 = \frac{\rho_1\mathcal{R}_{31}\gamma_1 + T_{31}}{\rho_1\left(b_3^T\mathcal{R}_{31}\gamma_1\right) + \left(b_3^T T_{31}\right)}. \tag{9}$$

Plugging ρ_1 obtained from Equation (7), resulting in an expression for γ_3 in terms of γ_1, γ_2, and relative poses.

$$\gamma_3 = \frac{\begin{array}{r}\left[\left(b_3^T T_{21}\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right) - \left(b_3^T \mathcal{R}_{21}^T T_{21}\right)\right]\mathcal{R}_{31}\gamma_1 \\ + \left[1 - \left(b_3^T \mathcal{R}_{21}\gamma_1\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right)\right] T_{31}\end{array}}{\begin{array}{r}\left[\left(b_3^T T_{21}\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right) - \left(b_3^T \mathcal{R}_{21}^T T_{21}\right)\right]\left(b_3^T \mathcal{R}_{31}\gamma_1\right) \\ + \left[1 - \left(b_3^T \mathcal{R}_{21}\gamma_1\right)\left(b_3^T \mathcal{R}_{21}^T \gamma_2\right)\right]\left(b_3^T T_{31}\right)\end{array}}. \tag{10}$$

Let t_1, t_2, t_3 be the unit tangent vectors at γ_1, γ_2, and γ_3, respectively. In the first camera coordinate, γ_1 and t_1 are on the same plane whose normal vector is $n_1 = \gamma_1 \times t_1$ and similarly, $n_2 = t_2 \times \gamma_2$ is the normal vector of the plane spanned by γ_2 and t_2 in the second camera coordinate. They can be related by first expressing the two tips of n_1 and n_2 as $(\Gamma_1, \Gamma_1 + n_1)$ and $(\Gamma_2, \Gamma_2 + n_2)$, respectively, so that

$$\begin{cases}\Gamma_2 = \mathcal{R}_{21}\Gamma_1 + \mathcal{T}_{21}, & (11)\\ \Gamma_2 + n_2 = \mathcal{R}_{21}\left(\Gamma_1 + n_1\right) + \mathcal{T}_{21}. & (12)\end{cases}$$

Subtracting Equation 11 by Equation 12 gives the transformation of n_1 in the first camera coordinate to n_2 in the second camera coordinate,

$$n_2 = \mathcal{R}_{21}n_1, \tag{13}$$

or,

$$\gamma_1 \times t_1 = \mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right). \tag{14}$$

Observe that the 3D tangent T_1 in the first camera coordinate arising from (γ_1, t_1) and (γ_2, t_2) is effectively the intersection of the two planes, Figure 1, *i.e.*, T_1 is orthogonal to both $\gamma_1 \times t_1$ and $\mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right)$, we have

$$\epsilon T_1 = \frac{\left(\gamma_1 \times t_1\right) \times \mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right)}{\left\|\left(\gamma_1 \times t_1\right) \times \mathcal{R}_{21}^T\left(t_2 \times \gamma_2\right)\right\|}, \quad \epsilon = \pm 1. \tag{15}$$

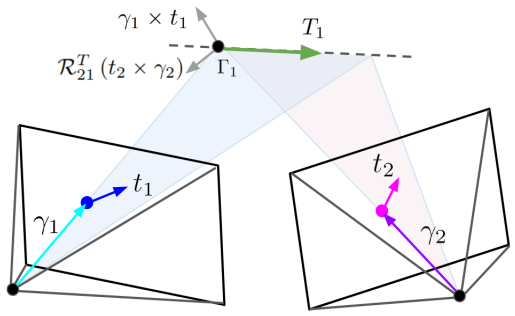

Figure 1. The 3D tangent T_1 in the first camera coordinate lies in the intersection of the two planes spanned by (γ_1, t_1) and (γ_2, t_2), respectively. T_1 is thus orthogonal to the normal vectors $\gamma_1 \times t_1$ and $\mathcal{R}_{21}^T(t_2 \times \gamma_2)$.

From Equation 13, the unit vector T_1 in the first camera coordinate can be transformed to T_3 in the third camera coordinate via $\mathcal{R}_{31}$, *i.e.*,

$$\epsilon T_3 = \frac{\mathcal{R}_{31}\epsilon T_1}{\|\mathcal{R}_{31}\epsilon T_1\|} = \frac{\mathcal{R}_{31}\left[(\gamma_1 \times t_1) \times \mathcal{R}_{21}^T(t_2 \times \gamma_2)\right]}{\|\mathcal{R}_{31}\left[(\gamma_1 \times t_1) \times \mathcal{R}_{21}^T(t_2 \times \gamma_2)\right]\|}. \tag{16}$$

In the third camera coordinate, let $(\Gamma_3, \Gamma_3 + \epsilon T_3)$ be the two tips of the vector ϵT_3 so that the projection of ϵT_3 is

$$\begin{aligned}
\bar{t}_3 &= \frac{\Gamma + \epsilon T_3}{b_3^T(\Gamma_3 + \epsilon T_3)} - \frac{\Gamma_3}{b_3^T\Gamma_3} \\
&= \frac{\left(b_3^T\Gamma_3\right)(\Gamma_3 + \epsilon T_3) - \left(b_3^T\Gamma_3\right)\Gamma_3 - \left(b_3^T\epsilon T_3\right)\Gamma_3}{\left[b_3^T(\Gamma_3 + \epsilon T_3)\right]\left(b_3^T\Gamma_3\right)} \\
&= \frac{\left(b_3^T\Gamma_3\right)\epsilon T_3 - \left(b_3^T\epsilon T_3\right)\Gamma_3}{\left[b_3^T(\Gamma_3 + \epsilon T_3)\right]\left(b_3^T\Gamma_3\right)} \\
&= \frac{\epsilon T_3 - \left(b_3^T\epsilon T_3\right)\gamma_3}{b_3^T(\Gamma_3 + \epsilon T_3)}.
\end{aligned} \tag{17}$$

Normalizing $\bar{t}_3$ gives

$$t_3 = \frac{\epsilon T_3 - \left(b_3^T\epsilon T_3\right)\gamma_3}{\|\epsilon T_3 - \left(b_3^T\epsilon T_3\right)\gamma_3\|}. \tag{18}$$

Plugging ϵT_3 from Equation 16 to Equation 18 solves the expression for t_3. ■

The above proof gives the following two corollaries.

Corollary 1. *Given a 3D unit tangent vector T_1 starts from point Γ_1 in the first camera frame, and two cameras whose poses are related by $(\mathcal{R}_{21}, T_{21})$ such that the 3D point Γ_1 in the second camera is $\Gamma_2 = \mathcal{R}_{21}\Gamma_1 + T_{21}$, then the 3D tangent vector T_1 in the second camera is $T_2 = \mathcal{R}_{21}T_1$.*

Corollary 2. *Given a 3D unit tangent vector T associated to Γ which arises from γ on the image, the projected unit tangent vector t is*

$$t = \frac{T - (e_3^T T)\gamma}{||T - (e_3^T T)\gamma||} \tag{19}$$

2. Mapping Epipolar Angles in Two Views

Let Γ be a 3D point in space arising from a pair of point correspondences $\hat{\gamma}_1$ and $\hat{\gamma}_2$ in pixels on two images, Figure 2, where the corresponding epipoles q_1 and q_2 (shown in green) have epipole angles θ_1 and θ_2, respectively. The locations of the point $\hat{\gamma}_1$ and $\hat{\gamma}_2$ can be expressed in terms of the epipole and the angle, *i.e.*,

$$\begin{cases} \hat{\gamma}_1 = q_1 + \tau_1 \begin{bmatrix} \cos(\theta_1) \\ \sin(\theta_1) \\ 0 \end{bmatrix} \\ \hat{\gamma}_2 = q_2 + \tau_2 \begin{bmatrix} \cos(\theta_2) \\ \sin(\theta_2) \\ 0 \end{bmatrix}, \end{cases} \tag{20}$$

where τ_1 and τ_2 are the distances between the epipoles and the points. Let F be the fundamental matrix describing the relative pose of the two cameras, so that the epipolar constraint

$$\hat{\gamma}_2^T F \hat{\gamma}_1 = 0, \tag{21}$$

relates the two epipole angles,

$$\begin{bmatrix} \cos(\theta_2) & \sin(\theta_2) & 0 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} \cos(\theta_1) \\ \sin(\theta_1) \\ 0 \end{bmatrix} = 0, \tag{22}$$

where

$$F = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix}. \tag{23}$$

Expanding the equation, we get

$$(f_{11} + f_{21}\tan(\theta_2)) + (f_{12} + f_{22}\tan(\theta_2))\tan(\theta_1) = 0, \tag{24}$$

so that

$$\tan(\theta_2) = -\frac{f_{11} + f_{12}\tan(\theta_1)}{f_{21} + f_{22}\tan(\theta_1)}. \tag{25}$$

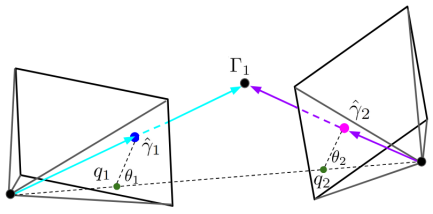

Figure 2. The epipole angles θ_1 and θ_2 can be related by the fundamental matrix describing the relative pose of the two views.

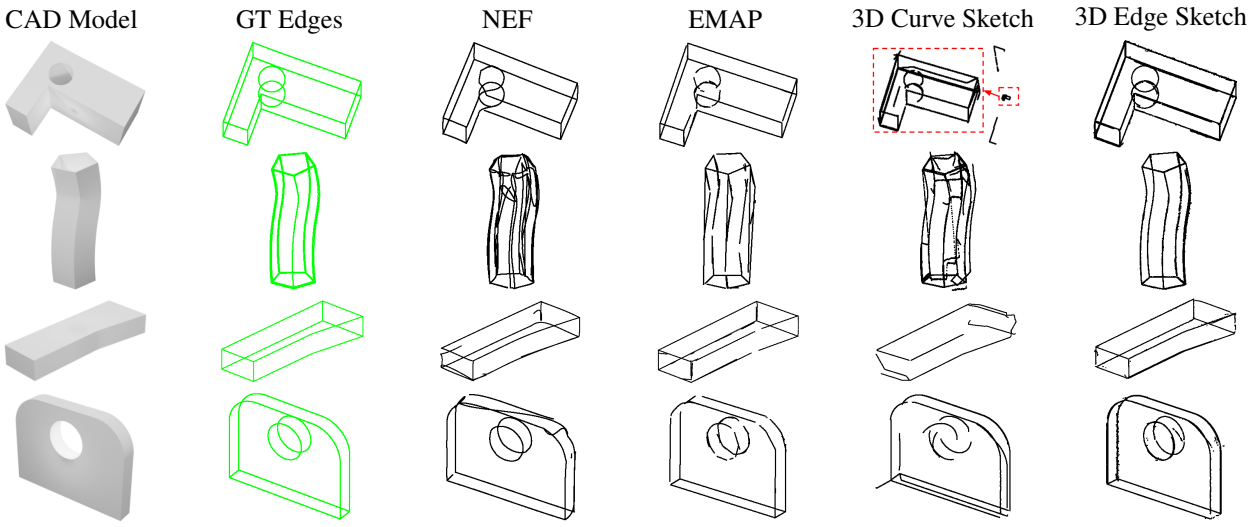

Figure 3. Qualitative comparisons between 3D curve sketch [1], NEF [4], EMAP [3], and 3D edge sketch on the ABC-NEF dataset [2]. 3D edge sketch provides comparable performance against other existing methods (**Top Row**), and outperforms others in some challenging cases (**Bottom Row**).

3. Additional Qualitative Comparisons on the ABC-NEF Dataset

Figure 3 shows additional qualitative comparisons of 3D Edge Sketch against the baseline approaches. Observe that NEF, EMAP, and 3D Curve Sketch exhibit gaps in their 3D reconstruction while 3D Edge Sketch provides a complete, high recall 3D edges.

4. Precision-Recall Curves of 3D Curve Sketch vs 3D Edge Sketch

The precision and recall curves of the proposed 3D edge sketch presented in the main paper show that *(i)* 3D edge sketch is insensitive to the hyperparameters, and *(ii)* the settings $\Delta = 0.3$ pixels, $\Delta\theta = 15^{\circ}$, and $N = 4$ views, result in optimal performance and are used in all the experiments. Compared to 3D edge sketch, varying the hyperparameters of 3D curve sketch [1] on the ABC-NEF dataset [4], Figure 4 where the number of validation views supporting a hypothesis curve pair is $N = 4$ views, reveals that 3D curve sketch is also not sensitive to hyperparameters. In addition, the precision is generally high, but its recall is low compared to 3D edge sketch, showing that a lot of 3D curve points are missing in the reconstruction, creating gaps which on the other hand 3D edge sketch does not have.

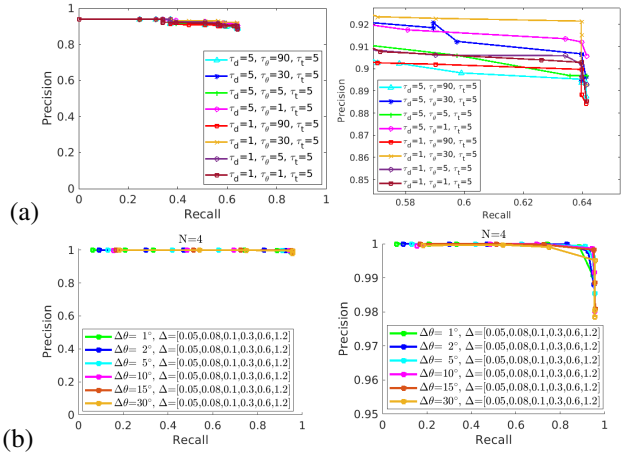

Figure 4. The precision-recall curve (**Left**) for varying the hyperparameters of (a) 3D curve sketch and (b) 3D edge sketch on the ABC-NEF dataset and their zoomed-in version (**Right**). The number of validation views supporting a hypothesis pair is $N = 4$ views for both 3D curve sketch and 3D edge sketch. It shows that the proposed 3D edge sketch has very high recall while 3D curve sketch typically generates missing reconstruction as reflected by its low recall rate. Threshold τ for measuring precision (P_τ) and recall (R_τ) is 0.02.

References

[1] Ricardo Fabbri and Benjamin Kimia. 3d curve sketch: Flexible curve-based stereo reconstruction and calibration. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1538–1545. IEEE, 2010. 3

[2] Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. ABC: A big cad model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9601–9611, 2019. 3

[3] Lei Li, Songyou Peng, Zehao Yu, Shaohui Liu, Rémi Pautrat, Xiaochuan Yin, and Marc Pollefeys. 3D neural edge reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21219–21229, 2024. 3

[4] Yunfan Ye, Renjiao Yi, Zhirui Gao, Chenyang Zhu, Zhiping Cai, and Kai Xu. NEF: Neural edge fields for 3d parametric curve reconstruction from multi-view images. In *Proceedings*

of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8486–8495, 2023. 3