

概念表現から映像を構成するシステム及び行為の構造についての考察

A System for Constructing Animated Movie from Conceptual Representation and a Consideration on the Structure of Actions

富手 瞬^{*1}
Shun Tomite

小方 孝^{*2}
Takashi Ogata

花田 健自^{*2}
Kenji Hanata

^{*1} SANKYO
Sankyo

^{*2} 岩手県立大学
Iwate Prefectural University

As visual representation in narrative generation project, our goal is to develop a mechanism for translating narrative conceptual representation into animated movie automatically. This research is one of the first attempts and we implement a system to generate a sequence of rough CG representation from a set of given narrative conceptual representation. To make a conceptual representation as the system's input information, we analyze all shots in "Tokyo Story" by Ozu and acquire detailed data about actions. We succeed to translate Ozu based conceptual representation to animated movie by TVML, a text based CG descriptive tool. Here the most essential problem is how to combine more abstract actions in conceptual representation and more concrete actions to be expressed by animated movie. In the current system, although each action is described as a script of body movement, truly it is necessary to construct a hierarchical actions system.

1. まえがき 位置づけと目的

筆者らの物語生成システム([小方 1996],[小方 2003ab])の中で、物語内容や物語言説は概念表現として記述されるが、そこから自動的に映像表現化を行うための研究を進めて来た[小方 2008]. 求めるのは映像自体の精緻さや自然さではなく、物語的な映像の連続の生成であるが、そこで問題となるのは、概念表現における抽象的な行為を表層的な映像表現の具体的な行為(動作やその系列)に変換する作業である. 最終的には行為を巡る体系的記述, 行為オントロジーが必要となるのであるが、本研究では、この問題を具体的なシステムを構築することで考察することを目指す. 物語の概念表現の系列から極めて粗野であるが連続する映像を生成する映像構成システムを構築し、小津安二郎の映画『東京物語』における画面ごとの行為を詳細に分析して概念表現化し、それを映像構成システムを使ってTVML スクリプトに自動変換した例を示す. これらの作業を通じて、上記の問題、すなわち抽象的な概念表現を具体的な映像表現に架橋する方法について考察する.

2. 映像構成システムの構成

本研究で試作した自動映像構成システム(以下、システムと呼ぶ)の概要について述べる. システムの構成図を図1に示す. これは[小方 2007]及び[小方 2008]のアイデアをもとに今回新たに実装したものである.



図1: 映像構成システムの構成図

システムの入力データは、以下のような形式で表される物語の概念表現記述である ((話す (agent 周吉) (counter-agent とみ) (instrument nil) (object nil) (location 平山家_居間) (goal nil) (from nil) (time 朝) (narration \"これじゃとおおさかるくじじゃなあ\") (caption \"これじゃと、大阪6時じゃなあ\"))). これは、「周吉がとみに、朝、平山家の居間で、「これじゃと、大阪6時じゃなあ」と話す」ということを意味している.

システムは、この概念表現記述を、以下の図2のようなTVML(TV program Making Language)形式に変換する. TVMLとはテレビ番組を制作することを目的としてNHKが開発したテキストベースのツールであり、専用プレーヤーで再生することにより、映像化することが出来る. 本研究では映像化のためにこのスクリプトファイルを使用する.

```

reset()
skipscript(switch=on)
//キャスト定義
character: casting(name= 周吉)
character: openmodel(modelname=mattu , filename="mattu¥mattu.bm")
character: bindmodel(name= 周吉, modelname=mattu)
character: setvoice(name= 周吉, voicetype="male01")
character: casting(name= とみ)
character: openmodel(modelname=cherry ,
filename="cherry¥cherry.bm")
character: bindmodel(name= とみ, modelname=cherry)
character: setvoice(name= とみ, voicetype="female01")
    
```

図2: TVML形式

システムの処理手順の概略であるが、まず概念表現を読み込み、入力データに登場する全てのキャスト・オブジェクト・ロケーションモデルを定義する. 定義には定義関数ごとに中身の異なるふたつのデータベースを使用する. 例としてキャストを定義する場合、まず agent もしくは counter-agent の名前をキーとして人物データベースからそのキャストの種別・年齢・性別を取得し、取得された情報をキーとして CG キャストデータベースからキャスト情報(TVML スクリプト)を探す. その後、入力データを

シーンごとにリスト化する。シーンは入力データにおける location と time から判別する。最後に、リスト化したシーンのデータ内に登場するキャスト・オブジェクト・ロケーションをリストアップし、最初に定義しておいたそれらの CG モデルスクリプトと対応させ、動作・時間(照明)・配置のスクリプトと共に TVML として書き出す。

3. 映画の分析

『東京物語』全編の概念表現をもとに上記システムで TVML スクリプトへの自動変換を行うことを目標として、映画分析を試みた。特に、映像作品には身体動作と結びついた具体的な行為(動作)が現れるため、概念表現における抽象的な行為との相互関連性の考察に有益である。『東京物語』は基本的に日常生活の場面から出来上がっており、あまり複雑な行為が現れないところから、基礎的な考察を行いやすいとも考えた。

『東京物語』を分析する際、「シーン票」と「動作票」の二種類の票を用いた。それぞれの例を図3、図4に示す。

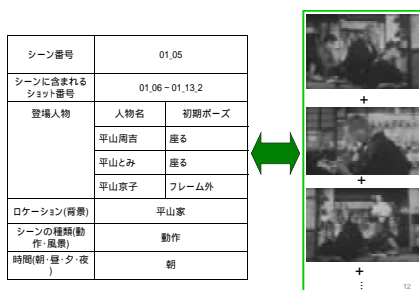


図3: シーン票と映画内のショット

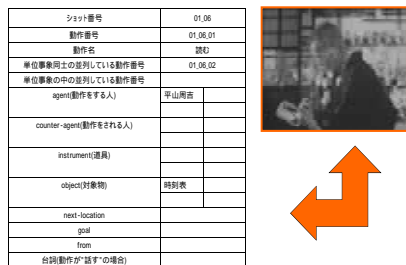


図4: 動作票と映画内のショット

シーン票には映画内のシーンごとの情報を記述する。この映画では屋内のシーンが多かったため、同じ屋内であっても、部屋や廊下単位で判別を行った。シーン票はシーン番号、シーンに含まれるショット番号、登場人物、ロケーション名、シーンの種類、時間の諸項目から成る。シーン番号は、そのシーンの映画内におけるシーンの通し番号であり、ショット番号は、映画内のショットの通し番号である。このうちロケーション名と時間は概念表現(図8に例を示す)における location と time と対応する。

一方、動作票とは登場人物の動作情報を記述するものであり、シーン票内の「シーンの種類」が「動作」であった場合に使用する。また、原則として映画内のショット単位で分析をしていく。動作票の構成要素はショット番号、動作番号、動作名、単位事象同士の並列している動作番号、単位事象の中の並列している動作番号に加え、agent, counter-agent, instrument, object, goal, from, 台詞から成る。動作番号とは、ショット内でのその動作が何番目に行われた動作か記述するものである。ショット・動

作番号以外の要素は概念表現の要素とそれぞれ対応する。動作は動詞の基本形で記述すると同時に、映画内に出てくる動作を見逃さないよう細部に渡って記述した。また、カメラワークによって登場人物が現れたり消えたりした場合は、「(*現れる)」または「(*見えなくなる)」と記述した。

単位事象とは一人ないし複数人が一緒に行っている行為である。「単位事象の中の並列」とは、「Aさんが話しながら読む」のように、「~しながら~する」という形の動作であり、「単位事象同士の並列」は「Aさんが話している後ろでBさんが掃除をしている」のような事象の並列的な動作を表す。

4. システムの実行 概念表現と映像表現をつなぐ方法

2節で述べたシステムと3節で述べた分析票を用いて、『東京物語』を概念表現化し、映像に自動変換する実験を行った。システムの入力となる概念表現は、図5に例を示すように、シーン票と動作票から入力データの形式に沿ってそれぞれの要素を動作ごとに記述した。動作は動作番号が若い順に記述していき、映画内で動作が行われた順番に直列な動作として扱う。使用する動作はあらかじめ、TVML による動作記述を視覚的に見えるツールとして[真部 2007ab]によって開発された Action Script Maker を用いて作り込み、データベースに登録した。



図5: シーン票と動作票からの概念表現の作成

行為の多くは身体部位と関連しており、以下のような分類を行うことができた。「頭」を動かす行為: うなだれる, うなづく, 見回す, 視線をそらすなど / 「腕(手も含む)」を動かす行為: Ex. 拾う, 触る, むしる, ノックする, 叩くなど / 「腰」を動かす行為: 屈む, 起きる, 身を乗り出すなど / 「足」を動かす行為: 足を振る, 足を伸ばす, 歩こうとするなど。行為によっては、これらの部位の組み合わせによって成り立つものがある。例えば、寄りかかる

腰を後ろに曲げる + 両腕を後ろに下げる, など。身体動作と密着した動作を(相対的に)基本的な動作と考えることができ、その組合せによって様々の複合的な動作を作ることができると思われる。

システムは、表1に示すような行為と身体動作の対応関係をデータベースとして保持する。

表1: 行為と身体動作の対応関係

行為名	身体動作
扇ぐ	右肘を曲げる 右腕を内側へ持っていき 右腕全体を外側へ動かす

アコーディオンを弾く	両腕を外側へ上げる 両肘を曲げる 左肘を外側へ動かす 右肘を外側へ動かす
開ける	右腕を上へあげる 右腕を内側へと動かす
あごで指す	頭部を上方向へ少し曲げる
汗を拭く	右肘を曲げる 右腕を顔の横まで上げる 右腕を少し上下に動かす
温める	右手を軽くあげる 右掌を前に向ける 右手首を軽く右に動かす
頭を掻く	右肘を曲げる 右腕を上げる 頭部に右手を当てる
頭をかしく	右腕を上へあげる 右肘を曲げる 手首を内側に返す 頭を右側へ傾ける 右手首を内側へ2回返す
頭を下げる	頭部を下方向へやや曲げる
あてる	右腕を前に少し上げる 右手首を反らす
屈む	腰を曲げる
歩こうとする	左足を前に出す 右足を後ろに出す
いじる	両腕を前へ上げる 両手首を内側に曲げる
移動させる	両腕を右側に上げる 両腕を左側へ動かす
動かす	両腕を右側に上げる 両腕を左側へ動かす
うつむく	頭部をやや下方向へ曲げる
腕を乗せる	右腕を内側へ動かす 右手首を返す
腕をはなす	右腕を戻す(デフォルトの位置に戻す)
手を拭く	両手を体の前で合わせる 両手首を曲げ合わせる
うなづく	頭を前に2回傾ける
ギターを弾く	左肘を曲げる 左手首を内側に曲げる 右肘を曲げる 右腕を上下に動かす
起きる	腰を戻す(腰の位置をデフォルトに戻す)
置く	右腕を戻す(デフォルトの位置に戻す)
叩く	右肘を曲げる 右腕を前に振り出す

システムにおける動作スクリプトは以下の図 6 に示すような形式で定義した。Script 以下の記述が TVML の記述に該当する。下線部が身体動作の定義であり、上述の action script maker を使って記述したものである。システムは、概念表現の中に存在する行為に対応する動作スクリプト定義を検索し、TVML スクリプトを書き出す。

```
(sosogu
  (type instance)
  (is-a 注く)
  (script
    (character definepose(name= @NAME ¥, pose=注く1 ¥,
      joint=RightUpperArm ¥, rotx=-75.00 ¥, roty=0.00 ¥, rotz=0.00))
    (character definepose(name= @NAME ¥, pose=注く1 ¥,
      joint=RightHand ¥, rotx=0.00 ¥, roty=85.00 ¥, rotz=-25.00))
    (character definesequencepose( name= @NAME ¥, sequencepose=
      @SEQ ¥, pose=注く1 ¥, style=gravity ¥, time=0.5))
    (character definepose(name= @NAME ¥, pose=注く2 ¥,
      joint=RightHand ¥, rotx=0.00 ¥, roty=40.00 ¥, rotz=-25.00))
    (character definesequencepose( name= @NAME ¥, sequencepose=
      @SEQ ¥, pose=注く2 ¥, style=gravity ¥, time=0.5))
    (character sequencepose( name= @NAME ¥, sequencepose= @SEQ ))
```

図 6: 動作スクリプトの一例

5. 実行結果

総シーン数 399, 総動作数延べ 4300 個の分析票を作成した後、映画の冒頭から中盤にかけての部分(映画本編約 145 分中の 45 分)を概念表現化し、映像構成システムを用いて映像化を試みた。以上のように、行為と身体動作の定義とを対応させたデータベースを使って、システムは相対的に抽象的な記述である物語の概念表現を、より具体的な技術である TVML スクリプト

に変換する。図 5 に示すのは、生成された TVML 映像と、それに対応する実際の映画の中の場面である。

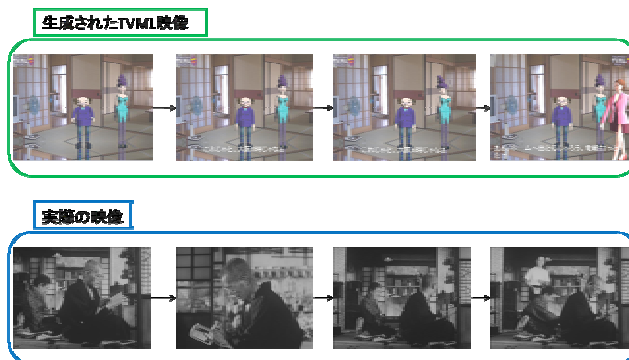


図 7: 生成された TVML 映像の例と対応する映画の場面

図 8 のリストは、このための入力概念表現である。

(読む (agent 周吉) (counter-agent nil) (instrument 時刻表) (object nil) (location 平山家) (goal nil) (floom nil) (time 朝) (narration nil) (caption nil))

図 8: 入力概念表現(一部)

さらに図 9 のリストは、生成された TVML スクリプトであり、上の映像のためのデータである。

```
reset()
skipscript(switch=on)
//キャスト定義
character: casting(name= 周吉)
character: openmodel(modelname=mattu, filename="mattu¥mattu.bm")
character: bindmodel(name= 周吉, modelname=mattu)
character: setvoice(name= 周吉, voicetype="male01")
character: casting(name= とみ)
character: openmodel(modelname=cherry, filename="cherry¥cherry.bm")
character: bindmodel(name= とみ, modelname=cherry)
character: setvoice(name= とみ, voicetype="female01")
character: casting(name= 京子)
character: openmodel(modelname=mina, filename="mina¥mina.bm")
character: bindmodel(name= 京子, modelname=mina)
character: setvoice(name= 京子, voicetype="female01")
//ローション定義
set: change(name=NULL)
prop: assign(name= hirayama)
prop: openimageplate(name= hirayama, filename="¥image¥平山家.jpg", platesize=5.5, platesizev=4.0)
prop: position(name=hirayama, x=0.1, y=1.0, z=0.3)
//タイム定義
light: assign(name= no1)
light: model(name= no1, type=flat, X=1.0, y=1.0, z=1.0, r=1.0, g=1.0, b=1.0)
light: assign(name= no2)
light: model(name= no2, type=flat, X=1.0, y=1.0, z=1.0, r=1.0, g=1.0, b=1.0)
light: assign(name= no3)
light: model(name= no3, type=flat, X=0.0, y=-1.0, z=1.0, r=1.0, g=1.0, b=1.0)
cgenv: refraction(switch=on)
cgenv: fog(switch=on, density=0.1)
//配置定義
character: position(name= 周吉, x=0, y=0.0, z=0.0)
character: position(name= とみ, x=1, y=0.0, z=0.0)
character: position(name= 京子, x=3, y=0.0, z=0.0)
skipscript(switch=off)
//アクション定義
//読む...
character: pose(name=周吉, pose=default)
character: definepose(name= 周吉, pose=yomu1, joint=Head, rotx=20.00, roty=0.00, rotz=0.00)
character: definepose(name= 周吉, pose=yomu1, joint=RightUpperArm, rotx=-45.00, roty=0.00, rotz=0.00)
character: definepose(name= 周吉, pose=yomu1, joint=RightLowerArm, rotx=0.00, roty=10.00, rotz=0.00)
character: definepose(name= 周吉, pose=yomu1, joint=RightHand, rotx=-45.00, roty=0.00, rotz=0.00)
character: definepose(name= 周吉, pose=yomu1, joint=LeftUpperArm, rotx=-45.00, roty=0.00, rotz=0.00)
character: definepose(name= 周吉, pose=yomu1, joint=LeftHand, rotx=-45.00, roty=0.00, rotz=0.00)
```

図 9: 生成された TVML スクリプト

これは概念表現からの非常に原始的なレベルでの一括映像化であり、シーンごとに映像を生成し、キャストやオブジェクトを重ねられない位置に配置、また概念表現に登場する動作を順番通りにキャストに行わせるという作業は達成できた。しかし上の例に見るように、登場人物の配置は単に重ねられないようにするだけでそれ以上の指定はできない。

実際の映画と比較してみると、現在の概念表現で表せない情報は、行為の具体的なパラメーター、すなわち歩く方向や距離、

初期ポーズなどである。また映像における多くの場面は「並列動作」から成り立っている。例えば、「AさんがBさんに「話し」ながら時刻表を「読み」、同時にBさんが後ろで旅行道具を鞆に「詰める」といった画面が普通に現れる。現在の概念表現の記述法では、並列動作を判定する方法がないため、順番にひとつづつ動作を行わせている。

しかしこれだけでも、多くのスクリプトを手作業で書く手間を省くという効果はある。今回の概念表現は手作業で作成したが、概念表現が、物語生成システムにおける物語内容及び物語言語システムにより自動生成されていることを仮定すれば、システムの有用性は増すであろう。

6. 考察

行為には、意味を持った具体的な行為(動作)と、より抽象的な行為がある。例えば、図10において、「髪をセットする」という行為は、定型的な下位行為の列をその下位に含むという意味で抽象的な行為と考えられ、その下の「手入れをする」、「ピンカールをする」といった行為によって実現される。さらに、これらは「触る」、「巻く」といったより具体的な、そして一種汎用的な行為の組み合わせによって実現される。しかしこれらが(写実主義的な意味での)映像によって表現されるためには、より具体的な身体運動の系列によって表現されなければならない。

多くの場合、映像は「身体運動」で表現される。しかし、概念表現レベルの記述には、抽象的な行為が含まれる。概念表現と映像表現を繋ぐには、身体運動を最下層に置く「行為の階層的体系」が必要である。つまり、現在は、すべての行為を同じ階層でデータベースに登録し、行為ひとつひとつのすべてに身体運動を定義しており、そのため、抽象的な行為を定義しようとした際、行為ごとに多くの身体運動を定義しなければいけなくなるが、行為をもっと効率よく表現するためには、行為の階層的な体系を構築することが必要となる。図11に示すように、物語生成システムが要求する行為の階層体系において、最下位の階層は「身体運動」であり、その上位が行為の階層に当るが、行為の中でも意味を持った最小単位を特に「動作」と呼べる可能性がある。動作は抽象的な行為と具体的な身体運動を架橋する位置にあり、非常に汎用的に意味を持っている。すなわち、様々な上位レベル行為をその組み合わせによって表現する能力を持つ。また、それが身体運動に直接つながる行為のレベルであると考えられる。

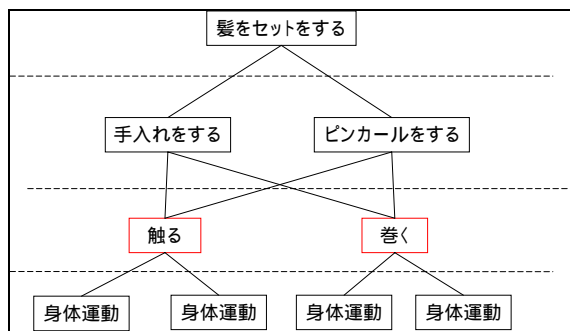


図10: 行為の抽象性(具体性)の違い

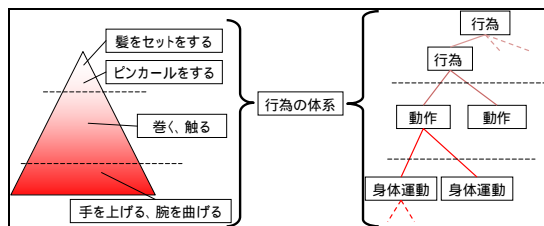


図11: 行為の階層体系のイメージ

さらに、図12に示すように、同じ動作によって表される行為であっても、対象となる要素(人や物など)の違いにより身体運動の組み合わせが異なる。行為の階層体系の中にはこのような性質も含まれている必要がある。

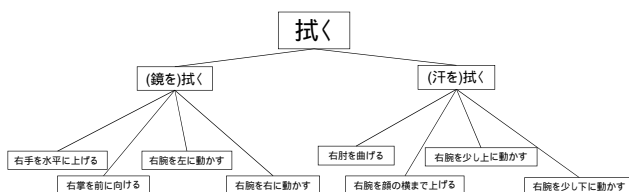


図12: 対象による行為の定義の違い

7. むすび 今後の展望

本稿では、物語概念表現からの自動映像構成の実現に向けたシステムを紹介し、『東京物語』を素材とした行為に基づき『東京物語』を概念表現化し、自動映像化の実験を試みた。非常に粗野なレベルではあるが、概念表現からの自動映像化に成功した。課題は多数であるが、特に行為と身体運動を架橋する階層的体系の構築が必須の要求である。

参考文献

[真部 2007a] 真部雄介, 小方孝: 物語生成における映像からの動作概念記述のボトムアップアプローチ 動作から行為への階層性に基づいて, 人工知能学会全国大会(第21回)論文集, 1F1-7, 2007.

[真部 2007b] 真部雄介, 小方孝: 物語生成システムにおける映像の自動生成のための行為概念記述に関する基礎的考察, 日本認知科学会第24回大会発表論文集, 420-423, 2007.

[小方 1996] 小方孝, 堀浩一, 大須賀節雄: 物語のための技法と戦略に基づく物語の概念構造生成の基本的フレームワーク, 人工知能学会誌, 11(1), 148-159, 1996.

[小方 2003a] 小方孝: 物語の多重性と拡張文学理論の概念システムナラトロジーに向けてⅠ, 吉田雅明編: 複雑系社会理論の新地平, 専修大学出版局, 127-181, 2003.

[小方 2003b] 小方孝: 拡張文学理論の試み システムナラトロジーに向けてⅡ, 吉田雅明編: 複雑系社会理論の新地平, 専修大学出版局, 309-356, 2003.

[小方 2007] 小方孝, 松田亜矢子, 内藤祐介, 真部雄介, 高橋昇, 中嶋美由紀, 吉尾貴史, 沼田真克: 物語生成システムにおける映像表現, 人工知能学会第2種研究会ことば工学会研究会(第25回)資料, 19-59, 2007.

[小方 2008] 小方孝: 物語生成システムにおける映像構成へ向けて, 金井明人, 丹羽美之編著, 映像編集の理論と実践, 法政大学出版会, 165-235, 2008.