Vol.2021-CG-181 No.3
2021/2/16

Variational Recurrent Neural Networkを用いた人物動作生成モデルの構築

村上 真[1,a)]　生澤 隆広[1]

概要：3 次元コンピュータグラフィックスを使用した映像コンテンツには人型のキャラクタが登場し，人のように行動することが多い．本研究の目的は，多様で自然なキャラクタの動作を生成することができるシステムを構築することである．本研究では，動作の生成過程は複雑で非線形だと考え，この過程を深層ニューラルネットワークによりモデル化する．動作の生成過程（深層ニューラルネットワークのパラメータ）は直接観測できないため，観測可能な動作データ（モーションキャプチャシステムにより収録された動作データ）から学習により推定することになる．その際に，生成とは逆の過程（推論過程）も同様に深層ニューラルネットワークにより表現し，観測可能な動作データから推論と生成を行い，元のデータが得られればよいという基準で深層ネットワークのパラメータを推定する．本研究では recurrent neural network と variational autoencoder を用いた動作生成モデルを構築し，低次元の潜在空間から多様な人物動作が生成可能であることを確認した．

Human Motion Generative Model using Variational Recurrent Neural Network

1. はじめに

映画やゲームといった 3 次元コンピュータグラフィックスのコンテンツには人型のキャラクタが登場し，人のように行動することが多い．そのため，キャラクタの動作を生成・制御・編集することは重要なタスクである．いくつかのキーフレームに対してキャラクタの姿勢を指定し，キーフレーム間の姿勢を補間するキーフレームアニメーションでは，自然な動作を生成することが難しい．自然な動作を生成するにはモーションキャプチャシステムで収録した動作データを使用することが多く，モーションキャプチャデータを制御・編集することで多様で自然な動作を生成する手法が数多く提案されている [1].

近年，深層ニューラルネットワークを使用し，モーションキャプチャデータから学習することで動作を制御する方法が提案された．Holden ら [2],[3] は畳込みオートエンコーダを使用して学習することで低次元の多様体として動作データを表現した．また，高次の動作制御パラメータを多様体上の動作データに写像する深層ニューラルネットワークをオートエンコーダの前段に置くことで，床の上の移動の軌跡といった高次の動作制御パラメータから動作を生成することができるシステムを構築した．

一方, 深層ニューラルネットワークを用いて様々なデータを生成することができる生成モデルが提案された．Kingma ら [4], [5] は Variational AutoEncoder(VAE) を提案し，画像生成に適用することで多様な画像が生成できることを示した．Goodfellow ら [6] は generative adversarial networks を提案し，様々な対象の画像データセットを用いて学習することで，その対象の画像を生成することができることを示した．Bowman ら [7] は再帰型ニューラルネットワーク（Recurrent Neural Network:RNN) を VAE を組み合わせた RNN-based VAE を提案し，文書データセットを用いて学習することで，低次元多様体で文を表現した．エンコーダを用いて多様体上に 2 つの文を射影し，多様体上で 2 つの文の間を補間する潜在変数を求め，デコーダによって潜在変数から文を生成した結果，意味的に 2 つの文の間を補間する文が生成できることを示した．Fabius ら [8] は variational recurrent autoencoder を提案し，MIDI 形式の音楽データを生成した．Sabathe ら [9] は LSTM-based

[1] 東洋大学
Toyo University, Kawagoe, Saitama 350–8585, Japan
[a)] murakami_m@toyo.jp

VAE を提案し，様々な音楽特徴と性質を持つ楽曲の一部を自動生成できることを示した．

本研究の目的は，RNN と VAE を使用し，動作データから学習することで，多様な動作データを生成することができるモデルを構築することである．

2. Variational Autoencoder

2.1 生成モデルと推論モデル

2.1.1 生成過程と推論過程

D 次元のデータ $\boldsymbol{x}^{(i)} \in \mathbb{R}^D$ からなるデータセット $\mathrm{X} = \left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{N}$ があるとし，各データ $\boldsymbol{x}^{(i)}$ は事前確率 $p(\boldsymbol{z})$ にしたがって生成されると考える．確率変数 $\boldsymbol{z}$ を潜在変数と呼び，低次元の潜在変数から高次元のデータが生成されると考える．また，事前確率は単純であると考え，ここでは標準正規分布 $\mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, I)$ とする．まず，単純な事前確率 $p(\boldsymbol{z})$ からある潜在変数 $\boldsymbol{z}^{(i)}$ がサンプリングされ，次に，各データ $\boldsymbol{x}^{(i)}$ が尤度と呼ばれる確率分布 $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ に従って生成されるとする．

次に，観測されたデータ $\boldsymbol{x}$ から潜在変数 $\boldsymbol{z}$ を推定する推論過程を考える．この確率分布を事後確率 $p_\theta(\boldsymbol{z}|\boldsymbol{x})$ と呼び，事後確率は

$$p_\theta(\boldsymbol{z}|\boldsymbol{x}) = \frac{p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{\int p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})d\boldsymbol{z}} \tag{1}$$

のように事前確率 $p(\boldsymbol{z})$ と尤度 $p_\theta(\boldsymbol{x}|\boldsymbol{z})$ によって表すことができる．

2.1.2 推論モデル

式 (1) の分母の全ての $\boldsymbol{z}$ に関して積分する部分が現実的な時間で計算できないため，事後確率は intractable である．そこで，事後確率の近似である確率分布 $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ を導入する．推論過程も複雑で非線形な確率過程と考えられるため，非線形写像であるニューラルネットワークと正規分布を組み合わせて表現する．この過程は観測されたデータから潜在的な意味のようなものを取り出す過程と見なせる．データをエンコードするという意味で，推論モデルで使用するニューラルネットワークをエンコーダと呼ぶ．エンコーダネットワークではデータ $\boldsymbol{x}$ を入力し，正規分布の平均 $\boldsymbol{\mu}_\phi(\boldsymbol{x})$ と標準偏差 $\boldsymbol{\sigma}_\phi(\boldsymbol{x})$ を出力する．ここで，潜在空間の各軸は互いに独立した意味を持っていると考え，共分散が 0 である対角行列 $\mathrm{diag}(\boldsymbol{\sigma_\phi} \odot \boldsymbol{\sigma_\phi})$ を共分散行列とする正規分布を使用する．$\odot$ は 2 つのベクトルを要素ごとに掛け算する演算子を表す．

2.1.3 生成モデル

低次元の潜在変数 $\boldsymbol{z}$ から高次元のデータ $\boldsymbol{x}$ を生成する過程も複雑で非線形な確率過程と考えられるため，非線形写像であるニューラルネットワークと正規分布を組み合わせて表現する．この過程はエンコードされたデータをデコードして元に戻す過程と見なせるため，生成モデルで使用するニューラルネットワークをデコーダと呼ぶ．デコーダネットワークでは潜在変数 $\boldsymbol{z}$ を入力し，正規分布の平均 $\boldsymbol{\mu}_\theta(\boldsymbol{z})$ と標準偏差 $\boldsymbol{\sigma}_\theta(\boldsymbol{z})$ を出力する．

2.2 誤差関数

2.2.1 誤差関数

生成モデルのパラメータ θ と推論モデルのパラメータ ϕ は未知であるため，観測されたデータセット X から学習により推定することになる．ここでは，事後確率 $p_\theta(\boldsymbol{z}|\boldsymbol{x})$ とその近似として導入した $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ との差異がなるべく小さくなる θ と ϕ を求める．$p_\theta(\boldsymbol{z}|\boldsymbol{x})$ に対する $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ の KL Divergence は

$$\begin{aligned} & D_{KL}\left(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p_\theta(\boldsymbol{z}|\boldsymbol{x})\right) \\ =\ & \mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{q_\phi(\boldsymbol{z}|\boldsymbol{x})p_\theta(\boldsymbol{x})}{p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}\right] \\ =\ & -\mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})\right] + D_{KL}\left(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})\right) \\ & + \log p_\theta(\boldsymbol{x}) \end{aligned} \tag{2}$$

となる．右辺第 3 項は intractable であるが，右辺第 1 項と第 2 項は tractable であるため，右辺第 1 項と第 2 項の和を誤差関数 $E(\theta, \phi; \boldsymbol{x})$ とし，誤差ががなるべく小さくなる θ と ϕ を求めることにする．

$$E(\theta, \phi; \boldsymbol{x}) = -\mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})\right] + D_{KL}\left(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})\right) \tag{3}$$

2.2.2 再現誤差と正則化

誤差関数 $-\mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})\right] + D_{KL}\left(q_\phi(\boldsymbol{z}|\boldsymbol{x})||p(\boldsymbol{z})\right)$ の第 1 項は，あるデータ $\boldsymbol{x}$ をエンコードし，潜在変数 $\boldsymbol{z}$ を求め，それをデコードしたものが元のデータと同じになる場合に小さな値となることから，再現誤差と考えることができる．

また，第 2 項は事後確率の分布 $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ が事前確率 $p(\boldsymbol{z})$ の分布と近くなった場合に小さくなる．事後確率の分布には自由度があるため，これがなるべく単純な分布となるようにする効果がある．したがって第 2 項は正則化の役割を果たしていると考えられる．

再現誤差は実際のデータとエンコード・デコードしたデータとの差であるため，データの空間における誤差となる．一方，正則化項は事後確率と事前確率との差であるため，潜在変数の空間における誤差となる．このように，データ空間と潜在空間の両方で最適化が行われることがわかる．

2.2.3 再現誤差の計算

再現誤差はデータのバッチサイズを M とすると

$$-\mathbb{E}_{\boldsymbol{z}\sim q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_\theta(\boldsymbol{x}|\boldsymbol{z})\right] = -\frac{1}{M}\sum_{i=1}^{M}\log p_\theta\left(\boldsymbol{x}^{(i)}|\boldsymbol{z}^{(i)}\right) \tag{4}$$

と求められる．

ここで，$\boldsymbol{z}^{(i)}$ は $q_\phi(\boldsymbol{z}|\boldsymbol{x}^{(i)})$ からサンプリングされた潜在

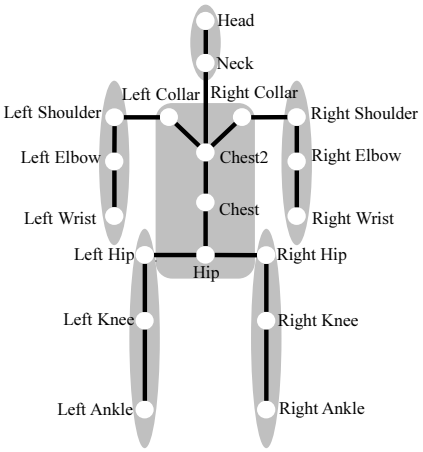

図 1 Joint structure in motion data.

変数であるが，サンプリングの操作が入ると $\boldsymbol{z}$ に関する勾配計算ができなくなるため，勾配計算ができるように変更する．今，$\boldsymbol{z}$ が平均 $\boldsymbol{\mu_\phi}$ 分散 $\boldsymbol{\sigma_\phi} \odot \boldsymbol{\sigma_\phi}$ の正規分布からサンプリングされるとすると

$$\boldsymbol{z} \sim q_\phi(\boldsymbol{z}|\boldsymbol{x}^{(i)}) = \mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu_\phi}, \mathrm{diag}(\boldsymbol{\sigma_\phi} \odot \boldsymbol{\sigma_\phi})) \tag{5}$$

となるが，一旦標準正規分布から確率変数 $\boldsymbol{\epsilon}$ をサンプリングし，

$$\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}; \boldsymbol{0}, I) \tag{6}$$

$\boldsymbol{\epsilon}$ を平均 $\boldsymbol{\mu_\phi}$ だけ平行移動させ，標準偏差 $\boldsymbol{\sigma_\phi}$ で拡大縮小すれば，同じ潜在変数 $\boldsymbol{z}$ が得られると考える．

$$\boldsymbol{z} = \boldsymbol{\mu_\phi} + \boldsymbol{\sigma_\phi} \odot \boldsymbol{\epsilon} \tag{7}$$

このようにすることで，$\boldsymbol{z}$ に関する勾配計算ができるようになる．

2.2.4 正則化項の計算

正則化項は $p(\boldsymbol{z})$ に対する $q_\phi(\boldsymbol{z}|\boldsymbol{x})$ の KL Divergence である．$p(\boldsymbol{z})$ は標準正規分布 $\mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, I)$ であり，$q_\phi(\boldsymbol{z}|\boldsymbol{x})$ は，分散共分散行列 Σ が対角行列 $\mathrm{diag}(\boldsymbol{\sigma} \odot \boldsymbol{\sigma})$ である正規分布である．$\mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, I)$ に対する $\mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma} \odot \boldsymbol{\sigma}))$ の KL Divergence は

$$\begin{aligned} &D_{KL}\left(\mathcal{N}(\boldsymbol{z}; \boldsymbol{\mu}, \mathrm{diag}(\boldsymbol{\sigma} \odot \boldsymbol{\sigma}))||\mathcal{N}(\boldsymbol{z}; \boldsymbol{0}, I)\right) \\ &\quad = \frac{1}{2}\sum_{j=1}^{J}\left(\sigma_j^2 + \mu_j^2 - 1 - \log\sigma_j^2\right) \end{aligned} \tag{8}$$

と求められる．ここで，J は潜在変数の次元数である．

3. 動作データ

本章では，動作生成モデルの学習に使用する動作データセットと前処理について説明する．

本研究では，光学式モーションキャプチャシステムで計測した 2,505 個の人物動作データからなる CMU Graphics Lab Motion Capture Database[10] を使用する．元データのサンプリングレートは 120fps であるが，本研究では 30fps にダウンサンプリングしたものを使用する．

元の動作データは，図 1 に示すような 19 関節の 3 軸中心の局所回転角度とルート関節（Hip）の大局平行移動量として表現されている．本研究ではこれらをまず大局位置 $\boldsymbol{p}_j^{(g)}(t)$ に変換する．ここで，j は関節のインデクス番号を，t は時刻を表す．

次に，時刻 t における尻・左肩・右肩の大局位置をそれぞれ $\boldsymbol{p}_h^{(g)}(t)$, $\boldsymbol{p}_{ls}^{(g)}(t)$, $\boldsymbol{p}_{rs}^{(g)}(t)$ すると，体の前方を表すベクトル $\boldsymbol{v}_f(t)$ は

$$\boldsymbol{v}_l(t) = \boldsymbol{p}_{ls}^{(g)}(t) - \boldsymbol{p}_h^{(g)}(t) \tag{9}$$

$$\boldsymbol{v}_r(t) = \boldsymbol{p}_{rs}^{(g)}(t) - \boldsymbol{p}_h^{(g)}(t) \tag{10}$$

$$\boldsymbol{v}_f(t) = \boldsymbol{v}_l(t) \times \boldsymbol{v}_r(t). \tag{11}$$

と求められる．

本研究では，時刻 t における局所座標系の基底ベクトル $\boldsymbol{e}_y, \boldsymbol{e}_x(t), \boldsymbol{e}_z(t)$ がそれぞれ鉛直上向き，進行方向に対して右向き，進行方向となるように基底ベクトルを求める．

$$\boldsymbol{e}_y = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T \tag{12}$$

$$\boldsymbol{e}_x(t) = \frac{\boldsymbol{v}_f(t) \times \boldsymbol{e}_y}{|\boldsymbol{v}_f(t) \times \boldsymbol{e}_y|} \tag{13}$$

$$\boldsymbol{e}_z(t) = \boldsymbol{e}_y \times \boldsymbol{e}_x(t). \tag{14}$$

時刻 t における局所座標系の傾きを表す回転行列は

$$R_{gl}(t) = \begin{bmatrix} \boldsymbol{e}_x(t) & \boldsymbol{e}_y & \boldsymbol{e}_z(t) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{15}$$

と表すことができる．

$p_{\min,y}^{(g)}(0)$ を初期フレームにおける全ての関節位置の y 座標の最小値とし，初期フレームにおいていずれかの関節が床に接地しているとすると，$p_{\min,y}^{(g)}(0)$ は床の高さを表すこととなる．本研究では，時刻 t における局所座標系の原点をルート関節を床の上に真下に投影した点とする．時刻 t における局所座標系の位置を表す平行移動行列は

$$T_{gl}(t) = \begin{bmatrix} 1 & 0 & 0 & p_{h,x}^{(g)}(t) \\ 0 & 1 & 0 & p_{\min,y}^{(g)}(0) \\ 0 & 0 & 1 & p_{h,z}^{(g)}(t) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{16}$$

と表すことができる．ここで，$p_{h,x}^{(g)}(t)$, $p_{h,z}^{(g)}(t)$ はそれぞれ時刻 t におけるルート関節の x 座標，z 座標である．よって，局所座標系と大局座標系の相互変換行列は

$$M_{gl}(t) = T_{gl}(t)R_{gl}(t) \tag{17}$$

$$M_{lg}(t) = R_{gl}^T(t)T_{gl}^{-1}(t), \tag{18}$$

と表すことができ，時刻 t における各関節の局所座標は

$$\boldsymbol{p}_j^{(l)}(t) = M_{lg}(t)\boldsymbol{p}_j^{(g)}(t). \tag{19}$$

と求めることができる．

本研究では，ルート関節の y 座標と他の 18 の関節の xyz 座標を使用する． また，xz 平面内の速度と y 軸周りの角速度を使用する．これらは行列 $\Delta M(t)$ から計算することができる．

$$\Delta M(t) = M_{lg}(t-1)M_{gl}(t). \tag{20}$$

このようにして得られた動作データは各フレームに対して 58 次元ベクトルとなる．

最後に，フレームサイズ 128（約 4 秒）のウィンドウを 64 フレーム（約 2 秒）ずつオーバーラップさせて動作を分割した．その結果，13,032 個の動作データを得た．また，これらから平均を引き，標準偏差で割ることにより，データの標準化を行った．

4. 人物動作生成モデル

4.1 Variational Recurrent Neural Network

時刻 t の動作データを $\boldsymbol{x}_t$，潜在変数を $\boldsymbol{z}_t$ とすると，時間長 T の動作データと潜在変数の同時確率 $p(\boldsymbol{x}_{\leq T}, \boldsymbol{z}_{\leq T})$ は，連鎖率を使用して各時刻 t の確率の積に分解することができる．

$$p(\boldsymbol{x}_{\leq T}, \boldsymbol{z}_{\leq T}) = \prod_{t=1}^{T} p(\boldsymbol{x}_t, \boldsymbol{z}_t | \boldsymbol{x}_{<t}, \boldsymbol{z}_{<t}) \tag{21}$$

ただし，$p(\boldsymbol{x}_0, \boldsymbol{z}_0) = 1$ とする．また，各時刻の確率に対して条件付き確率の関係を使用すると，時刻 t における事前確率 $p(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t})$ と時刻 t における尤度 $p(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{\leq t})$ の積に分解することができる．

$$p(\boldsymbol{x}_{\leq T}, \boldsymbol{z}_{\leq T}) = \prod_{t=1}^{T} p(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{\leq t}) p(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t}) \tag{22}$$

本研究では，時刻 t における事前確率 $p(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t})$ と尤度 $p(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{\leq t})$ をニューラルネットワークにより表現する．また，時刻 t における事後確率 $p(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t})$ は intractable であるため，事後確率の近似 $q(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t})$ を別のニューラルネットワークを用いて表現する．

時刻 t における事前確率 $p(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t})$，尤度 $p(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{\leq t})$，事後確率の近似 $q(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t})$ は全て過去の時刻の動作データ $\boldsymbol{x}_{<t}$ と潜在変数 $\boldsymbol{z}_{<t}$ に依存することから，RNN によりこの依存関係を表現する．RNN を ρ，RNN の時刻 t における隠れ状態ベクトルを $\boldsymbol{h}_t$ とすると

$$\boldsymbol{h}_t = \rho(\boldsymbol{x}_t, \boldsymbol{z}_t, \boldsymbol{h}_{t-1}) \tag{23}$$

となる．事前確率 $p(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t})$ を正規分布 $\mathcal{N}(\boldsymbol{z}_t; \boldsymbol{\mu}_{p_t}, \mathrm{diag}(\boldsymbol{\sigma}_{p_t} \odot \boldsymbol{\sigma}_{p_t}))$ とすると，平均 $\boldsymbol{\mu}_{p_t}$ と標準偏差 $\boldsymbol{\sigma}_{p_t}$ をニューラルネットワーク ϕ_μ, ϕ_σ によりそれぞれ

$$\boldsymbol{\mu}_{p_t} = \phi_\mu(\boldsymbol{h}_{t-1}) \tag{24}$$

$$\boldsymbol{\sigma}_{p_t} = \phi_\sigma(\boldsymbol{h}_{t-1}) \tag{25}$$

と表す．ただし，$t = 0$ のときの事前確率は標準正規分布とする．また，事後確率の近似 $q(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t})$ は正規分布 $\mathcal{N}(\boldsymbol{z}_t; \boldsymbol{\mu}_{z_t}, \mathrm{diag}(\boldsymbol{\sigma}_{z_t} \odot \boldsymbol{\sigma}_{z_t}))$ と仮定し，平均 $\boldsymbol{\mu}_{z_t}$ と標準偏差 $\boldsymbol{\sigma}_{z_t}$ をエンコーダニューラルネットワーク $\varphi_\mu, \varphi_\sigma$ によりそれぞれ

$$\boldsymbol{\mu}_{z_t} = \varphi_\mu(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) \tag{26}$$

$$\boldsymbol{\sigma}_{z_t} = \varphi_\sigma(\boldsymbol{x}_t, \boldsymbol{h}_{t-1}) \tag{27}$$

と表す．また，尤度 $p(\boldsymbol{x}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{\leq t})$ は事前確率あるいは事後確率の近似からサンプリングされた潜在変数 $\boldsymbol{z}_t$ とデコーダニューラルネットワーク ψ を組み合わせて表現する．デコーダニューラルネットワーク ψ では潜在変数 $\boldsymbol{z}_t$ と隠れ状態ベクトルを $\boldsymbol{h}_{t-1}$ を入力し，動作データ $\boldsymbol{x}_t$ を出力する．

$$\boldsymbol{x}_t = \psi(\boldsymbol{z}_t, \boldsymbol{h}_{t-1}) \tag{28}$$

時刻 t における動作データ $\boldsymbol{x}_t$ の再現誤差は，$\boldsymbol{x}_t$ からエンコーダネットワークにより事後確率の近似を求め，サンプリングされた潜在変数 $\boldsymbol{z}_t$ からデコーダネットワークによって求められた動作データ $\tilde{\boldsymbol{x}}_t$ と元の動作データ $\boldsymbol{x}_t$ との平均 2 乗誤差によって求める．

$$\begin{aligned}&\mathbb{E}_{\boldsymbol{z}_t \sim q(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t})} |\psi(\boldsymbol{z}_t, \boldsymbol{h}_{t-1}) - \boldsymbol{x}_t|^2 = \\ &\mathbb{E}_{\boldsymbol{z}_t \sim \mathcal{N}(\boldsymbol{x}_t; \psi_\mu(\boldsymbol{z}_t, \boldsymbol{h}_{t-1}), \psi_\sigma(\boldsymbol{z}_t, \boldsymbol{h}_{t-1}))} |\psi(\boldsymbol{z}_t, \boldsymbol{h}_{t-1}) - \boldsymbol{x}_t|^2\end{aligned} \tag{29}$$

また，時刻 t における正則化項は，事前確率に対する事後確率の近似の KL Divergence によって求める．

$$D_{KL}\left(q(\boldsymbol{z}_t|\boldsymbol{x}_{\leq t}, \boldsymbol{z}_{<t}) || p(\boldsymbol{z}_t|\boldsymbol{x}_{<t}, \boldsymbol{z}_{<t})\right) \tag{30}$$

$$\begin{aligned}&= D_{KL}\left(\mathcal{N}(\boldsymbol{z}_t; \boldsymbol{\mu}_{z_t}, \boldsymbol{\sigma}_{z_t}) || \mathcal{N}(\boldsymbol{z}_t; \boldsymbol{\mu}_{p_t}, \boldsymbol{\sigma}_{p_t})\right) \\ &= \frac{1}{2}\sum_{j=1}^{J}\left(\frac{\sigma_{z_t,j}^2}{\sigma_{p_t,j}^2} + \frac{(\mu_{p_t,j} - \mu_{z_t,j})^2}{\sigma_{p_t,j}^2} - 1\right. \\ &\left. + \log\sigma_{p_t,j}^2 - \log\sigma_{z_t,j}^2\right)\end{aligned} \tag{31}$$

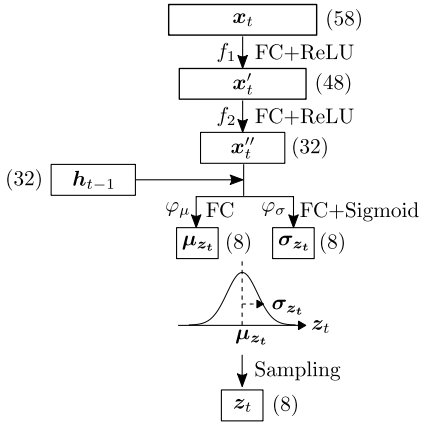

図 2 Encoder network.

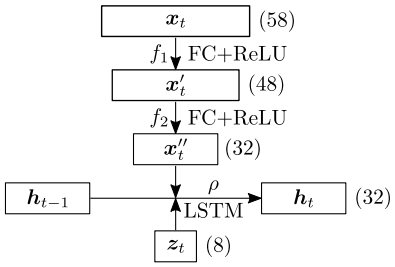

図 3 Recurrent network.

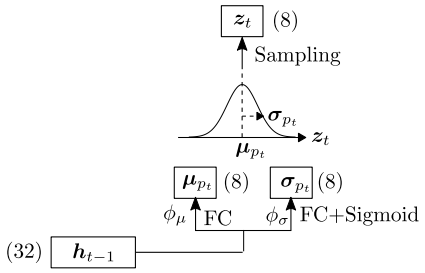

図 5 Prior network.

ここで，J は潜在変数の次元数である．時刻 t における誤差関数は再現誤差と KL Divergence の和とし，全時刻における誤差関数はこれを時間方向に加算したものとする．

4.2 モデルの構造

本研究で構築したモデルでは，58 次元の動作データ $\boldsymbol{x}_t$ を 8 次元の潜在変数 $\boldsymbol{z}_t$ で表現する．

$\boldsymbol{x}_t$ を $\boldsymbol{z}_t$ に変換するエンコーダネットワークの構造を図 2 に示す．提案モデルでは 2 層の特徴抽出器 f_1, f_2 により動作データから特徴抽出を行っている．f_1, f_2 は全結合層 (FC) と活性化関数 Rectified Linear Unit(ReLU) により構成されており，f_1, f_2 によって抽出される特徴量の数はそれぞれ 48, 32 とした．

また，図 3 に示すように RNN ρ には Long Short-Term Memory(LSTM) を使用し，動作特徴量 $\boldsymbol{x}_t''$ と潜在変数 $\boldsymbol{z}_t$ から隠れ状態ベクトル $\boldsymbol{h}_t$ を更新する．隠れ状態ベクトル $\boldsymbol{h}_t$ の次元数は 32 とした．

エンコーダネットワークでは，図 2 に示すように動作特徴量 $\boldsymbol{x}_t''$ と隠れ状態ベクトル $\boldsymbol{h}_{t-1}$ から 1 層の全結合層 $\varphi_\mu, \varphi_\sigma$ により平均と標準偏差を出力する．標準偏差を出力する層 φ_σ には sigmoid 関数をかけている．

デコーダネットワークは図 4 に示すように 3 層の全結合層 ψ, f'_1, f'_2 により構成されている．1 層目の ψ が潜在変数 $\boldsymbol{z}_t$ から動作特徴量 $\boldsymbol{x}''_t$ への変換であり，残りの 2 層 f'_1, f'_2 が動作特徴量 $\boldsymbol{x}''_t$ から動作データ $\boldsymbol{x}_t$ への変換となっている．ψ, f'_1 では ReLU 活性化関数をかけている．

事前確率ネットワークは図 5 に示すように 1 層の全結合層 ϕ_μ, ϕ_σ により構成されている．標準偏差を出力する層 ϕ_σ では sigmoid 関数をかけている．

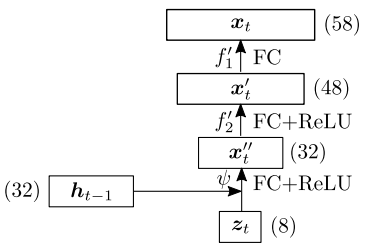

図 4 Decoder network.

5. 実験

3 章で述べた動作データを使用し，4 章で示した動作生成モデルの学習を行った．具体的には，学習データ数を 10,426，評価用データ数とテスト用データ数をそれぞれ 1,303，バッチサイズを 1,600 として学習を行った．学習のエポック数は 2,000 とした．また，最適化には adam を使用した．

構築した動作生成モデルを評価するための実験を行った．具体的には，事前確率ネットワークにより推定された事前確率分布から潜在変数 $\boldsymbol{z}_t$ をサンプリングし，それをデコーダネッワークによりデコードすることで，動作データ $\boldsymbol{x}_t$ を生成した．ランダムにサンプリングした 36 個の $\boldsymbol{z}_0$ から生成した動作データを図 6 に示す．図 6 に示すように多様な動作を生成することができた．生成された動作の各フレームの姿勢は自然な姿勢であることが多かったが，時間方向の姿勢の変化が滑らかでなく，高周波成分の多い動作となった．

学習データには，逆立ちしてから床に寝転がって回転するブレイクダンスのような動作やクロールや平泳ぎといった泳ぐ動作も含まれている．図 7 に示すように，逆立ちするような動作が生成されることはあったが，動作の途中で人体の構造が崩れてしまうことが多かった．また，泳ぐような動作も生成され，図 8 に示すように各フレームの姿勢は自然な姿勢であったが，時間方向の姿勢の変化が不自然であった．これは，学習データに含まれる動作のほとんどは立った状態で行われるのに対して，寝転がって行われる動作が少数であるためだと思われる．

6. まとめ

本研究では，3 次元コンピュータグラフィックスのキャラクタの動作生成を目的とし，RNN により過去の動作との依存関係を表現すると同時に，多階層のニューラルネットワークにより動作特徴量を抽出し，低次元の潜在変数空間に確率分布として動作特徴を表現することができるモデルを構築した．モーションキャプチャデータセットを用いて学習した結果，提案モデルは 8 次元の潜在変数空間からサンプリングすることで多様な動作を生成できることを示

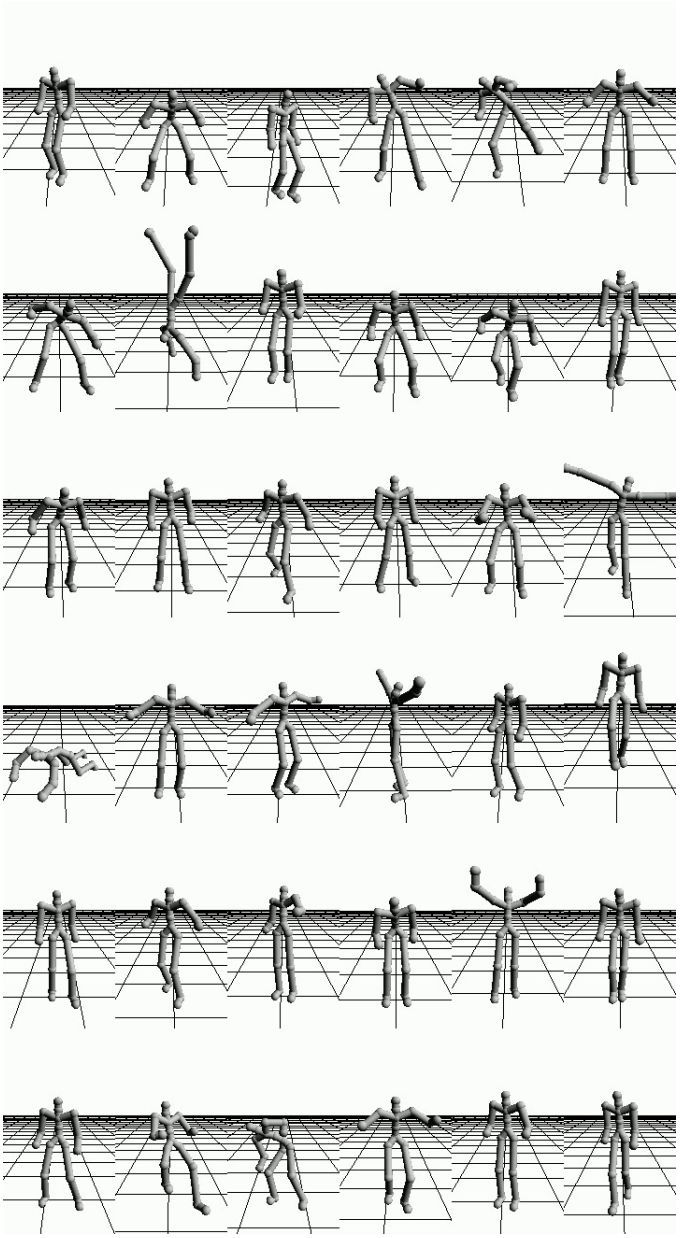

図 6 36 randomly generated motions.

した．

提案モデルで生成した動作の各フレームの姿勢は自然であったが，時間方向の姿勢の変化が滑らかでなかった．提案モデルでは動作データを 1 フレームずつモデルに入力し学習を行ったが，今後は，あるフレーム長の動作データをモデルに入力し，短時間フィルタリングをモデル中で行う

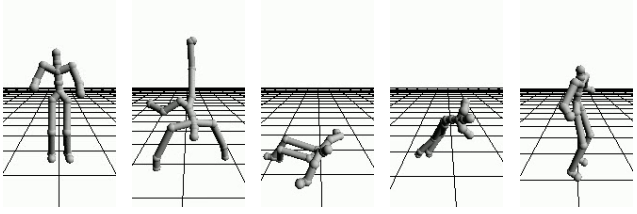

図 7 Generated handstand and spin motion.

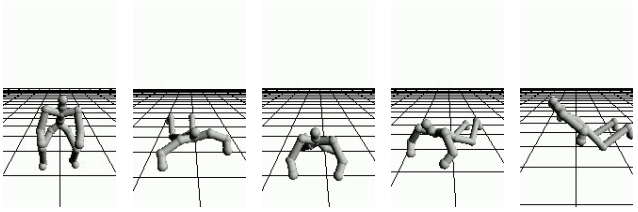

図 8 Generated swim motion.

予定である．これにより動作の短時間の連続性をモデル化できれば，滑らかな動作の生成が期待できる．

謝辞 本研究は JSPS 科研費 JP19K12287 の助成を受けたものです．

参考文献

[1] Wang, X., Chen, Q. and Wang, W.: 3D Human Motion Editing and Synthesis : A Survey, *Computational and Mathematical Methods in Medicine*, Vol. 2014 (online), DOI: http://dx.doi.org/10.1155/2014/104535 (2014).

[2] Holden, D., Saito, J., Komura, T. and Joyce, T.: Learning Motion Manifolds with Convolutional Autoencoders, *SIGGRAPH Asia 2015 Technical Briefs*, SA '15, New York, NY, USA, ACM, pp. 18:1—-18:4 (online), DOI: 10.1145/2820903.2820918 (2015).

[3] Holden, D., Saito, J. and Komura, T.: A Deep Learning Framework for Character Motion Synthesis and Editing, *ACM Trans. Graph.*, Vol. 35, No. 4, pp. 138:1—-138:11 (online), DOI: 10.1145/2897824.2925975 (2016).

[4] Kingma, D. P. and Welling, M.: Auto-Encoding Variational Bayes, *International Conference on Learning Representations 2013* (2013).

[5] Kingma, D. P., Rezende, D. J., Mohamed, S. and Welling, M.: Semi-Supervised Learning with Deep Generative Models, *Neural Information Processing Systems 2014* (2014).

[6] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y.: *Advances in Neural Information Processing Systems 27*.

[7] Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Jozefowicz, R. and Bengio, S.: Generating Sentences from a Continuous Space, *SIGNLL Conference on Computational Natural Language Learning 2016* (2016).

[8] Fabius, O. and van Amersfoort, J. R.: Variational Recurrent Auto-Encoders, *International Conference on Learning Representations 2014* (2014).

[9] Sabathe, R., Coutinho, E. and Schuller, B.: Deep recurrent music writer: Memory-enhanced variational autoencoder-based musical score composition and an objective measure, *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 3467–3474 (online), DOI: 10.1109/IJCNN.2017.7966292 (2017).

[10] CMU: *Carnegie Mellon University - CMU Graphics Lab - motion capture library*, http://mocap.cs.cmu.edu.