Shape normalization for keypoint data

Caleb Weinreb and Kai Fox

May 2023

1 Modeling framework

The goal of shape normalization is to map keypoint observations from many animals into a standardized pose space where the effects of body morphology have been removed. Naively, one might hope to normalize using simple summary statistics like average nose-to-tail distance. But these statistics are problematic because they conflate body shape with behavior. For example, larger average nose-to-tail distance could reflect a larger body, or simply a higher frequency of stretched-out poses. Thus the central challenge of shape normalization is to isolate the effects of behavior and body shape so that the latter can be cleanly removed. To that end, we propose a generative model that factors behavior and morphology into separate terms that are disentangled during fitting.

1.1 Input data

We consider a dataset of poses $\{y_{n,t}\} \subset \mathbb{R}^{KD}$ measured from animals $n = 1, \ldots, N$ at timepoints $t = 1, \ldots, T_n$ (the timepoints need not be sequential), where each pose contains coordinates of K keypoints in D dimensions. We assume that the keypoints are already centered and rotationally aligned, and that differences in body size have already been minimized as much as possible through uniform scaling of each animal.

1.2 Generative model

We model each pose $y_{n,t}$ as the animal-specific realization of a standardized pose $x_{n,t}$ that is sampled from a Gaussian mixture. The mixture components are shared across animals, but the mixture weights are animal-specific. Formally, the model is

$$
\begin{aligned}
y_{n,t} &= f(x_{n,t}, \phi_n) && \text{(observed pose given standardized pose)} \\
x_{n,t} &\sim \mathcal{N}(m_{z_{n,t}}, Q_{z_{n,t}}) && \text{(standardized pose given mixture component)} \\
z_{n,t} &\sim \text{Cat}(\pi_n) && \text{(mixture component given animal-specific weights)} \\
m_z &\sim \mathcal{N}(m_0, \lambda_0^{-1} Q_z) && \text{(prior on mixture means given covariances)} \\
Q_z &\sim \text{Wishart}^{-1}(W_0, \nu_0) && \text{(prior on mixture covariances)} \\
\pi_n &\sim \text{Dir}(\beta) && \text{(prior on animal-specific mixture weights)} \\
\beta &\sim \text{Dir}(\gamma, \ldots, \gamma) && \text{(prior on mixture weight hyperparameters)}
\end{aligned}
$$

where f is a morph function that maps standardized poses $x_{n,t}$ to animal-specific poses $y_{n,t}$ and ϕ_n is a parameter vector that captures the unique shape of animal n. Several forms for f (and priors on ϕ_n) are described in Section XXX and tested in the paper. In general, f should be invertible in its first argument and differentiable in its second. Abusing notation, we will use f^{-1} to denote the inverse of f in its first argument.

1.3 Expectation maximization

The model parameters $\theta = (\phi, m, Q, \pi, \beta)$ are fit using expectation maximization (EM), which aims to maximize the expected log likelihood $\ell(\theta) = \mathbb{E} \log P(y, z \mid \theta)$. EM alternates between two steps.

E-step Given current parameter estimates $\theta^* = (\phi^*, m^*, Q^*, \pi^*, \beta^*)$, calculate the posterior $q(z) = P(z \mid y, \theta^*)$, as follows.

$$q(z) = \prod_{n,t} P(z_{n,t} \mid y_{n,t}, \phi^*, m^*, Q^*, \pi^*) \tag{1}$$

$$\propto \prod_{n,t} \mathcal{N}(f^{-1}(y_{n,t}, \phi_n^*) \mid m_{z_{n,t}^*}, Q_{z_{n,t}^*}) \cdot \pi_{n,z_{n,t}}^* \tag{2}$$

M-step Given $q(z)$ from the E-step, obtain new parameter estimates by optimizing the following objective with respect to $\theta = (\phi, m, Q, \pi, \beta)$ (using gradient ascent).

$$A(\theta; \theta^*) = \sum_{n,t} \mathbb{E}_q \log P(y_{n,t}, z_{n,t} \mid \theta) + \log P(\theta) \tag{3}$$

$$= \sum_{n,t} q(z_{n,t}) \left[\log \mathcal{N}(f^{-1}(y_{n,t}, \phi_n) \mid m_{z_{n,t}}, Q_{z_{n,t}}) + \log \pi_{n,z_{n,t}} \right] + \log P(\theta) \tag{4}$$

1.4 Initialization

Initialization of the morph parameters ϕ is described in section XXX. Given morph parameters, we fit a standard Gaussian mixture model to the points $x_{n,t} = f^{-1}(y_{n,t}, \phi_n)$ (e.g. with sklearn), yielding mixture components $(m_z, Q_z), z = 1, ..., L$ and cluster weights $\lambda_{n,t,z}$ for each data point. Using the weights, we initialize π and β as follows.

$$\pi_{n,z} = \frac{1}{T_n} \sum_{t=1}^{T_n} \lambda_{n,t,z}, \quad \beta_z = \frac{1}{N} \sum_{n=1}^{N} \pi_{n,z} \tag{5}$$

2 Morph models

2.1 Low-rank affine morph

One way to enforce simplicity $f(x, \phi)$ is to make it an affine map in x. However, because $x \in \mathbb{R}^{KD}$ is rather high dimensional, this still leaves a huge number of degrees of freedom. We therefore also enforce in this morph model that the linear component of f be low-rank.

The main adjustment of posture will be performed by the affine offset. For example, this might set certain bone lengths to be longer for a particular session. It is unlikely that these updates will be sufficient across all poses however - e.g. when limbs are extended in 2D keypoints the bone lengths adjustments need to be greater, or in 3D keypoints the updated needed to extend bone lengths during a rear are different from those needed during running. We therefore also add an update to the first L PCs ("modes"), inferred from a reference session and fixed during training.

The morph function described above may be formulated in terms of the following hyperparameters:

$m \in \mathbb{R}^{KD}$	Centroid of population pose space
$U \in \mathbb{R}^{L \times KD}$	Orthonormal matrix of posture modes
$\nu_m, \nu_U \in \mathbb{R}$	Prior standard deviation for centroid and mode adjustments, respectively

and a tuple $\phi_n = (\hat{m}_n, \hat{U}_n)$ of trainable parameters, for which we now also specify priors:

$\hat{m}_n \in \mathbb{R}^{KD}$	$\sim \mathcal{N}(\mathbf{0}_{KD}, \nu_m I_{KD})$	Session-wise adjustments to centroid pose
$\hat{U}_n \in \mathbb{R}^{L \times KD}$	$\sim \mathcal{N}(\mathbf{0}_{LKD}, \nu_U I_{LKD})$	Session-wise adjustments to posture modes

We may now write down a mapping of poses for low rank affine morph. We use the projection onto orthogonal complement of U's columns as a pass-through on the dimensions of pose space beyond the first L PCs. Since U is a tall matrix with orthogonal columns, this projection takes the particularly simple form $U^\perp = I - UU^T$. Using this, we may write

$$f(x, \phi_n) = \left(U + \hat{U}_n\right) U^T (x - m) + U^\perp (x - m) + (m + \hat{m}_n) \tag{6}$$

$$= \left[I + \hat{U}_n U^T\right] (x - m) + (m + \hat{m}_n) \tag{7}$$

The inverse of this map in its first argument is simply

$$f^{-1}(x, \phi_n) = \left[I + \hat{U}_n U^T\right]^{-1} (x - (m + \hat{m}_n)) + m \tag{8}$$

3 E-step

In the E-step we seek to calculate $q_{n,t}(z) = P(z \mid y_{n,t}, \theta^*)$ for a given keypoint observation $y_{n,t}$ and estimated parameters θ^*. This is usually done using Bayes' rule

$$q_{n,t}(z) = P(z \mid y_{n,t}, \theta^*) = \frac{P(y_{n,t} \mid z, \theta^*) P(z \mid \theta^*)}{\sum_{z'} P(y_{n,t} \mid z', \theta^*) P(z' \mid \theta^*)} \tag{9}$$

Each probability above is given directly by the generative model (Sec 1.2), and so may be expanded as

$$q_{n,t}(z) = \frac{\mathcal{N}(f^{-1}(y_{n,t}, \phi_n^*) \mid m_z^*, Q_z^*) \cdot \pi_{n,z}^*}{\sum_{z'} \mathcal{N}(f^{-1}(y_{n,t}, \phi_n^*) \mid m_{z'}^*, Q_{z'}^*) \cdot \pi_{n,z'}^*} \tag{10}$$

which yields the proportionality result stated in Eq. 2 with the additional specification of normalization so that $q_{n,t}$ is a probability distribution in z.