Supplementary Material

Zhe Min, Zachary M. C. Baum, Shaheer U. Saeed, Mark Emberton, Dean C. Barratt, Zeike A. Taylor, and Yipeng Hu

[1] Shandong University, Jinan, China
[2] University College London, London, UK
[3] University of Leeds, Leeds, UK

In this appendix, we present more details about equations in the main paper in Sect. 1, and implementation details of the algorithms in Sect. 2.

1 Details of Equations

Chamfer Loss The Chamfer loss between the transformed source point set $\mathsf{T}(\mathbf{P}_{\mathcal{S}})$ (or only surface points $\mathbf{P}_{\mathcal{S}}^{\text{surface}}$) with $\mathbf{p}_s \in \mathbb{R}^3$ and the target point set $\mathbf{P}_{\mathcal{T}}$ (or only surface points $\mathbf{P}_{\mathcal{T}}^{\text{surface}}$) with $\mathbf{p}_t \in \mathbb{R}^3$ is given by

$$\phi(\mathsf{T}(\mathbf{P}_{\mathcal{S}}), \mathbf{P}_{\mathcal{T}}) = \frac{1}{|\widetilde{N}_t|}\Big(\sum_{t\in\widetilde{N}_t} \min_{s\in\widetilde{N}_s} ||\mathsf{T}(\mathbf{p}_s) - \mathbf{p}_t||_2^2\Big) + \frac{1}{|\widetilde{N}_s|}\Big(\sum_{s\in\widetilde{N}_s} \min_{t\in\widetilde{N}_t} ||\mathsf{T}(\mathbf{p}_s) - \mathbf{p}_t||_2^2\Big) \tag{1}$$

where $\widetilde{N}_s \subseteq \{1, ..., N_s\}$ and $\widetilde{N}_t \subseteq \{1, ..., N_t\}$ denote sets of utilised points, $|\widetilde{N}_s|$ (N_s or N_s^{surface}) and $|\widetilde{N}_t|$ (N_t or N_t^{surface}) are numbers of utilised points.

Nonlinear Strain-displacement Equations The explicit forms of nonlinear strain-displacement equations are $\varepsilon_{ii}^s = \frac{\partial d_s^i}{\partial i} + \frac{1}{2}[(\frac{\partial d_s^x}{\partial i})^2 + (\frac{\partial d_s^y}{\partial i})^2 + (\frac{\partial d_s^z}{\partial i})^2]$ where $i \in \{x, y, z\}$, $\varepsilon_{ij}^s = \frac{1}{2}(\frac{\partial d_s^i}{\partial j} + \frac{\partial d_s^j}{\partial i}) + \frac{1}{2}\left[\frac{\partial d_s^x}{\partial i}\frac{\partial d_s^x}{\partial j} + \frac{\partial d_s^y}{\partial i}\frac{\partial d_s^y}{\partial j} + \frac{\partial d_s^z}{\partial i}\frac{\partial d_s^z}{\partial j}\right]$ where the pair of dual indexes $\langle i, j\rangle \in \{\langle x, y\rangle, \langle x, z\rangle, \langle y, z\rangle\}$.

Deformation Gradients In the nonlinear constitutive and elastic energy equations, the deformation gradient $\mathbf{F}_s \in \mathbb{R}^{3\times3}$ at $\mathbf{p}_s$ is $(\mathbf{F}_s)_{ij} = \frac{\partial(\mathbf{p}_s+\mathbf{d}_s)_i}{\partial(\mathbf{p}_s)_j} = \delta_{ij} + \frac{\partial d_s^i}{\partial p_s^j}$ where $\mathbf{d}_s = [d_s^x, d_s^y, d_s^z]^\mathsf{T}$, $\langle i, j\rangle \in \{\langle x, x\rangle, \langle y, y\rangle, \langle z, z\rangle, \langle x, y\rangle, \langle x, z\rangle, \langle y, z\rangle\}$, $\delta_{ij} \in \{0, 1\}$ is the Kronecker delta with $\delta_{ij} = 1$ for $i \neq j$ and $\delta_{ij} = 0$ for $i = j$. $\mathbf{F}_s$ can be expanded more explicitly as

$$\mathbf{F}_s = \begin{bmatrix} \frac{\partial(x+d_s^x)}{\partial x} & \frac{\partial(x+d_s^x)}{\partial y} & \frac{\partial(x+d_s^x)}{\partial z} \\ \frac{\partial(y+d_s^y)}{\partial x} & \frac{\partial(y+d_s^y)}{\partial y} & \frac{\partial(y+d_s^y)}{\partial z} \\ \frac{\partial(z+d_s^z)}{\partial x} & \frac{\partial(z+d_s^z)}{\partial y} & \frac{\partial(z+d_s^z)}{\partial z} \end{bmatrix} = \begin{bmatrix} 1+\frac{\partial d_s^x}{\partial x} & \frac{\partial d_s^x}{\partial y} & \frac{\partial d_s^x}{\partial z} \\ \frac{\partial d_s^y}{\partial x} & 1+\frac{\partial d_s^y}{\partial y} & \frac{\partial d_s^y}{\partial z} \\ \frac{\partial d_s^z}{\partial x} & \frac{\partial d_s^z}{\partial y} & 1+\frac{\partial d_s^z}{\partial z} \end{bmatrix}. \tag{2}$$

Left Cauchy–Green Deformation Tensor Given $\mathbf{F}_s$ in Eq. (2), with some further manipulations, the symmetric left Cauchy–Green deformation tensor $\mathbf{F}_s\mathbf{F}_s^\mathsf{T} \in \mathbb{S}^{3\times3}$ used in the nonlinear constitutive equation is

$$(\mathbf{F}_s\mathbf{F}_s^\mathsf{T})_{ii} = 1 + 2\frac{\partial d_s^i}{\partial i} + \sum_{j\in\{x,y,z\}} (\frac{\partial d_s^i}{\partial j})^2, \quad i \in \{x, y, z\}, \tag{3}$$

$$(\mathbf{F}_s\mathbf{F}_s^\mathsf{T})_{ij} = \frac{\partial d_s^i}{\partial j} + \frac{\partial d_s^j}{\partial i} + \sum_{k\in\{x,y,z\}} \frac{\partial d_s^i}{\partial k}\frac{\partial d_s^j}{\partial k}, \quad \langle i,j\rangle \in \{\langle\langle x,y\rangle, \langle x,z\rangle, \langle y,z\rangle\}, \tag{4}$$

where for brevity $(\mathbf{F}_s\mathbf{F}_s^\mathsf{T})_{11} = (\mathbf{F}_s\mathbf{F}_s^\mathsf{T})_{xx}$ (similar for index pairs $\langle 2,2\rangle$ and $\langle 3,3\rangle$) and $(\mathbf{F}_s\mathbf{F}_s^\mathsf{T})_{12} = (\mathbf{F}_s\mathbf{F}_s^\mathsf{T})_{xy}$ (similar for index pairs $\langle 1,3\rangle$ and $\langle 2,3\rangle$) hold.

2 Implementation Details

First, the 3-by-3 TNet in the original PointNet is adjusted with a TNet 4-by-4 that outputs 4×4 rigid transformation matrix, due to better performance of recovering pose differences between two point sets as suggested in [1]. The TNet consists of shared MLPs with layers' output sizes being 64, 128, 1024, a max pooling layer, two fully connected layers with output sizes being 512, 256, and a linear layer with output being 4×4. The rigidly aligned point sets will then go through shared MLP(64, 64). Afterwards, the learnt features will go through the other TNet 64×64, shared MLP(64, 128, 1024), a max-pooling layer. The final global feature $\phi(\cdot)$ learnt from one point set using PointNet is of size 1024. At the end of the global feature extraction (GFE) module, the global features $\phi(\mathbf{P}_\mathcal{S})$ and $\phi(\mathbf{P}_\mathcal{T})$ learnt from $\mathbf{P}_\mathcal{S}$ and $\mathbf{P}_\mathcal{T}$ are concatenated to form the feature of size 2048. In the point transformation (PT) module, the concatenated global feature is repeated for N_s times and further concatenated with $\mathbf{P}_\mathcal{S}$. The resulting feature map of size $N_s\times 2051$ will go through shared MLP(1024, 512, 256, 128, 64) and another shared MLP(256) without the ReLU layer. At the end, MLP(3) and 6 MLP(1)(for Cfg1, and it is MLP(6) for Cfg2 as depicted in Fig.1 in the paper) are used in branches $g_{\theta_g}(\mathcal{D}_k)$ predicting displacements $\mathbf{D}_\mathcal{S}\in\mathbb{R}^{N_s\times 3}$ and $h_{\theta_h}(\mathcal{D}_k)$ predicting stress $\boldsymbol{\sigma}$, respectively. In Cfg1, both branches possess their individual GFE and PT modules while one common GFE module is shared between two branches in Cfg2.

In the forward problem, in the nonlinear constitutive and elastic energy equations, lame parameters $\lambda_s\in\mathbb{R}$ and $\mu_s\in\mathbb{R}$ at $\mathbf{p}_s$ need to be known. They are computed with $\lambda_s = \frac{E_s\nu_s}{(1-2\nu_s)(1+\nu_s)}$ and $\mu_s = \frac{E_s}{2(1+\nu_s)}$, where Young's Modulus $E_s\in\mathbb{R}$ and Possion's ratio $v_s\in\mathbb{R}$ were set as those values within two subregions (i.e., the peripheral and transition zones) used in the finite element generation process [2]. In objective functions of both forward and inverse problems (i.e., Eq. (1) and Eq. (2) in the main paper), the empirical weight $w=10^5$ with the Chamfer loss was used. All experiments were run for 20000 epochs on an Intel(R) Xeon(R) Gold 5215 CPU with an NVIDIA Quadro GV100 32GB GPU.

References

1. Baum, Z.M., Hu, Y., Barratt, D.C.: Real-time multimodal image registration with partial intraoperative point-set data. Medical image analysis **74**, 102231 (2021)
2. Hu, Y., Carter, T.J., Ahmed, H.U., Emberton, M., Allen, C., Hawkes, D.J., Barratt, D.C.: Modelling prostate motion for data fusion during image-guided interventions. IEEE transactions on medical imaging **30**(11), 1887–1900 (2011)