# Supplementary Material: Biomechanics-guided Facial Action Unit Detection through Force Modeling

## A. Motion Law in the Generalized Coordinate

Given the Euler-Lagrangian equation in general, we leverage the construction of 3D facial mesh as a linear combination of AU specific blendshapes, i.e.,

$$\boldsymbol{u} = c_1 B_1 + c_2 B_2 + \cdots + c_K B_K \tag{1}$$

where $\{B_j\}_{j=1}^{K}$ are known and $K$ is the total number of bases defined by the reconstruction model. Both $\boldsymbol{u}$ and $B_j$ are of $N \times 3$ dimension. $\{c_j\}_{j=1}^{K}$ are the coefficients specific to each reconstructed mesh $\boldsymbol{u}$. Leveraging the known blendshapes as our bases, we define the motion via coefficients $c_j$ as our generalized coordinate.

Following Lagrangian equations, we derive our motion law for the whole mesh in the generalized coordinate. Given the $u_i \in R^{3\times 1}$ being the position for $i$-th vertex in the Cartesian coordinate, we have $\boldsymbol{u}_i = \boldsymbol{u}_i(c_1, c_2, ..., c_K)$. Virtual displacement then becomes

$$\begin{aligned} \delta \boldsymbol{u}_i &= \sum_{j=1}^{K} B_j(i)\delta c_j \\ &= \begin{bmatrix} B_1(i) & B_2(i) & ... & B_K(i) \end{bmatrix} \begin{bmatrix} \delta c_1 \\ \delta c_2 \\ ... \\ \delta c_K \end{bmatrix} \end{aligned} \tag{2}$$

where $\delta c_j = c_j(t + \Delta t) - c_j(t)$ and $B_j(i) \in R^{3\times 1}$ is the position for $i$-th vertex in $j$-th base. Given the virtual displacement, we obtain the velocity in the generalized coordinate as

$$\begin{aligned} \dot{\boldsymbol{u}}_i &= \sum_{j=1}^{K} B_j(i)\dot{c}_j \\ &= \begin{bmatrix} B_1(i) & B_2(i) & ... & B_K(i) \end{bmatrix} \begin{bmatrix} \dot{c}_1 \\ \dot{c}_2 \\ ... \\ \dot{c}_K \end{bmatrix} \\ &= \boldsymbol{B}(i)\dot{\boldsymbol{c}} \end{aligned} \tag{3}$$

where $\dot{c}_j = \frac{\delta c_j}{\Delta t}$ and $\boldsymbol{B}_j = [B_1(i), B_2(i), ..., B_K(i)] \in R^{3\times K}$. The kinetic energy for $i$-th vertex then can be computed as

$$\begin{aligned} T_i &= \frac{1}{2} m_i \dot{\boldsymbol{u}}_i^T \dot{\boldsymbol{u}}_i = \frac{1}{2} m_i (\boldsymbol{B}(i)\dot{\boldsymbol{c}})^T (\boldsymbol{B}(i)\dot{\boldsymbol{c}}) \\ &= \frac{1}{2}\dot{\boldsymbol{c}}^T (m_i \boldsymbol{B}(i)^T \boldsymbol{B}(i))\dot{\boldsymbol{c}} = \frac{1}{2}\dot{\boldsymbol{c}}^T m_i^B \dot{\boldsymbol{c}} \end{aligned} \tag{4}$$

where

$$\begin{aligned} m_i^B &= m_i \boldsymbol{B}(i)^T \boldsymbol{B}(i) \\ &= m_i \begin{bmatrix} B_1(i) \\ B_2(i) \\ ... \\ B_K(i) \end{bmatrix} \begin{bmatrix} B_1(i) & B_2(i) & ... & B_K(i) \end{bmatrix} \\ &= m_i \hat{\boldsymbol{B}}(i) \end{aligned} \tag{5}$$

and $\hat{\boldsymbol{B}}(i) \in R^{K\times K}$. By computing the virtual work, we can define the generalized forces as

$$\boldsymbol{f}_i \cdot \delta \boldsymbol{u}_i = \boldsymbol{f}_i \cdot \sum_{j=1}^{K} B_j(i)\delta c_j \equiv \sum_{j=1}^{K} Q_{ij}\delta c_j = \boldsymbol{Q}_j \cdot \delta \boldsymbol{c} \tag{6}$$

$Q_i \in R^{K\times 1}$ is the generalized force applied to $i$-th vertex, and $Q_{ij} = B_j^T(i)\boldsymbol{f}_i$ is the generalized force along $j$-th dimension in the generalized coordinate. Given the Lagrangian equation for each vertex

$$\frac{d}{dt}\left(\frac{\partial T_i}{\partial \dot{\boldsymbol{c}}}\right) - \frac{\partial T_i}{\partial \dot{\boldsymbol{c}}} = Q_i \tag{7}$$

we plug the derived kinetic energy into the equation, and the left hand side of the equation becomes

$$\begin{aligned} \frac{d}{dt}(m_i^B \dot{\boldsymbol{c}}) &= m_i^B \ddot{\boldsymbol{c}} + \frac{dm_i^B}{dt}\dot{\boldsymbol{c}} - \frac{1}{2}\dot{\boldsymbol{c}}^T \left(\frac{\partial m_i^B}{\partial c}\right)^T \dot{\boldsymbol{c}} \\ &= m_i^B \ddot{\boldsymbol{c}} + C(\boldsymbol{c}, \dot{\boldsymbol{c}}) \end{aligned} \tag{8}$$

Since $m_i^B$ is not a function of time $t$, nor a function of $\boldsymbol{c}$, $C(\boldsymbol{c}, \dot{\boldsymbol{c}}) = 0$. In the end, consider the whole mesh, and we have the Lagrangian equation

$$\frac{d}{dt}\left(\frac{\partial \sum_{i=1}^{N} T_i}{\partial \dot{\boldsymbol{c}}}\right) - \frac{\partial \sum_{i=1}^{N} T_i}{\partial \dot{\boldsymbol{c}}} = \sum_{i=1}^{N} Q_i \tag{9}$$

The motion equation for the whole mesh in the generalized coordinate is

$$M^B \ddot{\boldsymbol{c}} = \boldsymbol{Q} \tag{10}$$

$M^B = \frac{1}{2}\sum_i m_i^B = \frac{1}{2}\sum_i m_i \hat{B}(i)$. $\hat{B}(i) = B(i)^T B(i)$ and $B(i) = [B_1(i), B_2(i), ..., B_j(i), ..., B_J(i)] \in R^{3\times K}$. $Q$ is the generalized force with $Q_{ij} = B_j^T(i)\boldsymbol{f}_i$. Particularly, the force $\boldsymbol{f}_i$ only contains external force due to muscle contraction.

## B. Verification of the Dynamic Law

We verify the derived dynamic law in the generalized coordinate with synthetic data. We use FaceGen software and extract 13 AU bases (i.e., AU1, AU2, AU4, AU5, AU7, AU9, AU12, AU15, AU16, AU20, AU23), and AU26. The mesh $\boldsymbol{u}$ contains $5,850$ vertices, and is computed following Eq. **??** with $K = 13$. We apply the synthetic muscle activation force $\boldsymbol{f}^{mus}$ lifting the mouse, which is corresponding to AU12 and set ground truth mass $m_i$ of each vertex as $1$. The magnitude of the applied force is visualized in Figure 1 (a). Given the muscle activation force and the masses, we can compute the generalized force $\boldsymbol{Q}$ and the generalized mass $\boldsymbol{M}^B$. Following the forward dynamic in Eq. **??**, we predict linear coefficients $\boldsymbol{c}$ over future time steps as plotted in Figure 1(b). As expected, the coefficient corresponding to AU12 is increasing over time, while other coefficients remain zeros. The deformation $\boldsymbol{u}$ over time obtained with the solved $\boldsymbol{c}$ is identical to the deformation obtained in the Cartesian coordinate.

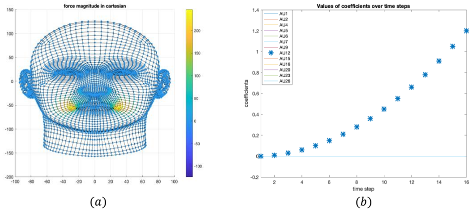


Figure 1. Verification of the dynamic law. (a) Visualization of the synthetic force; (b) Predicted linear coefficients $\boldsymbol{c}$.

## C. Experimental Settings and Results

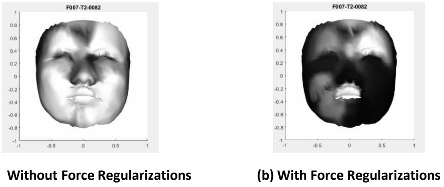
### C.1. Experimental settings

**Datasets.** The BP4D dataset is a spontaneous facial AU dataset, and consists of 328 videos from 41 subjects. with 23 female adults and 18 male adults of various ethnicity. About 140,000 frames are annotated with AU labels. DISFA is a spontaneous expression database, containing 27 videos from 27 subjects. In total, 130,815 frames are annotated with AU intensities in the range from 0 to 5.

**Implementation Details.** The reconstructed 3D mesh contains $35,709$ vertices. We down sample the mesh to a sparse one containing $2,232$ vertices. For 3D mesh convolution, the down sample factor is $[2,2,2,2,2]$ and the corresponding convolution filters are of size $[16,16,16,32,32]$. In image branch, the ResNet50 is pre-trained on ImageNet. The extracted image feature is of dimension $512$. For AU prediction, a fully connected layer is employed (Eq. 12). In the total training objective, $\lambda_{3d} = 1$, $\lambda_{au} = 0.1$, $\lambda_c = 1e{-}5$ and $\lambda_s = 1e-4$. Adam optimizer is employed for training. Learning rate is $1e-3$ with a learning rate decay of $0.85$ every epoch. The batch size is $4$ and the sequence length is $4$ for all experiments. Time interval for Euler integration method is $\Delta t = 0.04s$. We implement the proposed method with Tensorflow.

### C.2. Ablation study on force regularization

We introduce the force regularization (Eq. 17) to remove noisy forces and to enforce the estimated forces from adjacent frames similar. Without force regularization, the AU detection performance on BP4D is $62.0\%$, which is worse than the performance with force regularization terms $(64.1\%)$. In Fig. 2 below, we further visualize the magnitude of the estimated forces on BP4D. As shown, without force regularization (Fig. 2 (a)), there exists significant noisy forces as forces in most areas are non-zeros. While adding force regularization (Fig. 2 (b)), the estimated forces can accurately reflect the activated AUs (AU1, 2, 10).



(a) Without Force Regularizations

(b) With Force Regularizations

Figure 2. Force visualization (subject F007, 82nd frame of T7).