

キャラクターアニメーションの 数理とシステム

3次元ゲームにおける
— 身体運動生成と人工知能 —

博士（工学） 向井 智彦

博士（工学） 川地 克明 【共著】

三宅陽一郎

コロナ社

ま え が き

本書は、インタラクティブな3次元コンピュータグラフィックス(3次元CG)の映像における、キャラクタの動きを生成する技術に焦点をあてた教科書である。対象とする読者は3次元CGの数理的な基礎知識を持つことを前提とし、動的なキャラクタアニメーションを担うソフトウェアシステムに必要な技術要素とその構成方法について、初めて学ぶことを想定している。

現在、3次元CGは映画やテレビコマーシャルなどの映像を作成する手段として広く利用されている。また、プレイヤーの対話的な操作入力によって映像を動的に変化させることを目的として、ビデオゲームやバーチャルリアリティシステムといったソフトウェアにも応用されている。エンターテインメント作品では人間や動物に類する形態の魅力的なキャラクタがCG映像中に登場し、こうしたキャラクタは作品の表現を支える中心的な働きをする。

計算機の演算処理能力向上により、近年ではCG映像の中で高度に実在感のあるキャラクタの所作を表現することが可能になっている。これは同時に、映像を生成する計算機の処理能力によって表現力が制限されることを意味している。一般に、インタラクティブな映像生成システムで利用できる計算機の能力は、映画などの静的な映像制作に使用される計算能力よりもはるかに制限されたものになる。ビデオゲームの映像で身体運動の表現力が場面によって大きく変化する現象はその好例であり、キャラクタがプレイヤーの入力操作によって動作している場面と、事前に準備した静的データに従って運動している場面とでは、利用している計算能力に大きな差があることを直接反映している。

本書では、計算能力に制約のある3次元CGアプリケーションにおけるアニメーションシステムの設計概念や各種要素技術について、体系的な説明を試みている。具体的には、プレイヤーの操作入力やゲームAIの状況判断に応じてキャ

ラクタの運動を高速かつ堅牢に即時生成するための基盤技術を取り扱う。

本書に類する題材を扱った書籍としては、ヒューマノイドロボットの運動生成に関する技術書が挙げられる。著者も含む多くのキャラクターアニメーション技術の研究開発者は、ロボット工学分野の書籍を通じてフォワードキネマティクス(3.4節)やインバースキネマティクス(5.1節)、動力学計算(5.2節)などの要素技術を学んできた。しかし、現実世界のロボットとは異なり、ビデオゲームなどのアプリケーションでは物理法則よりも制作者の演出意図を優先する必要がある、専門技能を持つアニメータの手によってキャラクターの振舞いがデザインされている。本書ではこのようなCGアプリケーションに特有の制作工程を踏まえ、著者自身が初学者のときに学びたかった内容を中心に、キャラクターアニメーション生成技術の基礎知識を網羅的に説明している。なお、本書の執筆分担は、向井(第2~4章, 付録), 川地(第1, 5, 6章, 付録), 三宅(第7章)である。

本書にまとめた知識は、アニメーション生成技術の開発や研究だけでなく、各種ゲームエンジンで提供されているアニメーションシステムの動作を深く理解する手助けにもなると期待している。また、可能な限り原典を掲載するとともに、比較的新しい研究事例についても紹介している。本書を手にとった学生・開発者・研究者の一助になればたいへん幸いである。

そして、本書で解説したキャラクターアニメーションの技術構成は、あくまで現時点での局所的な最適解の一つであることに留意してほしい。将来、革新的な人工知能のアルゴリズムがキャラクターの振舞いを司ることで、先進的な映像出力デバイス上にいままでにない遊びの楽しさが現れるであろうことを見据え、これをキャラクターの身体運動として実現するアニメーション技術にもまた革新が必要である。本書をきっかけとしてキャラクターアニメーション技術に携わる開発者・研究者の同志が増えることで、素晴らしいCGアニメーションを目にする機会に恵まれることを願う。

2020年6月

著者一同

目 次

1. 概 論

1.1	ゲームプログラムにおけるアニメーションシステム	1
1.2	ソフトウェア機能部品としてのアニメーションシステム	3
1.3	アニメーションシステムのための静的データ	4
1.4	アニメーション計算処理アルゴリズムの特徴	5
1.4.1	計算の即応性に関する制約	6
1.4.2	計算の制御性に関する制約	6
1.4.3	計算の安定性に関する制約	7
1.5	ま と め	8

2. 形状変形アニメーション

2.1	形状変形アニメーションのデータ表現	9
2.2	頂点アニメーション	11
2.3	ブレンドシェイプ	13
2.3.1	ブレンドシェイプの原理	13
2.3.2	線形ブレンドシェイプ	16
2.3.3	加算ブレンドシェイプ	17
2.4	ポーズスペース変形法	20
2.4.1	区分線形補間法	22
2.4.2	散布データ補間法	24
2.5	自由形状変形	30

2.6 スキンモデル	34
2.6.1 スキンモデルの概要	34
2.6.2 スキンモデルの形状変形	35
2.6.3 スキンモデルとスケルトン	36
2.7 物理シミュレーション	38
2.7.1 質点系力学に基づく形状変形	39
2.7.2 バネ-マス-ダンパモデル	42
2.7.3 フィードバック制御	43
2.8 リ グ	45
2.9 発展的な話題	50

3. スケルトンアニメーション

3.1 キャラクタスキンモデル	52
3.1.1 ジョイント座標系	53
3.1.2 ワールド座標変換	55
3.1.3 線形ブレンドスキニング	56
3.1.4 行列パレット	59
3.2 スケルトン	61
3.2.1 スケルトンの概要	62
3.2.2 ジョイント階層構造	65
3.2.3 アニメーションジョイント	67
3.2.4 ル ー ト	68
3.3 副次ジョイント	71
3.3.1 補助ジョイント	71
3.3.2 アタッチメントジョイント	73
3.4 フォワードキネマティクス	75
3.4.1 ローカル姿勢とワールド姿勢	75
3.4.2 ローカル行列の構成	78
3.4.3 ワールド行列の計算	80

3.5 スケルトンアニメーションデータ	83
3.5.1 キーフレームアニメーション	83
3.5.2 キーポーズ	85
3.5.3 区分線形キーフレーム補間法	86
3.5.4 スプライン補間法	89
3.6 アニメーションデータの圧縮	91
3.6.1 定数キーの省略	91
3.6.2 アニメーションキーの圧縮	92
3.6.3 キーフレームリダクション	93
3.7 発展的な話題	95

4. アニメーションシステム

4.1 アニメーションシステムの概要	96
4.2 アニメーションの滑らかさ	100
4.3 アニメーションクリップ	102
4.3.1 アニメーションクリップのデータ構造	102
4.3.2 サンプリング	103
4.3.3 クリップの即時切り替え再生	104
4.3.4 ループアニメーション	105
4.4 ポーズブレンド	106
4.4.1 線形ポーズブレンド	107
4.4.2 加算ポーズブレンド	109
4.4.3 パラメトリックポーズブレンド	111
4.5 トランジション	114
4.5.1 線形クロスフェード	116
4.5.2 イーズイン・アウトクロスフェード	117
4.5.3 フリーズトランジション	118
4.5.4 慣性補間	120
4.5.5 補間トランジション	121

4.6	アニメーションレイヤ	123
4.7	ステートマシン	125
4.7.1	ステートマシンの概要	125
4.7.2	単一クリップをステートとする構成	126
4.7.3	アニメーションレイヤとステートマシン	128
4.8	ブレンドツリー	129
4.8.1	ブレンドツリーの概要	129
4.8.2	ステートマシンとブレンドツリーの連携	131
4.9	ステート遷移の自動化	132
4.9.1	位相情報を用いた周期的動作の同期	132
4.9.2	モーショングラフ	134
4.9.3	モーションマッチング	137
4.10	発展的な話題	139

5. 環境への適応

5.1	インバースキネマティクス	141
5.1.1	外的環境への対応	142
5.1.2	姿勢のブレンドによる関節位置の誤差	143
5.1.3	1関節のインバースキネマティクス	145
5.1.4	2関節のインバースキネマティクス	147
5.1.5	3関節以上のインバースキネマティクス	149
5.1.6	CCD IK	152
5.1.7	パーティクル IK	153
5.2	動力学の利用	156
5.2.1	動力学シミュレーション手法の利用	156
5.2.2	動力学シミュレーションのパラメータ	157
5.2.3	代表形状による身体形状表現	157
5.2.4	体節に働く力と運動の変化の関係	158
5.2.5	力と加速度に対する拘束条件	159
5.2.6	動力学シミュレーションにおける摩擦とダンピング	161

5.2.7	パーティクル IK によるラグドールシミュレーション	162
5.2.8	PD 制御による動きの重畳	164
5.3	環境形状の検査	166
5.3.1	線分との交差判定	167
5.3.2	複雑な形状との交差判定	168
5.3.3	移動する形状との交差判定	168
5.3.4	地形の凹凸に沿った移動	169
5.3.5	交差判定のための代表形状	171
5.3.6	交差判定の効率化	171
5.3.7	交差判定の集約	172

6. 連携と疎通

6.1	アニメーションシステムの外部連携	173
6.1.1	外部システムとの連携メカニズム	174
6.1.2	通知のコールバックとキュー	175
6.1.3	通知の優先度	176
6.1.4	通知データの作成と記録	177
6.1.5	時間幅を持つ通知	178
6.2	アニメーション付加情報の活用	179
6.2.1	アニメーション特徴情報の補間	179
6.2.2	位相情報を利用したアニメーションの遷移	179
6.3	アニメーションシステムと AI システムの間の情報伝達	180

7. キャラクターアニメーションと人工知能

7.1	身体と知能を持つゲームキャラクタ	183
7.2	キャラクタ制作	185
7.3	キャラクタ周りの人工知能	186
7.4	階層の具体的なシステム	188

7.4.1	アニメーションシステムと AI の多階層化	189
7.4.2	身体 of 知識表現	191
7.5	サブサンプリング・アーキテクチャ	192
7.6	意思決定とアニメーションの間の領域	194
7.6.1	1 アクションの場合	195
7.6.2	2 アクション以上の場合	196
7.6.3	物や地形を使うシステム「スマートオブジェクト」	198
7.6.4	同期アニメーション	199
7.7	ま と め	199
付	録	201
A.1	数式 of 表記	201
A.2	3次元空間における座標変換	201
A.3	3次元回転のパラメータ表現	209
引用・参考文献		218
索 引		226



本書は、ビデオゲームのようなリアルタイム CG (Computer Graphics) アプリケーションにおいてキャラクターアニメーションを生成するシステムに着目し、そのシステムを構成する技術について説明する。

リアルタイム CG アプリケーションの特徴は、映像として出力される内容がアプリケーションのユーザによる対話的な操作入力に応じて変化することにある。こういったアプリケーションの中でも、典型的なビデオゲームにおいては計算機上にモデル化された 3 次元空間に人間や動物などのキャラクタを配置し、これらのキャラクタがユーザによって直接操作されるか、あるいはプログラムによってその動きが制御される。本書の主題としているキャラクターアニメーションを生成するシステムは、キャラクタの身体各部の位置姿勢を時々刻々変化させることによって 3 次元空間内でのキャラクタの動きを作り出すために利用される。

1.1 ゲームプログラムにおけるアニメーションシステム

ゲームプログラムでは、計算機上にモデル化された 3 次元空間の物体について、その位置や姿勢を一定の時間間隔で更新した結果を画面に表示する。時間間隔はゲームの設計や表示装置の性能によって決まるが、おおむね 1/30 秒よりも短い間隔で更新し続けることにより、物体の姿勢が少しずつ変化する様子が表示装置に出力される。一回分の更新のための計算処理の時間と、この更新によって出力される画像のことをフレーム (frame) と呼ぶ。

前後するフレームにおける物体の位置姿勢は離散的に変化しているが、30 Hz や 60 Hz といった頻度で姿勢を変化させ続けることにより、人間の目には物体があたかも連続して運動しているかのように見える。これは、映画やテレビ

に映される物体が滑らかに動いているように感じられる現象とまったく同一である。

ある一つのフレームにおいて、ゲームプログラムが行う計算処理の入出力に着目すると、入力にはユーザによる操作であり、最終的な出力はその瞬間を一枚の画像として表すことによって生成される。この計算処理は、映像を出力するようなゲームプログラムにおいても実行されている。また、計算機の処理能力向上にしたがって、より複雑な計算処理が可能となり、映像の細部まで詳細な描写を行えるようになってきている。

映像を生成する処理は計算機のソフトウェアとして実装されるため、計算機の処理能力が許す限りにおいて、その内部でどのように処理を行うことも自由である。しかし実際には、あるフレームにおける計算処理をいくつもの処理機能部品（コンポーネント）の組合せとして捉えれば、映像生成の計算処理をコンポーネントへと分割する構成の方法は、多くのゲームプログラムでおおむね共通した方法が採用されている。

ゲームプログラムにおけるアニメーションシステムは、上述の計算処理における一つのコンポーネントである。アニメーションシステムを中心としてあるフレームにおける計算処理を説明すると、処理の各ステップは以下のような入出力を持つ機能として記述できる。

- (1) **ゲームロジックによるキャラクターの内部運動状態の更新** ゲームロジックは、ゲームプログラムをゲームたらしめる核となる処理である。このステップでは、ユーザの操作入力によってプレイヤーを表すキャラクターの運動状態を変化させる。また、計算機によって制御されるキャラクターは、3次元空間内でのプレイヤーとの位置関係などを考慮して行動を決定し、自らの運動状態を変化させる。後者はいわゆるゲーム AI と呼ばれるアルゴリズムに基づいて計算処理が行われる。これらの計算から、プレイヤーを含む複数のキャラクターが相互に影響を及ぼすゲームプレイが生み出される。
- (2) **アニメーションシステムによるキャラクター姿勢の更新** アニメーシヨ

ンシステムに対する入力として、各キャラクターが目標とする運動状態がゲームロジックから与えられる。このような運動状態の例としては、移動の方向や速度を上げることができる。このステップでは、キャラクターの内部状態として保持している現在の運動状態を開始点とし、与えられた運動状態を目標として内部運動状態を更新する処理を行う。また、更新した運動状態に基づいて、キャラクターの姿勢を表す関節の角度を変化させて出力する。

- (3) **描画システムによる画像の生成** 描画システムには、アニメーションシステムが更新した関節の角度が入力として与えられる。このステップでは、関節の角度によってキャラクターの位置姿勢を変化させ、その表面形状を変形させる。さらに、3次元空間内に配置されたキャラクターや物体の位置関係と、あらかじめ詳細に定義しておいた表面の色や質感、そして、空間内の光源などを考慮することにより、3次元空間がある視点からどのように見えるかを計算し、1枚の画像として出力する。

1.2 ソフトウェア機能部品としてのアニメーションシステム

ゲームプログラムを構成する処理機能部品（コンポーネント）は、その入力としてユーザの操作やほかのコンポーネントからの出力を与えられる。そして、コンポーネントに実装されたアルゴリズムに基づいて、与えられた入力に計算処理を施した結果を出力する。出力された計算結果は、別のコンポーネントの入力として利用される。

個々のコンポーネントに与えられる入力は、さまざまに変化するゲームの状態を反映している。このような入力に加えて、事前に準備しておいた静的なデータを参照し組み合わせることにより、コンポーネントにおける複雑な計算処理が可能になっている。静的なデータの代表例としては、キャラクターや物体の形状を表現する多面体データや、色や質感を表すテクスチャデータ、そして、関節角度の時間的変化を表すアニメーションデータなどを上げることができる。

コンポーネントの計算処理において静的なデータを利用する目的の一つは、データ生成の複雑な演算を事前に済ませておくことで、プログラムを実行するとき、すなわちランタイム (runtime) での計算量を減らすためである。もう一つの目的は、計算アルゴリズムとして定式化できない、高度に専門的な手作業によって生み出されるデータを格納することにある。近年では、このような静的データを作成・管理するソフトウェアとゲームプログラムを一つに統合した、ゲームエンジン (game engine) と呼ばれる統合開発環境が広く一般に利用されている。

1.3 アニメーションシステムのための静的データ

アニメーションシステムが実行時に参照するアニメーションデータは、キャラクターの全身骨格構造を構成する関節について、その位置や姿勢の変化を時間軸に沿って記録している。このようなデータの作成作業はアニメーション作成の専門的スキルを持つアニメーターによって行われ、**DCC ツール** (Digital Content Creation tool) と呼ばれるオーサリング用のソフトウェアが利用される。

アニメーションデータ作成時には、人間の演者による運動をモーションキャプチャ (motion capture) によって測定したデータを参考のために利用する場合も多い。ただし、演者とキャラクターの体節の長さの違いに起因する修正や、要求されるゲームデザインに合致させるための調整、測定時に発生したノイズの除去といった作業が必要になるため、測定したデータに手を加えないまま静的なアニメーションデータとして使われるわけではない。

静的なアニメーションデータはキャラクターの動作をそのまま記録しているため、ゲームによって必要とされる動きの性質や自然さを備えたデータを準備することができる。その一方で、体格が大きく異なるキャラクターや動きの性質の異なる運動に対しては、すべての組合せについて別々のアニメーションデータをそれぞれ準備しておく必要がある。また、静的なアニメーションデータの作成は熟練の技術者であるアニメーターによる手作業を必要とするため、制作用業

のコストが高い。さらに、ゲームプログラムで使用できる主記憶装置（メモリ）の容量は限られており、その大部分が多面体形状データやテクスチャデータに占有される。データの大きさに起因するこうした制約から、ゲーム中で行われるあらゆる動作を静的なアニメーションデータとして用意しておくことは不可能であることがわかる。

アニメーションデータの総量に対して上述のような制約がある中で、多数の短いアニメーションデータを動作の基礎部品として準備し、キャラクターの運動状態に応じてこれらを組み合わせることにより、多様な動作を作り出す方法が利用されている。このような動作部品としてのデータは、アニメーションクリップと呼ばれる。アニメーションクリップを組み合わせる場合には、もとになるデータの性質をできるだけ損なわないように配慮しつつ、あるクリップから別のクリップへと滑らかに切り替えたり、複数のクリップを混ぜ合わせるといった演算操作が行われる。

また、キャラクターが置かれる環境がさまざまに変化する場合に対応した多様なアニメーションを生成するため、静的なアニメーションデータの組合せによって生成した動きに対して、計算処理によって動的に生成したアニメーションを重畳する方法が利用されている。その例としては、ほかのキャラクターとのインタラクションに対応して姿勢を変化させる効果の生成や、キャラクターの置かれた環境から外力を受けたことによるリアクション動作の生成などが挙げられる。

以上のような静的データとアルゴリズムの組合せによるアニメーション生成の方法は、少数のアニメーションデータからキャラクターの多様な動きを生み出すことが可能であり、現在多くのゲームプログラムで共通して用いられるようになっていく。

1.4 アニメーション計算処理アルゴリズムの特徴

ゲームプログラムは、ユーザによる対話的な操作を入力として動作する。そ

索引

【あ】		【お】		【け】	
アクションステートマ シン	125	オイラー角	210	ゲイン	43
アタッチメントジョイ ント	73	オイラー法	39	ケージ	30
アニメーション圧縮	91	【か】		ゲームエンジン	4
アニメーションカーブ	10	加算ブレンドシェイプ	17	【さ】	
アニメーションキー	84	加算ポーズブレンド	109	最近傍補間法	29
アニメーションクリップ	102	慣性補間	120	座標軸	201
アニメーションシステム	96	完全非弾性衝突	161	座標変換	55
アニメーションジョイ ント	68	【き】		サブサンプシヨン・ アーキテクチャ	192
アニメーションチャン ネル	102	キー	85	散布データ補間法	24
アニメーションパラ メータ	10	木構造	65, 129	サンプリング	103
アニメーションブレンド	106	キーフレーム法	84	【し】	
アニメーションレイヤ	123	キーフレーム補間法	84	シェパード補間法	24
アフオーダンス	192	キーフレームリダク シヨン	93	四元数	212
【い】		キーポーズ	85	次元の呪い	29
イーズイン・アウト	89, 117	逆運動学	141	シミュレーション・ ベストな意思決定	198
位相	133	逆バインドポーズ	59	自由形状変形	30
インバースキネマティ クス	141	キャラクター AI	186	順運動学	75
【う】		キャンディラッパー現象	38	ジョイント	34
運動方程式	157	球面線形補間	88	ジョイント階層構造	65
【え】		行列バレット	60	ジョイント座標系	54
エフェクタ	65	【く】		ジョイント姿勢	54
エルボー破綻現象	38	空気抵抗	162	しらみつぶし探索法	138
		クォータニオン	212	自律型エージェント	184
		区分線形キーフレーム 補間法	86	心身問題	185
		区分線形補間法	22	身体の知識表現	191
		クリップ	102	ジンバルロック	211
		グローバル座標系	202	【す】	
		クロスフェード	116	スキニング	34

スキン	34	動径基底関数補間	26	不足制動	166
スキンウェイト	34, 57	同次座標系	205	物理シミュレーション	38
スキンモデル	34	同次変換行列	206	ブラックボード	181
スクリプティッド AI	184	動摩擦係数	161	フリーズトランジション	118
スケルトン	36, 62	動力学	156	フレーム	1
ステート	125	トランジション	114	ブレンドウェイト	13
ステートパラメータ	131	トランジションポイント	132	ブレンドシェイプ	13
ステートマシン	98, 125	ドリフト	41	ブレンドツリー	98, 129
スプライン補間法	90			ブレンドトランジション	115
スマートオブジェクト	198	【な】		プロシージャル制御	71
スマートトレイン	199	ナビゲーション AI	186	【へ】	
スマートロケーション	199			ベジェ補間法	90
【せ】		【に】		ベルレ積分	40, 163
静止摩擦係数	161	任意軸周りの回転表現	211	【ほ】	
線形クロスフェード	117	【の】		補間	16
線形ブレンドシェイプ	16	ノン・プレイヤ・キャラクタ	183	ポーズ距離	134
線形ブレンドスキニング	53	【は】		ポーズスペース	20
線形ポーズブレンド	107	ハイトフィールド	172	ポーズスペース変形法	20
【た】		ハイトマップ	172	ポーズブレンド	107
ターゲット	13	バイリニア補間法	111	ボーン	62
ターゲットシェイプ	13	バインディング	34	補間トランジション	121
多階層化	189	バインドポーズ	59	補助ジョイント	71
単位クォータニオン	212	パーティクル IK	153	【ま】	
弾性衝突	161	バネ・マス・ダンパモデル	42	摩擦力	161
【ち】		パラメトリックポーズ		【み】	
知識表現	189	ブレンド	111	右手座標系	202
チャンネル	102	バーンスタイン基底	30	【め】	
頂点アニメーション	11	ハンドル	45	メタ AI	186
直交座標系	202	反発係数	161	【も】	
【つ】		【ひ】		モーションキャプチャ	4
通知システム	174	左手座標系	202	モーショングラフ	134
通知データ	174	比例微分制御	43	モーションマッチング	138
通知の優先度	176	【ふ】		モーフイング	16
【て】		フィードバックゲイン	43	【や】	
デカルト座標系	202	フィードバック制御	43	ヤコビ行列	150
【と】		フォワードキネマティクス	75, 80		
同期アニメーション	199	副次ジョイント	71		

	【ゆ】	リグ	45		【ろ】		
		量子化	92				
有限状態機械	125	臨界減衰	166		ローカル行列	78	
有限要素法	48			【る】	ローカル座標変換	76	
ユーザエクスペリエンス	188				ローカル姿勢	76	
	【ら】	ルート	63, 68		【わ】		
		ループアニメーション	105		ワールド行列	80	
ラグドール	156	ルンゲ・クッタ法	39		ワールド座標系	202	
ランタイム	4			【れ】	ワールド座標変換	55	
	【り】	レイキャスト	167		ワールド姿勢	56	
リギング	45	連続の交差検出	168				

	【A】	FFD 制御点	31	PSD 法	20
axis-angle 表現	211	FK	75	【Q】	
A ポーズ	59			QLERP	87
	【C】	IK	142	【R】	
CCD IK	152			RBF 補間法	26
C^n 連続性	101	【K】		【S】	
	【D】	KNN 補間法	28	SLERP	88
DCC ツール	4	K 最近傍補間法	28	【T】	
	【F】			T ポーズ	59
FABRIK	153	【L】			
FFD	30	LERP	85		
		【P】			
		PD 制御	43, 165		

—— 著者略歴 ——

向井 智彦（むかい ともひこ）

- 1999年 佐世保工業高等専門学校電子制御工学科卒業
- 2001年 豊橋技術科学大学工学部情報工学課程卒業
- 2003年 豊橋技術科学大学大学院工学研究科修士課程修了（情報工学専攻）
- 2006年 豊橋技術科学大学大学院工学研究科博士後期課程修了（電子・情報工学専攻）
博士（工学）
- 2006年 豊橋技術科学大学助教
- 2009年 株式会社スクウェア・エニックス勤務
- 2014年 東海大学専任講師
- 2017年 東海大学准教授
- 2018年 首都大学東京（現 東京都立大学）准教授
現在に至る

川地 克明（かわち かつあき）

- 1996年 東京大学工学部精密機械工学科卒業
- 1998年 東京大学大学院工学系研究科修士課程修了（精密機械工学専攻）
- 2002年 東京大学大学院工学系研究科博士課程修了（精密機械工学専攻）
博士（工学）
- 2002年 独立行政法人産業技術総合研究所勤務
- 2011年 株式会社スクウェア・エニックス勤務
現在に至る

三宅 陽一郎（みやけ よういちろう）

- 1999年 京都大学総合人間学部基礎科学科卒業
- 2001年 大阪大学大学院理学系研究科修士課程修了（物理学専攻）
- 2004年 東京大学大学院工学系研究科博士課程単位取得満期退学（電気工学専攻）
- 2004年 株式会社フロム・ソフトウェア勤務
- 2011年 株式会社スクウェア・エニックス勤務
現在に至る

キャラクタアニメーションの数理とシステム

—3次元ゲームにおける身体運動生成と人工知能—

Mathematical Models and Systems for Character Animation

—Motion Synthesis and Artificial Intelligence in 3D Videogames—

© Tomohiko Mukai, Katsuaki Kawachi, Youichiro Miyake 2020

2020年8月6日 初版第1刷発行



検印省略

著者 向井智彦
川地克明
三宅陽一郎
発行者 株式会社 コロナ社
代表者 牛来真也
印刷所 三美印刷株式会社
製本所 株式会社 グリーン

112-0011 東京都文京区千石 4-46-10

発行所 株式会社 コロナ社
CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替 00140-8-14844 · 電話 (03) 3941-3131(代)

ホームページ <https://www.coronasha.co.jp>

ISBN 978-4-339-02909-3 C3055 Printed in Japan

(新井)



<出版者著作権管理機構 委託出版物>

本書の無断複製は著作権法上での例外を除き禁じられています。複製される場合は、そのつど事前に、出版者著作権管理機構（電話 03-5244-5088, FAX 03-5244-5089, e-mail: info@jcopy.or.jp）の許諾を得てください。

本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めていません。落丁・乱丁はお取替えいたします。