

デジタルビデオカメラによるモーションキャプチャーシステム

Motion capture by the digital camcorder

安本 匡佑¹⁾ 佐藤 大介²⁾
Masasuke Yasumoto Daisuke Satou

- 1) 公立はこだて未来大学 システム情報科学部 情報アーキテクチャ学科
(m1200082@fun.ac.jp)
2) 公立はこだて未来大学 システム情報科学部 情報アーキテクチャ学科
(1200061@fun.ac.jp)

ABSTRACT. Up to this time, the motion capture system is very expensive one for professional. But, this system of this project, this is not expensive and easy to use for computer graphics creator. Because this system attach greater importance than to correctness to the flow of a motion.

1. 背景

近年、一部のプロ向けの技術であったCGが一般の人にも手が届くくらい身近になり、さまざまなCGを製作するためのソフトウェアが安価で発売され始めた。私も自分の中の世界を表現するツールとしてCGというものを始めたのが3,4年前になる。今ではいくつかの賞を受賞するまでになったが、CGで人のアニメーションを製作するには非常に困難が付きまとう。それは人をCGでアニメーション化する際にいかにして人間らしい動きを再現するかということだ。人は機械や架空の生物と違い普段われわれが日常的に目にするもののため少しでも不自然な箇所があると非常に目立つ。このようなことをふくめ、問題を解決するために、プロ向けの機材としてモーションキャプチャーシステムというものが存在する。これは光学式と機械式に大きく分類されるのだが、要は人の動きを数値化してコンピューターに取り込み、自分で製作したCGのキャラクターに対して適応するものだ。これは非常にすばらしい機材であり、CGでアニメーションを製作する人間としてはぜひとも手に入れたい機材だが、その価格は数千万というとても個人では手に入るものではない。かといってここであきらめてしまいたくなかったため自分で安価に個人ユーザーの視点に立った新しいモーションキャプチャーを作ろうということでのプロジェクトはスタートした。

さて実際に製作する上で選択しなければならないのは機械式を作るか光学式を作るかということなのだが、機械式とは全身に重厚なセンサーを取り付け、それによってデータを取得するものなのだが、これは機械の開発がメインとなり自分には向いていない。そこでソフトウェアの製作がメインであろうと思われる光学式を製作することにした。これは体の各部にマーカーを取り付けそれを撮影し画像を解析することによって動きを取得するものだ。

2. 目的

本プロジェクトの目的とすることは、市販のモーシ

ョンキャプチャのように高機能、高精度のものではなくクリエイターがCGを製作する上において必要となる人のうごきのおおまかな流れを捉え、その情報を使いやすい形で出力することにある。また、従来の高価格なものではなくあくまで個人用途にでも使える価格の抑えたものを作ることにある。具体的な説明は後述の概要にまかせる。

この目的としているところはいかにして、人の動きを数値化し、加工し、使いやすい情報にするかである。最終的にこのシステムで得られた情報は市販のCG用ソフトウェアなどにとりこみ、その後の編集も容易な状態となっていることを目標としている。これは従来のものと違ういくつかの点がある。ひとつは専用のCGソフトウェア以外の一般のソフトウェアでモーションデータを活用できること。これはプラグインソフトウェアの開発によりこのシステムで得られた情報を活用できるからである。さらにもうひとつの点は従来の問題点であるノイズを軽減しキーフレームを取得することにより編集しやすい情報を得られるということである。従来のシステムはより正確に数ミリ単位、より細かに数十分の1秒単位で位置を取得していたのがこれでは後の編集作業量は膨大になってしまう。そもそも個人用途ということで開発しているため、このシステムを身につけ演技するひとまたプロではない。またどんなにうまく演技しようとしてもそれがクリエイターの求める理想とする個性やしぐさをともなった動きを得ることは不可能であり、また人に不可能な動きを要求された場合はなすすべがなくなってしまう。このようなことからこのシステムでは大まかな動きを取り入れ人間らしさを含んだ大まかな情報を受け渡すに過ぎない、これはクリエイターにとっては非常に扱いやすく重宝するものである。

3. 概要

このプロジェクトで開発したのは、人に装着しその動きを示すためのマーカーとそれを装着するための衣服のハードウェア部分と、デジタルビデオカメラで撮影して保存したファイルを解析してモーションデータ

を出力するためのソフトウェアの部分である。

1. DV からの映像キャプチャー機能

二台のデジタルビデオカメラを使い、マーカーを装着した演技者の動きを撮影し、そのデータをコンピュータ上に二つの動画ファイルとして保存する。これは当初リアルタイムでの処理を考えていたためファイルを保存することなく直接処理を行っていたのだが、マーカーの数を増やすにつれ処理速度の問題が出てきたため、リアルタイム性を捨てこのようにいったんファイルとして保存してから後で読み込み処理を行うようにした。

2. 映像の自動レベル補正機能

暗闇でデジタルビデオカメラを使い撮影した場合必ずといっていいほどノイズが入る。また、コンピュータへ出力の際 DV フォーマットに圧縮されてしまうため、映像に乱れが生じることもあり、このノイズがマーカーの光と誤認することを防ぐ意味で画像の状態を自動で判断し適切なレベル補正をかけるようにした。

3. 映像からのマーカーの座標抽出機能

・マーカーの特定機能

現段階で実装しているモーションキャプチャー機能は片腕の動きと全身の動きの二つである。片腕のモーションキャプチャーは開発の実験として作ったものだがこちらに関してはほぼ完璧にマーカーの特定を行うことができる。さらに全身の動きに関してだが、こちらはマーカーの数が増えているため完全にマーカーの特定を行うことが困難であり、こちらにあらかじめ想定していない動きをされてしまった場合にマーカーを正しく特定できないという問題が生じているのだが、それ以外はほぼマーカーを特定することができる。

・人間の可能な動きの判別機能

マーカーの特定の際に人間の可能な動きの判別を行い、それをもとにマーカーを特定できるようにした。これによりより正確なマーカーの特定が可能となった。

4. 動きの流れを捉えるキーフレーム抽出機能

マーカーの特定後に、前のフレームのマーカーの位置とその前のフレームのマーカーの位置を用いてすべてのフレームから動きの変化の度合いを求めることにより、必要ないと判断されたフレームからマーカーの動きの情報を除去し、キーフレームを抽出する機能を実装した。これにより、より滑らかで、編集のしやすいモーションデータを得ることができる。

5. X ファイル独自形式のモーションデータ出力機能

すべての処理が終了した後に、オブジェクトデータとモーションデータをもった X ファイル形式のデータファイルを出力する機能を実装した。

6. 外部で作成した任意のオブジェクトデータとモーションデータの融合機能

モーションキャプチャーシステムで得たモーションデータと特定の条件を満たした任意のオブジェクトファイルを読み込み、それらを融合しひとつのファイルとする機能を実装した。ただ、どのようなオブジェクトファイルでも読み込めるわけではなく、特定の順番と名前でボーンを組み込んだオブジェクトファイルでなければ、モーションデータと融合したところで正常に表示することはできない。

7. 得られたデータの表示および簡易編集機能

得られたデータの表示に関して外部の X ファイルビューワーを利用することによる表示できますが、X ファイルの簡易編集機能を実装することは後に外部の CG 用アプリケーションに渡すことを考えるとあまり意味がないと判断したため、かわりにデジタルビデオカメラで撮影した映像を保存したファイルからデータを解析する際に、どのフレームを解析するかを指定できる機能を実装した。さらにアプリケーション上からさまざまな解析の際に必要なパラメータを変更できるようにした。

8. LW でのモーションデータの取り込み機能

これに関しては開発に余裕があれば開発しようとしていたのだが、現段階ではまったくの未実装である。

9. マーカー、フレーム

今回の開発で一番時間がかかってしまい、ソフトウェアの開発がおろそかになってしまった原因とも言えるマーカーの作成だが、デジタルビデオカメラで撮影してどう見えるかなどをソフトウェアともども修正しなければいけなかった部分であるため、数々の試行錯誤を重ね発光ダイオードによる小型のマーカーを作成した。だが、現状ではまだ問題点がいくつか残っている。

フレームに関してはデジタルビデオカメラを固定するという役割だったのだが市販の三脚で代用が利くため開発を行ってはいない。

4. 開発内容

・設計・実装

このシステムの大まかな流れを下記に示す

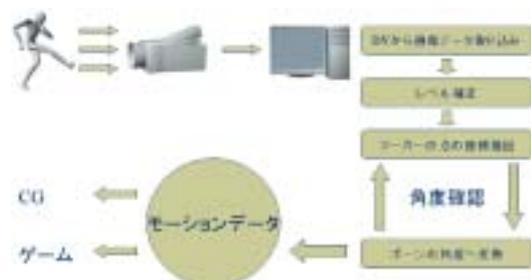


図1 システム概略図

上記の図を参照にし、おおまかには DV にとりこむまでのハードウェアの部分とそれ以降モーションデータを出力するソフトウェアの処理の部分に分かれる。

1. ハードウェアの処理

ここで言うハードウェアとはマーカーやそれを装着するための衣服をさす。この シス

テムは暗闇の中で発光ダイオードを使ったマーカーを装着し、それを撮影することがハードウェアの主な役割である。ここで重要なのが、あとでこの撮影した映像からいかにしてマーカーを特定するかということだ。この処理はマーカーの数が最終的に 24 個になることを考えるといかにして処理を軽くするかということに重点が置かれる。24 のマーカーがすべて一色で判別がつかない場合、非常の多くの可能性が考えられるため、処理速度が落ちるだけでなく、正確性も減少する。そこで発光ダイオードというものの特性を生かし、色によってマーカーの種類を特定することにした。詳細は次のようになる

- ・赤：両腕の手の甲、手首、肘、肩に赤色のマーカーを装着する。

- ・緑：右足の足の甲、足首、膝、腰に緑色のマーカーを装着する

- ・青：左足の足の甲、足首、膝、腰に青色のマーカーを装着する

- ・白：体の中心線上の、腰、胸、首下顎、額に白のマーカーを装着する

最初の案では白のマーカーをさらに二つに分け、白と黄色を想定していたのだが、DV で撮影したところ、黄色は赤と誤認しやすく、さらに緑が黄色とに見えたため黄色を削除し白としたのである。

では実際これらの色をした発光ダイオードを用いたマーカーの作成なのだが、これには非常にマーカーの選別に労力を費やした。なぜなら、人が演技するためどの角度からもなるべく見やすい発光ダイオードを使う必要がある、そのため視野角の狭いものは使えない。また、輝度が高すぎると白と誤認しやすく、さらに輝度が低すぎるとノイズとして除去してしまう。これらの条件にあった発光ダイオードを探したのだが、赤は容易に見つかったのだが、白や青は非常に数が少なく、選定に苦労を要した。

これらのはこうダイオードは成るべく人の演技の妨げにならぬよう小型化する必要があった。そのためボタン電池を用いることにした。

最後にこれらはいかにして人に装着するかなのだが当初はビニールテープを用いて人に巻きつけていたのだが、使い捨てになっってしまう上に演技者にも評判が悪かったため、マジックテープを使ったものになった。さらにあらかじめマジックテープのもう一方を服に貼り付けておくことで、いつでも取替えの聞くものに完成させた。



図 2 マーカーを装着した人

上の図は実際にこのマーカーを装着し演技をしてもらったときの図である。

このように準備を整え、いざ DV で撮影したのだが、ここに来て大きな問題点が発覚した。あらかじめ外部の CG ソフトを使い作成したマーカーの映像ファイルを元にシュミレートし作成した解析のためのプログラムにかけたところ、うまく判別できなかった。

この原因は以下のように考えられた

1. マーカーが小さすぎるため、DV で撮影した際につぶれてしまう。
2. DV の圧縮で色が分解されてしまう。
3. 色が判別しにくい

ここにきて DV の更なる問題点が発覚してしまったわけだが、これを放置しておくこともできないため、解決策を考えていたところ、竹内 PM の方からの提案もありいくつか解決策を見出した。

要は色が判別できるように DV に撮影されればよいので、光源を何かで覆いその光を撮影するというのだ。これは実験の際に、輝度が強すぎて白に見えてしまうマーカーにティッシュペーパーをかぶせたところ、マーカーの光が大きくなりばやけさらに白ではなくそのマーカーの色を正しく認識できるようになったところからも考え付いたのだが、同様に何かでこのマーカーを覆えばいいということだ。

これらのことを元にこれからの課題として開発を進めて生きたい部分である。

2. ソフトウェアの処理

上述のハードウェアの処理を経て、その映像を元にソフトウェアにて解析を行うのだが、役割を分担して開発していたこともあり、ひとつ重要な機能を実装した。それはハードウェアの処理を経て得られた映像ファイルは一旦ファイルとして保存され、ここで一旦処理を中断する。その後、会席用のプログラム

を呼び出したときにこの保存されたファイルを読み込み解析を解するのである。このことにより、同じ映像ファイルを何度もパラメータを変え解析できるとともに、ハードウェアと切り離して開発化が可能となった。

では実際にソフトウェアの処理の過程である

1. 二台の DV から映像を保存する
2. 保存されたファイルを読み込む
3. 正面の映像のファイルから XY 座標を検出
4. 横の映像のファイルから YZ 座標を検出
5. 得られた座標からマーカー特定
6. ボーンへの変換
7. X ファイルの出力

以上のようになっている。

1. 二台の DV からの映像を保存する
これは DirectX を利用することによりハードウェアに依存せずに DV の操作を行い指定された場所に指定された名前で映像を保存する。映像はそのフレームごとの明るさを判断し、ノイズを除去する
2. 保存されたファイルを読み込む
保存された二つのファイルを読み込み、それをフレームごとに処理するために呼び出されるごとにそのフレームの RGB 成分を配列に代入する。
3. 正面映像のファイルからの XY 座標の検出
映像の中のマーカーは必ずしも一点ではない。これは DV の特性上画像のサイズが 720*480 になり、この 1*1 にひとつのマーカーが収まらないということである。そのため、まずあらかじめ一度処理するフレームの配列を呼び出し、1*1 にマーカーの一転が収まるように処理をする。この処理が終わった後に RGBW のマーカーそれぞれを左上から順番に XY 座標を記録する。
4. 横の映像ファイルからの YZ 座標検出
これに関しては上記の 3. と同様にマーカーのそれぞれの YZ 座標記録する。
5. 得られた座標からのマーカーの特定
この処理はこのプログラムのメインとなる部分である。3.4. の処理によってそれぞれのマーカーの XY 座標 YZ 座標を記録したのだがどの尾のマーカーがどのマーカーを表しているのかがわかっていない。まずはここで Y 座標の差から対応する XZ 座標を求め、ばらばらであった XY 座標、YZ 座標をひとつの XYZ 座標を持ったマーカーの座標とする。ここで場合わけに入る
 1. 24 すべてのマーカーが取れているかどうか
 2. 8 つの赤マーカーがすべて取れているかどうか
 3. 4 つの緑マーカーがすべて取れているかどうか
 4. 4 つの青マーカーがすべて取れて

いるかどうか

5. 8 つの白マーカーがすべて取れているかどうか

これらの場合わけの後にさらにその成分のマーカーのいちから想定される動きのパターンとあてはめマーカーの特定を行う。個々ですべてのパターンを記述するのは煩雑になる恐れがあるため、大体の流れを記述する
処理の順番は G から行う

G の四つの成分のうち Y 座標の高い順番に G 成分を並び替え、G 成分の Y 座標の差と一番 Y 成分の大きいものと次に大きいもの間の距離をもとめ、それがどの程度の角度なのかを算出する。次に 2 番目と 3 番目も同様の処理を行い。これらの間の距離が同じくらいの場合、脚と判断し 3 番目、4 番目は足首と脚の甲のどちらかを表すと判断できる。仮にこのようなケースに当てはまると判断されれば 2 番目 3 番目のベクトルもしくは 2 番目 4 番目のベクトルのなす角を計算し、人間に可能な動きであるかどうかの判断から緑のマーカーを特定する。同様にして青も特定が終われば腰を表す青と緑のマーカーと肩を表す赤のマーカーは必ず平行四辺形をなすということを利用し、さらに青と緑の位置から体がどの向きを向いているかを求める。これにより白のマーカーを Y 座標の大きいものから順番に体の向きのベクトルと同じになるようにすれば白のマーカーが特定できる。その後赤の肩をなすマーカーが特定できれば残った 6 点の赤は、それぞれの間の距離を計算する。もっとも短いと判断されたのが二箇所あればそれは手首と手の甲を表していると判断され、それ以上の場合が手が交差していると考えられるため、この場合は判断できる場合は少なく、誤認してしまう恐れがあるため、前後のフレームに処理を任せこのフレームでの特定は除去する。さらに残った二点は肩と手首のマーカーから存在しえるかどうかを角度を計算することにより特定する。
このようにマーカーの取得数とその位置からあらかじめいくつものパターンを用意しそれに当てはめることによりマーカーを特定していく。この際に用いる重要な要素はある程度見切りをつけた後に人がとりうる角度なのかどうかを判断することによってマーカーを特定するという処理だ。

6. ボーンへの変換

得られた特定されたマーカーの情報を元にボーンの骨組みに合うように、ボーンを示す終点と始点のマーカーの位置からボーンの角度を算出する。もしボーンが取得できなかった場合はそれより子供に当たるボーンの処理は行わない。こうすることにより紛らわしいノイズを除去す

ることが可能となる。

7. X ファイルの出力

ボーン情報をすべてのフレームごとに取得すれば後はこれらをアニメーションデータとして、外部からの特定の条件を満たしたオブジェクトファイルを読み込み二つを結合し出力する。ここでの特定の条件というのが重要となるのだが、これはボーンがこちらの指定した順番通りに構成されているかどうかということである。この順番が違えばたとえば腕を動かしているはずの情報で足を動かしてしまったりするため。きっちりとあわせる必要があるのである。

3. テスト

ここでは片腕を使った実験の結果を示す



図3 マーカーを装着した腕

まずは実際に作成されたマーカーを体に装着した。



図4 マーカーを装着し部屋の照明を落とした

次にこれはマーカーを装着した演技者のマーカーを特定しやすいようにするため部屋の電気を切り、暗闇を作ったのである。この状態で演技者には演技をしてもらう。演技が始まったところでキャプチャー開始のボタンをおし、終わったところで終了させる。

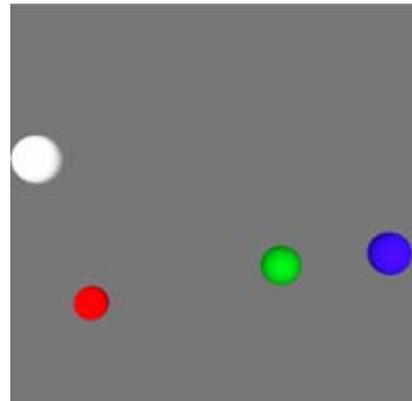


図5 解析後マーカーをCGの球体におきかえた

得られたファイルを解析し、解析が終了したとここで特定されたマーカーを実際のマーカーに対応する色のオブジェクトとして出力した結果である。

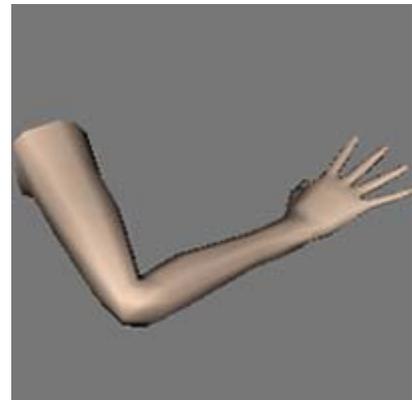


図6 ボーンへの変換後外部で制作した腕のCGにモーション情報を適用した

特定されたマーカーの情報からボーンの角度を算出し、実際に外部のオブジェクトをよみこみ出力した結果、上図のようにコンピュータ上で人の動きとして再現ができたのである。

5. 現状の問題点

1. 正確なものより動きの流れを重視し、取得不可能と判断したり、紛らわしいと判断したときはデータと取得しないため。編集しやすいデータを取れる反面、その間の部分を再生の際に補完してしまうので、実際とは違った動きをしてしまうことがある。これは最終的にCGの編集用ソフトウェアに出力したときには編集で簡単に直せる部分であるため、あまり問題にはならないのだが、これを修正するとなると見えない点を減らす必要があるためDVの数を増やす必要がある。
2. 今の開発段階ではかなり特定されたXファイルのオブジェクトファイルでなければボーンに変換したアニメーションデータを受理できないということである。これも外部のCGソフトウェアに出力できるようにして、その中のオブジェクトに適用できるのであれば問題ないのだが、現状ではそのプラグインソフトウェアの開発は計画しているものの、開始されていない。もっと柔軟

な対応のできるように改良の余地はある。

3. これはハードウェアの部分になるのだが、現状の DV では光をそのままきれいに捉えることは難しい。つまり、目で見ているものと DV を通してみるものでは色が違ってくるといふことだ。これはより性能のよい DV を使えば解決する問題なのかもしれないのだが、低価格で実現するということとは相反する、発光ダイオードを用いないメーカーを利用することも考え今後の課題としていきたい。

6. 今後の課題、展望

今後はまずこのソフトウェアの完成度を上げること、ハードウェアとの兼ね合いによる問題点を解決すること、プラグインソフトウェアの開発である。まず、このソフトの完成度に関しては、現状での問題点を解決し、さらに動きのパターンを増やすことにより多くの動きに対応すること、さらにはより多くのオブジェクトファイルを柔軟に使えるようにすることである。ハードウェアとの兼ね合いとは、このプロジェクトの大きな壁であった。すべてソフトウェア上でシュミレーションとして動かした場合はうまく動いてもハードウェアを使い実際に現実世界を撮影したりするとどうしても予測ではうまく動いてた部分が動かなくなったりとかなりの問題が生じてしまう。さらにはプラグイン

ソフトウェアの開発だがこれはこのプログラムだけではあまり意味を成さないということを感じた、つまりこのソフトの狙いがそもそも編集しやすいデータを作るといふことにある。そのためいくら編集しやすいデータを作ったところで編集することができる環境がなければ何の意味もない、ほとんどすべての場合においてモーションキャプチャシステムというのは後で編集してこそ有用なデータとなりうるのである。そのことを念頭において、今後はプラグインソフトウェアの開発にも取り組んでいきたい。

さて、これからの展望なのだが、まずはこのソフトを納得のいくまで仕上げ、プラグインソフトの開発も終わったとすれば、どんどんこのシステムを宣伝し、より多くの人に個人用のモーションキャプチャシステムとして使ってほしい。そうすることにより、自分の興味のある分野である CG アニメーションという世界の技術の発展貢献できれば幸いであるし、なによりもより手軽に自由な発想を具現化することができるようになることはクリエイターとして 100%制作に力を注げよりすばらしい作品を生み出すことができるだろう。

4. 参加企業及び機関

5. 参考文献