

## 舞踊譜 Labanotation による人間の身体運動の入力

高地泰浩 八村広三郎 英保 茂

京都大学 工学部 応用システム科学教室

舞踊譜 Labanotation に基づいて人間の身体運動を符号化するシステムについて述べる。複雑な身体運動を人間に理解しやすく記述する方法として、舞踊の分野では図形的な記号を用いた Labanotation が考案されている。我々は XWindow システムを実装したワークステーションの画面上で、この Labanotation を対話的に入力・編集し、運動を再現するシステムを開発した。開発にあたっては、理解しやすいユーザーインターフェイスを実現することを念頭に置いた。本システムによって、運動を直感的に理解できる Labanotation の特長を生かして、身体運動を効率的に入力することが可能となった。最後にいくつかの問題点と課題について述べる。

## Human Body Movement Coding Based on Labanotation

Yasuhiro Kochi Kozaburo Hachimura Shigeru Eiho

Division of Applied Systems Science

Faculty of Engineering  
Kyoto University

The coding of human body movement based on Labanotation is discussed. Graphical notation method called Labanotation has been devised and used in the field of dance for describing complicated movement of human body. We developed a system to input and edit Labanotation score interactively on a screen of XWindow terminal. The Labanotation data entered is interpreted by a simulator, and body motion is displayed. We developed it with emphasis on user-friendly interface. This system helps us to input the body movement efficiently.

## 1 まえがき

人間の身体運動を記述・入力し、コンピュータのディスプレイ上に再現することができれば、これを身体運動の解析や比較、マンマシンインターフェイス、アニメーション、ロボット制御、あるいは医療などに応用することができる。このためには、身体運動を記録・入力し、必要な処理を加えて再び表示するためのシステムが求められる。[1]

これまで、人間の身体運動を記述するための様々な手法やシステムが考案されているが、これらは、必ずしも、運動を正確に表現し、かつ容易に理解できるものではなかった。一方、身体運動を譜面（舞踊譜）の形で記録する、Labanotation と呼ばれる記述法が考案されている [2]。この記述法は、元来、ダンスや舞踊の記録・出版をおもな目的としたもので、コンピュータに入力を目的にして考案されたものではない。しかし、Labanotation は、舞踊に現れるさまざまな身体運動を身体各部の細かな動きに分解して記述できる一方、譜面として視覚的に表現されるため、人間に理解しやすい。このため、その基本的な思想や枠組みは身体運動のコンピュータ処理の研究でも広く利用されている。

我々は、この Labanotation の記述法に基づいて一般の身体運動を記述し、コンピュータで蓄積、編集、表示などを行うための総合的なシステムについて研究を行っている [1] が、今回、このシステムのデータ入力サブシステムの一つとして、Labanotation による身体運動の譜面を対話的に作成・入力するため、Labanotation エディタを作成したので報告する。

本サブシステムを利用することにより、図面としての Labanotation を対話的に入力・編集し、作成されたデータの表す運動を、運動シミュレータ [3][1] によりアニメーション動画としてただちに画面上に表示することができるようになった。

対話的図形エディタである Labanotation エディタで作成されたデータは、シミュレータの入力データの形式である BN に変換してからシミュレータに渡される。今回のシステムには、このデータ形式の変換を行うコンバータの部分も組み込まれている。BN を解釈して運動を表示するシミュレータについ

ては既に報告 [1] しているので、本報告では、特に、身体運動入力と編集のための Labanotation エディタについてだけ報告する。

## 2 Labanotation による身体運動の記述

人間の身体運動には以下のような特徴がある。

- 人体は複数の要素から構成されている
- 複数の要素が同時に、連携して動作する
- 関節は多くの自由度を持つが、各要素の動作により互いに制約を受ける
- 人体の運動は外界、特に、地面との相互関係で表現する必要がある

このような身体運動を正確に記述しようとする、身体の各要素の動きを細かく記録する必要がある。そのような記述は複雑となり、本質的な運動の様子は多くの情報の中に埋もれてしまう。その反対に基本的な運動情報や、単純なモデルと動作関数による記述では、多様で複雑な運動を再現することができない。

Labanotation による運動記述は、音楽における楽譜（五線譜）のように図形的に表現されるため人間にとって理解しやすいという長所がある。ただし、フルセットの Labanotation は、非常に豊富な表現能力を持っているため、この全容を理解し完全に習得するには多くの時間がかかるといわれている。しかし、Labanotation の基本的な部分だけでもある程度の身体運動を記述することが可能なため、我々のシステムではとりあえず Labanotation の基本部分を対象とすることにした。

図 1 は Labanotation のコラムのレイアウトを示している。それぞれのコラムに対応する体の部分の動きを現す図的記号（Labanotation シンボル）を書き込んでゆく。Labanotation では時間の流れは下から上の方向で表され、下から上へと縦に読んでいく。図 2 は Labanotation で使われるシンボルを示している。たとえば、内部を黒く塗りつぶした左向きの三角形で、左下方向への運動を示しており、

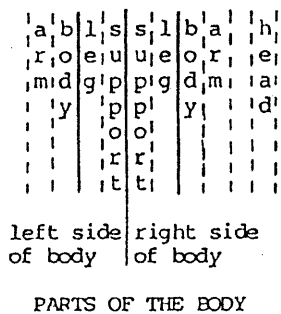


図 1: Labanotation のコラムレイアウト

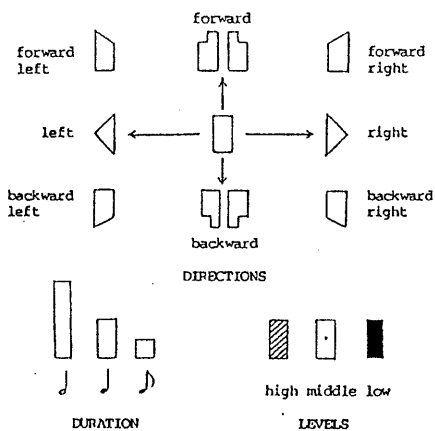


図 2: Labanotation のシンボル

これが左腕のコラムに記入されておれば、左腕を左下に動かすという動作を表現する。

### 3 Labanotation エディタ LNE

#### 3.1 システムの概略

身体運動を文字や数字だけで記述して、テキストエディタなどで入力するのは非効率的であるうえ、理解しやすいとはいえない。そこで身体運動を図形的に表現する Labanotation をコンピュータに入力するための専用のエディタが必要である。我々は、XWindow システムが使用可能な UNIX ワー

クステーション (SUN Sparc Station, サンマイクロシステムズ社製) 上で、主としてマウスを用いた操作で対話的に Labanotation を編集できるエディタ、LNE(LabaNotation Editor)を開発した。

開発にあたって特に留意した点は、初心者にも理解しやすいユーザーインターフェイスによる、必要かつ合理的な機能を提供することである。

LNE の開発には XWindow Ver11 上で GNU-C compiler を用いた。XWindow システムは、UNIX ワークステーション上で標準となったウィンドウシステムで、高度な画面表示や複数プログラムの同時実行が可能なシステムである。XWindow 上で画面に文字や図形を表示するためには、Xlib と呼ばれる、画面描画やキー入力を制御するルーチン群を利用する。[4][5]

#### 3.2 LNE の仕様と操作法

エディタ LNE では Labanotation の基本部分 (サブセット) を入力・編集の対象としている。すなわち、身体各部の運動の方向、高さ、長さを、それぞれ 8 方向、3 レベル、8 単位で表現し、このために必要な Labanotation のシンボルを用いて譜面を作成・編集することが可能である。

エディタの画面の一例を図 3 に示す。画面上部に並んでいるのが、それぞれの編集機能を起動するための機能ボタンである。画面の左側に表示された矢印はスクロールボタンで、譜面を上下方向にスクロールするために使用する。画面中央に表示されているのが、Labanotation そのものを書き込むためのコラムである。その右側に、シンボルの要素 (運動の方向、高さ、長さ) を選択するパレットがある。画面右下の部分はシステムからのメッセージを表示したりキーボードから文字を入力するのに使用する部分である。

LNE のおもな機能は、

- シンボルの記入 (PUT)
- 範囲を指定しての切り取り (CUT)
- 複写 (COPY)
- 貼り込み (PASTE)
- 空行の挿入

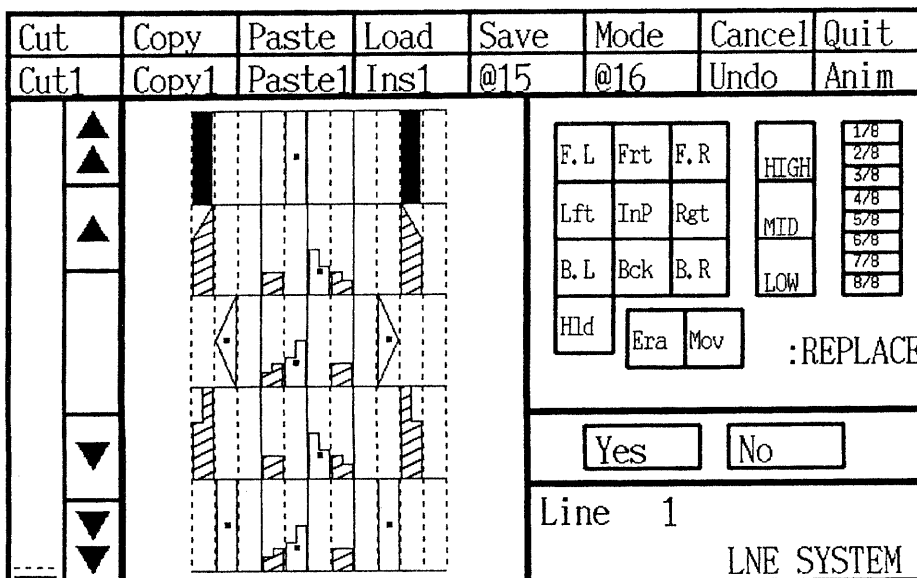


図 3: LNE の画面レイアウト

- 運動シミュレータ用の BN データ生成
- Labanotation データのファイル入出力

などである。画面上に表示された機能ボタンをマウスカーソルでクリックすることによって、これらの各機能が起動される。

LNE では、「どうする」ということを指示してから「何を」ということを指示する、いわゆる、Verb-Noun 型の指示体系を用いている。これは、一般のテキストエディタと異なり、現在注目している点を示すカーソルを使用しないことに起因している。

たとえば、CUT や COPY の機能は、対応する機能ボタンをクリックして機能を指定してから、処理の対象となる領域の始点と終点を指定することによって実行される。また、PASTE 機能は、PASTE ボタンをクリックした後で、始点を指定すると実行される。

PUT 機能に関しては、対応する機能ボタンは存在せず、機能ボタンをクリックする操作は必要ない。すなわち、Labanotation のシンボルは方向、高さ、長さの 3 要素を持つが、それぞれの要素について、画面上に表示されたパレット上で、必要なものをマウスカーソルをクリックすることで、3 要素を指定

し、その結果、特定のシンボルを選択することができる。このようにして選択されたシンボルは画面の右に表示されている。このようにしてシンボルを選択した後、マウスカーソルをカラム上に適当な位置に移動してマウスのボタンをクリックすると、その場所にシンボルが置かれる。

さらに、特別なシンボルとして消しゴム (eraser) と移動 (mover) がある。eraser を選択してから、カラム上のシンボルをクリックするとその場所にあったシンボルが消去される。mover を選択してからカラム上でクリックすると、そのシンボルが消去されるころまでは eraser と同様であるが、消去されたシンボルは取り込まれて表示される。再びカラム上でクリックすると、取り込まれていたシンボルがその場所に置かれる。

また、画面上に表示されるカラムは 5 小節分であるが、画面左側の矢印をクリックすることで上下にスクロールすることができる。

LNE は可能な限り誤操作を防ぐように設計されている。たとえば、CUT や COPY などの機能で始点や終点を指定する場合は、マウスカーソルの形がそれまでの手の形から台付き矢印に変化して、そ

れぞれ、始点や終点を指定する状態にあることを示すようにしている。また、ファイルへの書き出しの際に指定されたファイル名のファイルが既に存在する場合、ファイルを読み込む場合、終了などの場合には、必ずユーザーに確認を求めるようになっていいる。Yes/No ボタンはそのための確認用である。

### 3.3 LNE の内部構造

LNE はユーザーの操作に応答することで動作する対話型のシステムである。このため、すべての動作はユーザーが何らかの操作を与えることで起動される。ユーザーの操作とはマウスのボタンを押したりキーボードを押したりすることである。これらの事象をイベント (event) と呼び、イベントの発生は XWindow システムによって LNE に報告される。LNE は、発生したイベントによって駆動されるイベント駆動型のシステムとして実現している。<sup>1</sup>

LNE は、現在、どの機能のどの実行段階にあるかということ状態として保持している。その時点での状態と、発生したイベントの組み合わせによって、後述する遷移表を参照し、次に実行すべき機能を決定する。この機能はサブファンクションと呼び、基本的な機能 (例えば画面の再描画、位置取得、CUT など) を実行するものである。

実行すべきサブファンクションが決定すると、そのサブファンクションを呼び出す。各サブファンクションはその機能の成功・不成功を値として返す。その値によって、次に遷移すべき状態を決定し、状態遷移を行う。

これらの情報は図 4 のような状態表にまとめて記述されている。状態表はプログラムとは独立したファイルとして保存されており、LNE の起動時にプログラム内に読み込まれる。状態表の (状態番号) 行、(イベント番号) 列には実行すべきサブファンクションと、サブファンクション成功時の次状態、失敗時の次状態が記入されている。

このように、条件と実行機能の指定をデータとして分離したので、プログラムのコードがシンプルかつコンパクトになり、また、システムの機能を変

<sup>1</sup> XWindow システムが扱うイベントは数多くあるが、LNE はその一部のみを利用している

```

:00
; コメント
;   発生イベント                次状態
;   呼び出す機能 成功時 失敗時
P_C   PutSymbol      01    00
P_PAL_D GetSymbolD   00    00
P_PAL_H GetSymbolH   00    00
P_PAL_L GetSymbolL   00    00
P_CUT  CursorFrom    02    02
P_CPY  CursorFrom    08    08
P_PST  CursorTo      13    13
P_LD   GetFileName   17    00
P_SV   GetFileName   23    00
P_USL  RollUp1       00    00
P_USP  RollUpPage    00    00
P_DSL  RollDown1     00    00
P_DSP  RollDownPage  00    00

```

図 4: 状態表 (抜粋)

更したり、ユーザーが一部をカスタマイズしたりすることが容易に行えるという特徴がある。

個々の機能を実現するサブファンクション部分は、比較的単純な動作をするのみで、制御の流れも上から下への一方通行となっている。サブファンクションのなかで最も複雑な CUT と PASTE サブファンクションでも、C 言語のソースプログラムで 40 行程度のサイズに収まっている。

### 3.4 LNE の内部データの構造

LNE の内部データとしては、人体のリンク構造などを反映した特別のデータ構造を利用することは行っていない。すなわち、画面上の各図形 (シンボル) は、これと一対一で対応する単なる記号として表現され、フラットなデータとして扱われている。これは、LNE の入力、編集作業における処理の効率を考慮したためである。

このことは、LNE では、意味のない運動や、実現不可能な運動であっても入力は可能であることを意味している。ただし、運動の妥当性や可能性は、

現在では、動作シミュレータの方でチェック機構が働くようになっている。音楽の五線譜に記された音符の列が、音楽として必ずしも適切なものになってはいない場合があるように、入力するシンボルの意味的な正当性は人間によって保証されるべきであるという立場にたっていると考えることもできる。しかしながら、身体の物理的な条件からくる運動上の制約などについては、入力時にチェックするべきであろうし、LNE内で、データの探索(サーチ)を考えると、単純なストリングマッチ的な探索だけでなく、身体運動の動作の意味を考慮した探索機能を実現するためには、身体構造や各種の制約などを反映したデータ構造を持ったデータとして保持する必要があると考えられる。これは将来の課題である。

### 3.5 BN への変換と運動の表示

我々のシステムで対象とした、Labanotation の基本部分(サブセット)で記述できる身体運動の要素は、

- 運動の方向について水平面内で8方向および静止
- 高さについて上中下の3通り
- 運動に要する時間について1/8 ~ 8/8 単位時間

であり、それらを身体の11部位について個別に記述できる。

運動を画面に表示するためのシミュレータ(HBMと呼ぶ)は、BNを解釈・実行するように作成されているので、LNEで作成された運動データをBNへ変換する必要がある。BN自身がLabanotationを参考にして設計されているため、これは比較的簡単な処理で実行することが可能である。変換を受け持つコンバータはLNEの内部に含まれている。しかし、シミュレータは別個の独立プログラムとして実現されているので、何らかの形でデータを受け渡す必要がある。今回は最も単純な、ファイルによる受け渡しを採用した。

運動を対話的に編集する場合は、LNEとシミュレータHBMを、UNIXのマルチタスク機能を利用して同時に起動する。シンボルを入力・編集した

後、Animボタンをクリックすると、LabanotationデータがBNへ変換され、BNのファイルが生成される。このファイルをHBMで読み込み、スティックフィギュアでモデルで身体運動の表示を行う。LNEで入力・編集した結果をただちに身体運動の表示に反映させるためには、任意の時点でAnimボタンをクリックすればよい。現在再現中の運動が終了してHBMが次の運動を読み込むときに、更新された、新しい運動データを読み込み表示するようになっている。

### 3.6 Labanotation の入力例

図5にLNEで実際に入力したLabanotationの例を示す。図5(a)は体操で、屈伸と腕の運動を行っている。図5(b)は手を振り回す動作である。

図6は、XWindow上でLabanotationエディタLNEとシミュレータHBMを同時に起動して、入力とその結果の表示を行っている際の画面の様子を示している。このように、LNEをHBMと連動させることで、Labanotationで入力した運動をただちにスティックフィギュアモデルによるアニメーション動作で確認することができる。

## 4 評価

ここでは、いくつかの身体運動を記述してみた経験から、本システムの評価について述べる。

LNEでは、必要なシンボルをマウスでクリックするだけで舞踊譜上に記入でき、CUT,COPY,PASTE等でも不自然な操作を要求されることもなく、スムーズにシンボルを入力することが可能である。シンボルの表示等で画面操作を多用しているため、速度低下が心配されたが、十分な応答速度を得ることができた。

Labanotationを用いて身体運動を図形的に表現することの利点として、直感的に運動を理解できることがある。特に、水平面内での運動方向を表わすシンボルの形状については、実際の運動方向と対応しており理解しやすい。また、シンボル内部のパターンで表わされる垂直面内の方向も、高々3レベルであり、直ちに習得できるものである。フルセッ

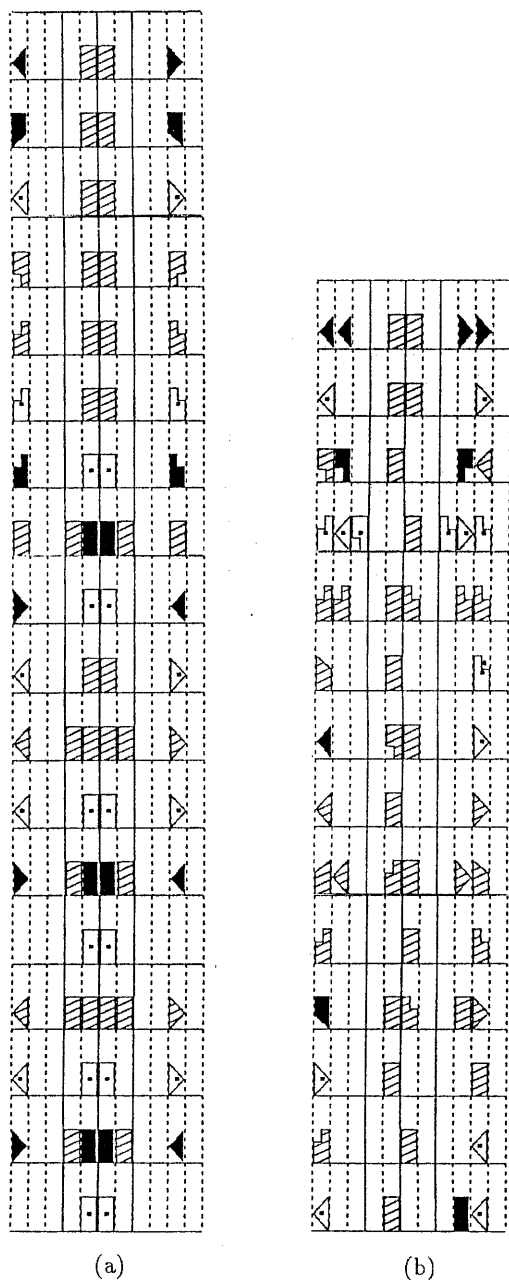


図 5: Labanotation の入力例

トの Labanotation は習得に時間がかかるものであるが、このような基本的な部分に限れば、容易に習得できるものであることが明らかになった。しかしながら、舞踊譜におけるそれぞれの列が身体のだの部位に対応するのかということについては、本来の Labanotation では譜面に記入されていないので分かりづらい。何らかの工夫が必要であろう。

このように、運動を直感的に理解できるという Labanotation の特徴を生かして、比較的容易に運動を入力することができることが分かった。

次に、本システムが扱うサブセットの Labanotation と BN の表現能力について考察する。Labanotation では 11 部位について 8 方向 × 3 レベルの表現が可能で、これはかなりの表現パターンを有しているように思える。ところが、実際にこのシステムで運動を記述してみたて次のような問題点があることが分かった。

Labanotation 自体の規格として、シンボルの時間軸方向の長さはその運動が達成されるまでの時間を表わすことになっている。そのため、シンボルの長さで運動の大きさを直接表わすことができない。これは元来 Labanotation が舞踊を記述するために開発されたものであるから、伴奏の音楽に合わせ、身体の大動作で表現を行う舞踊に適した表記となっているためであると考えられる。一方、我々の対象とする運動は一般の運動であり、必ずしも上に挙げたような運動ばかりではない。そのため、運動の大きさを指示することによって微妙な運動を記述することができないという問題点が残っている。これについては、各種の運動のマクロライブラリを作成して、これと Labanotation とを組み合わせることで表現能力の不足を補ったり、BN による低レベルの記述を組み合わせることで運動を表現することで改良することもできると考えている。

## 5 あとがき

Labanotation に基づく身体運動の記述・表示システムにおける入力サブシステム Labanotation エディタについて報告した。

Labanotation による運動の入力は、図形的な表現をとっているため、利用者にとって、直感的に理

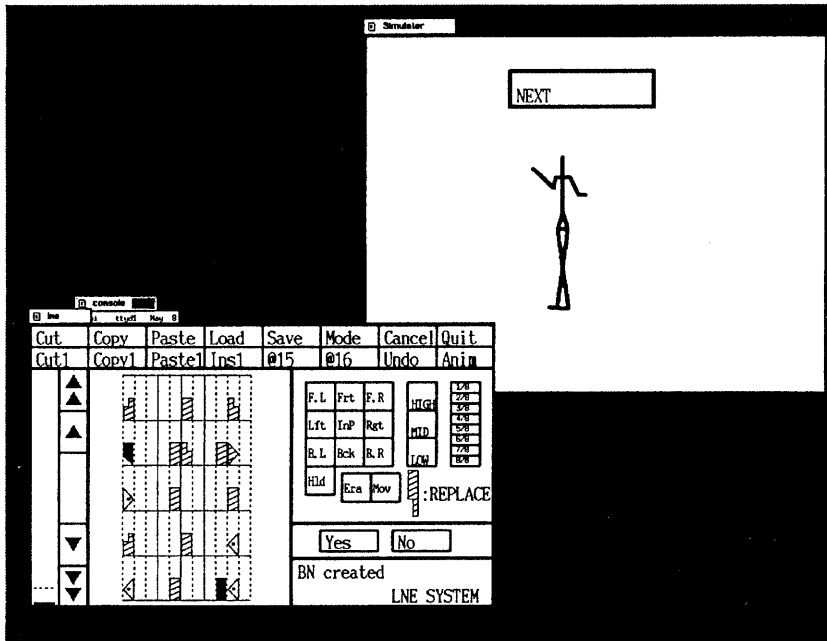


図 6: LNE と HBM による動作表示

解しやすい利点がある。また対話的に入力・編集と動作の確認ができるので、親しみやすく容易に利用することができる。しかし、現状のシステムは、必ずしも完全なものではなく、今後の課題としては

- より細かな動作の表示能力の改善
- ユーザーインターフェイスの Look-And-Feel の改善
- シミュレータ、特に表示モデルの改善
- 内部データ構造の改善

などが残されている。

## 参考文献

- [1] 八村広三郎:「身体運動情報の処理」、情報処理学会人文科学とコンピュータ研究会資料、1990/9/14
- [2] Peggy Hackney, Sarah Manno and Muriel Topaz: 'Study Guide for Elementary Labanotation', New York Danced Notation Bureau Press, 1977

- [3] 寺井英夫: 'A System for Description of Human Body Movement'、京大工学部情報工学教室修士論文、1984
- [4] 木下凌一:「XWindow Ver.11 プログラミング」、日刊工業新聞社、1989
- [5] D.A. ヤング (川手恭輔 訳):「X Toolkit プログラミング」、トッパン、1990