

背景と物体動作の連携方式を用いた動画生成システムの設計と実現

佐々木 敏晃[†] 片山 喜章[†] 高橋 直久[†]

[†] 名古屋工業大学大学院 工学研究科 情報工学専攻 〒466-8555 愛知県名古屋市昭和区御器所町

E-mail: †sasaki@moss.elcom.nitech.ac.jp, ††{katayama,naohisa}@nitech.ac.jp

あらまし 本稿では背景と物体動作を連携させた動画生成システムを提案する。提案システムは次の機能を有する。(1) 物体動作空間と背景空間の自由な部分変形機能。(2) 物体動作時間の自由な部分変形機能。(3) 物体動作空間（または物体動作時間）の変更による物体動作時間（または物体動作空間）の再設定機能。上記の機能により、物体動作空間または物体動作時間のいずれか一方を変更するだけで、目的に応じた動画が直感的かつ少ない手数で生成可能である。また、変更により作成される動画では、時間と空間の連続性が保たれる。本稿ではまた、プロトタイプを用いて提案システムの有用性を示す。

キーワード データの可視化, ユーザインタフェース, 時空間 DB

Design and realization of a movie-making system with an interaction between context and object motion

Toshiaki SASAKI[†], Yoshiaki KATAYAMA[†], and Naohisa TAKAHASHI[†]

[†] Department of Computer Science and Engineering Graduate School of Engineering,

Nagoya Institute of Technology

Gokiso, Showa, Nagoya, 466-855 Japan

E-mail: †sasaki@moss.elcom.nitech.ac.jp, ††{katayama,naohisa}@nitech.ac.jp

Abstract This paper describes a movie-making system creating a movie in which context and object motion interact with each other. The system has the following functions: (1) Open-ended transformation of any part of the object motion space and context space. (2) Open-ended transformation of any part of the object motion time. (3) Automatic reconfiguration of the object motion time or space, subjected to changes of the object motion space or time. These functions enables a user to create movies with intuitive operations according to user's purpose; simply by changing either space or time of the object motion. Continuity in the object motion space and time is retained as well. Moreover, this paper clarifies the usability of this movie-making system by several experiments using a prototype.

Key words Visualization of Data, User Interface, Spatial and Temporal Database

1. はじめに

ある物の位置や様子が連続して変化するような事象を表現するための手法として、言葉による表現や、変化の様子を断片的に画像として表現する方法、あるいは変化の様子を連続的に動画で表現する方法などが考えられる。これらの内、直感的に事象を理解するための手法としては、動画がもっとも優れていると考えられる [1]。

本稿では、動画で表現できる様々な事象のうち、ある決まった移動経路上を物体が移動するような事象を対象として考える。例えば、地図を用いたナビゲーションでのナビゲート対象の動きや分散アルゴリズム [2] におけるネットワークでやり取りさ

れるメッセージが本稿で考える事象に相当する。このような事象は、点と辺で構成された図形 (グラフ) 上を物体が移動する事象として抽象化できる。このような抽象化を出来る事象は、他にも様々なものが考えられる。つまり、グラフ上を物体が移動する動画により多くの事象が表現できると考える。

一方、動画により事象の理解を効率的に行う方法として、聞き手 (動画を見る者) に合わせた動画を用意することがある。これは、聞き手により説明する事象に対する理解度や興味を持っている点に相違があるためである。例えば、画面のある部分の移動が見つからない場合には、見つからない部分を拡大して再生したい、わずかな時間でいろいろな動作をしている場合には、その部分をゆっくり再生したいことがある。そのため、説明側 (動画を

作成する者) は効率的に理解させるために聞き手ごとに異なる動画を用意する必要がある。しかし、説明側が聞き手の要求を推測して予めいくつかの動画を用意していたとしても、不意の要求や推測が外れていた場合には、その場で新たに動画を用意する必要がある。この様に、ある事象を説明する動画を複数用意することは非常に多くの手間がかかり、またその場で新たに作成する必要があるなどの問題点がある。

そこで本稿では、一つの動画から聞き手の要求に応じてその場で新たに動画を生成するシステムを提案する。本提案システムは、説明側は予め複数の動画を用意する必要が無く、聞き手に説明する際に、動画の再生時間や表示空間を部分的に変更し、聞き手の要求に応じた動画をその場で生成することが可能になる。本提案システムは次の機能を有する。

機能 1 物体動作空間と背景空間の自由な部分変形機能。

機能 2 物体動作時間の自由な部分変形機能。

機能 3 物体動作空間（または物体動作時間）の変更による物体動作時間（または物体動作空間）の再設定機能。

上記機能により提案システムでは、利用者は物体動作空間または物体動作時間のいずれか一方を変更するだけで、目的に応じた動画を直感的かつ少ない手数で得ることができる。得られた動画では、変更によっても時間あるいは空間の連続性が保たれる。

本稿の以降の構成は次の通りである。2 章で関連研究について記し、続く 3 章で提案システムの概要を述べ、そして 4 章で提案システムの実現方式を記す。さらに 5 章で提案システムの適用実験について述べ、最後に 6 章でまとめを行ない今後の課題を明らかにする。

2. 関連研究

説明用の動画を作成するツールとして PowerPoint [3] が広く用いられている。PowerPoint は一般的なプレゼンテーションシステムであり、説明を行う際によく使用される。しかし、PowerPoint では、空間や時間の一部を変更した動画を作成するには、新たに作り直さなければならない。さらに背景画像の変形も他のシステムで行う必要がある。また、時間と空間の連携は無く、新たに動画を作成するときに時間と空間の関係を把握しながら作成しなければならない。

また、空間の一部を変形するシステムとしては Fisheys [4] [5] やゴム伸縮メタファによる認知地図再生システム [6] がある。これらシステムでは、空間の一覧性を損なうことなく空間の視認性を上げた変換が可能である。しかし、これらシステムでは、画像を対象としたシステムであるため、動画の時間との連携を行うことは考慮されていない。

更に、時間の一部を変更するものとして、動画の全体の再生時間を自由に変更可能な動画生成エンジンに関する研究 [7] や時間的な概観と詳細の表現およびインタラクションに関する研究 [8] がある。これらの研究では、動画の時間に対して変更が可能である。しかし、時間と空間の変更は独立しているため、利用者は時間と空間の関連を把握しながら変更を行わなくてはならない。

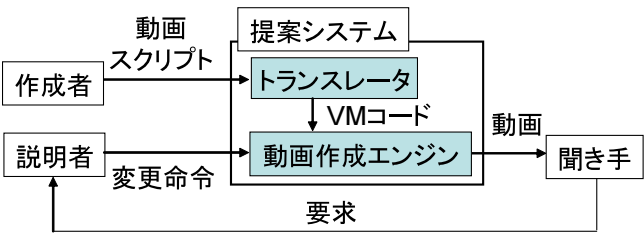


図 1 動画作成の流れ

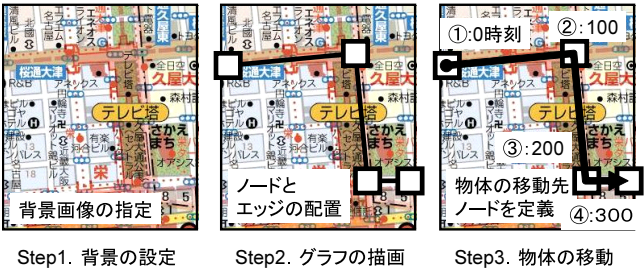


図 2 動画の作成手順

表 1 記述に用いる命令とその引数

命令	引数
create	ID, オブジェクトの種類
set	設定先 ID, 実行時刻, 設定属性名, 設定値
createNode	ID, 座標 (x, y)
createMsg	ID, 初期配置ノードの ID
moveMsg	ID, 実行時刻, 経過時間, ノードの ID
connection	ノードの ID, ノードの ID

3. 提案システムの概要

本章では、提案システムの概要と実現へのアプローチについて述べる。提案システムを利用した動画作成の大まかな手順は、以下の通りである (図 1 参照)。

手順 1 動画作成者が基本となる動画を記述した動画スクリプトをシステムに与える。

手順 2 動画説明者 (作成者と同じの場合もある) は動画を聞き手に提示する。

手順 3 聞き手から要求がある場合には、動画説明者は動画に対して必要に応じて変更を加える。

手順 4 以下、2, 3 を繰り返す。

つまり、動画を作成する際にはその基本となる動画をひとつ作成しておけば良く、聞き手の要求によってその場で動画を変更して提示することが出来るシステムである。上記の 2, 3 を繰り返す過程で、多様な要求に対応でき、またある要求に対してもきめ細かく対応することが可能となる。

以下では提案システムの動作の流れに沿って、動画スクリプトの作成、動画スクリプトからの動画の生成、動画の変更 (時空間の変換) について述べる。

3.1 動画スクリプトの作成

本システムを用いて動画を作成する手順を例を用いて説明する。本システムは利用者の作成した動画スクリプトと呼ぶ一種のプログラムを解釈実行することにより動画を生成する。このため利用者はまず、表 1 の命令を用いて次の Step で動画スクリプトを記述する。

```

create(背景, Plane)
0:set(背景, ..., image="栄.JPG", visible=true)
createNode(node1, 10, 200)
createNode(node2, 400, 190)
createNode(node3, 425, 420)
createNode(node4, 550, 420)
connecte(node1, node2)
.....
0:moveMsg(msg1, node2, 100)
100:moveMsg(msg1, node3, 100)
200:moveMsg(msg1, node4, 100)

```

図 3 動画スクリプトの記述例

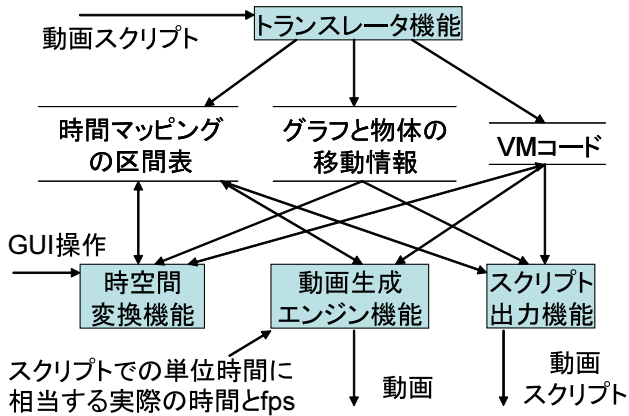


図 4 提案システムの全体構成図

Step1 背景の設定

- (1) 動画で用いる背景として画像を決める。
- (2) create 命令を用いて背景とするオブジェクトを作成する。
- (3) set 命令を用いて上記の画像と背景オブジェクトを結びつける。例えば、図 2 では地図画像と背景オブジェクトを結びつける。

Step2 背景にグラフを描画

- (1) createNode 命令を用いて背景画像の上にノードを配置する。例えば、図 2 の様に地図を背景としたときには、交差点や建物などにノードを配置する。
- (2) connecte 命令を用いてノード間を接続する。
- (3) set 命令を用いてノードの形状や大きさや色、エッジの太さや色など、グラフの表示形体を指定する。

Step3 グラフ上のノード間を動く物体の記述

- (1) createMsg 命令を用いて物体を生成し、ノード上に配置する。
 - (2) 生成した物体に対して set 命令を用いて表示形体やアイコン等を指定する。
 - (3) sendMsg 命令を用いて物体の移動先ノード名、移動開始時刻、移動に掛かる時間を指定する。例えば、図 2 のように物体の移動先ノードとその通過時刻を指定する。
- 上記の例では、Step1 から Step3 により図 3 に示す動画スクリプトを作成する。

3.2 動画の生成

システム利用者は次の項目を本システムに入力する。

入力 1 動画スクリプト S

入力 2 一秒あたりに表示するフレーム枚数 f

表 2 ノード情報

ID	x 座標	y 座標
node1	10	200
node2	400	190
node3	425	420
node4	550	420

表 3 物体の移動情報

ID	開始時刻	継続時間	移動先ノード
msg1	0	100	node2
msg1	100	200	node3
msg1	200	300	node4

表 4 動画を構成する VM コード

ID	種類	実行時刻	属性	設定値 (順再生)	設定値 (逆再生)
node1	plane	0	x	10	0
node1	plane	0	shape	rect	rect
node1	plane	0	thickness	4	1
msg1	plane	0	x	10	0
msg1	plane	0	incX	3.9	0
msg1	plane	100	incX	0.25	3.9

表 5 各区間での変換率

開始時刻	終了時刻	変換率 (ms)
0	100	10
100	200	10
200	300	10

入力 3 時間軸の変換率 α

入力 4 時空間を変更する操作 o

ここで時間軸の変換率とは後述のように、動画スクリプトに記述する時間軸と実際の時間軸との対応付けを与えるパラメータである。本システムは図 4 に示すように、トランスレータ機能と動画生成エンジン機能、時空間変換機能から成る。本システムは次の処理により動画を生成する。

生成処理 1 トランスレータ機能は動画スクリプト S から次の表を作成する。

- **ノード情報表** createNode 命令から、背景画像上に配置されたノードの ID と位置座標を保持する表を作成する。図 3 の動画スクリプトの場合には表 2 が作成される。

- **物体の移動情報表** moveNode 命令から、物体の移動先、物体が移動する開始時刻、物体の移動に掛かる時間を記録した表を作成する。図 3 の動画スクリプトの場合には表 3 が作成される。

- **VM コード表** 動画再生エンジンが実行可能なコード。動画スクリプトのすべての命令からオブジェクトの ID と種類、オブジェクトの持つ属性の変更値と変更時刻で構成される表を作成する。図 3 の動画スクリプトの場合に作成される VM コードの一部を表 4 に示す。

- **各区間での変換率** 物体がノードを通過する時刻を、物体の移動情報テーブルから取り出す。その時刻を境界とした区間に時間を分解する。さらに、利用者が指定した時間軸の変換率 α を各区間ごとに記録するテーブルを作成する。上記例では表 5 が作られる。動画生成エンジン機能では変換率を使用することで、動画スクリプトの時間軸と実際の時間軸の変換を行う。

生成処理 2 動画生成エンジン機能は 1/f 秒毎に生成処理 2-1 から生成処理 2-3 までを繰り返す。

生成処理 2-1 動画生成エンジン機能は、次のフレーム時刻 t と表 5 を用いて、動画スクリプトの時間軸での時刻 t' を求める。例の場合では、表 5 に示すように各区間の変換率は $\alpha = 10(\text{ms})$ である。このとき実際の時刻 $t=1500(\text{ms})$ はスクリプ

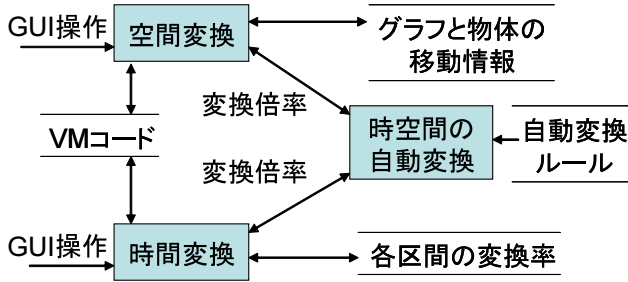


図5 時空間変換機能構成図

ト時間軸上の時刻 $t'=150$ に変換される。

生成処理 2-2 動画生成エンジン機能は、時刻 t' と VM コードから時刻 t' におけるフレームを作成する。例の場合では、 $t'=150$ とすると表4に示す VM コードから時刻 150 におけるオブジェクトの属性値を求める。msg1 の位置を示す属性 x は $t'=150$ のとき $x=412.5$ となる。この様に $t'=150$ における各オブジェクトの属性値を求め、属性に従い描画することでフレームを作成する。

生成処理 2-3 動画生成エンジン機能は、動画生成生成処理 2-2 で作成したフレームの表示を行う。

本システムでは上記の生成処理により、動画スクリプトを動画として表現する。

3.3 動画の時空間変換

利用者が本システムで動画の一部を変更する方法を述べる。利用者は次のような変更を行う。

変更方法 1 背景画像の差し替え

変更方法 2 空間の変換

変更方法 3 時間の変換

上記方法は、スクリプトを直接書き直す、あるいは時空間変換機能により空間と時間を直接変更することにより行われる。時空間変更機能は変更内容を VM コードや各区間の変換率に反映する。そして、スクリプト出力機能は変更した VM コード、各区間の変換率、グラフと物体の移動情報から変更内容を含んだ動画スクリプトを出力する。これにより変更を施した動画スクリプトを生成する。

時空間変換機能は図5に示すように空間変換機能と時間変換機能、時空間の自動変換機能から成る。利用者は次の操作手順により、空間変換機能に変更量を入力する。

空間変換操作 1 物体を一個選択する。

空間変換操作 2 選択した物体が移動する軌跡上のノードを2個選択する。

空間変換操作 3 選択した2個のノードの移動量を指定する。利用者により決定された2個のノードの移動量から、空間変換機能では次の処理により空間の一部分を変換する。

空間変換処理 1 空間変換機能は利用者が選んだ2個のノードとグラフと物体の移動情報表から、線形性を保つ空間を求める。

空間変換処理 2 時間変換処理1で求めた空間の線形性を保ち、その周りにしわ寄せが来る空間フィルタを、2個のノードの移動量から作成する。

空間変換処理 3 空間フィルタを使用して、背景画像の変形、

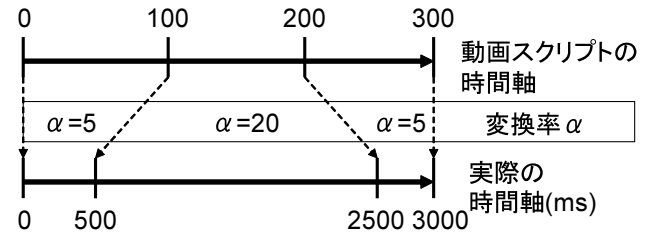


図6 時間変換例

ノードの位置、物体の移動速度といった VM コード中の項目を変更する。

また、利用者は次の操作手順により、時間変換機能に変更量を入力する。

時間変換操作 1 物体を一個選択する。

時間変換操作 2 選択した物体が移動する軌跡上のノードを2個選択する。

時間変換操作 3 選択したノード間を物体が通過する時間に対して、時間軸の変換率を設定する。

利用者により設定された一部区間に対する時間軸の変換率から、時間変換機能は次の処理を行ない、動画全体の再生時間を調節する。

時間変換処理 1 全体の再生時間が変化しないように、利用者の操作により選択されなかった時間の区分の変換率を調整する。

時間変換処理 2 調整された各区間の変換率を書き換えて表5を更新する。

例えば、図3に対して、node2 から node3 への区間を選択し、その区間の変換率を10から20に変更すると、図6の示す時間軸の変換率が求まる。例では、物体が node2 から node3 までの移動がゆっくりと成った再生となる。

さらに、時空間の自動変換機能は空間と時間を連携させて自動的に変換を行う。利用者は自動変換ルールを設定し、上記の空間変換操作または時間変換操作を行う。時空間の自動変換機能は次の処理により自動的に変換を行う。

自動変換処理 1 選択された物体に対して、時間変換機能、空間変換機能で変換された、移動軌跡上の距離と時間の変換倍率を求める。

自動変換処理 2 自動変換ルールに従って、物体の移動時間や移動速度、移動時間の内、追加して変更する項目を時間変換機能や空間変換機能で変換する。

4. 実現方式

本章では、本システムの特徴的な機能である時空間変換機能の実現方式について述べる。時空間変換機能は、空間変換機能、時間変換機能、時空間の自動変換機能から成る。以下、各節で各機能の実現方式を説明する。

4.1 空間変換

空間変換機能は入力パラメータとして動画上を移動する物体 A の ID id_A から以下を得る。

入力 1 物体 A の移動軌跡上の点 P_B 、および P_B を通過する時刻 t_B

入力 2 点 P_B とは異なる, 物体 A の移動軌跡上の点 P_C , および P_C を通過する時刻 t_C

入力 3 点 P_B , P_C の変換先の位置 P_D , P_E (直線 P_BP_C 上の点とする)

空間変換機能では, id_A , t_B , t_C , P_B , P_C により, 物体が移動する領域空間を求め, P_D , P_E により, ノードの移動先の位置座標に変換される空間の拡大縮小を行うフィルタを生成する. 最後に, 作成したフィルタを使用して, 物体の位置や移動量の変更と背景画像の変形を行うことで, 物体の移動空間と背景空間を変換する.

実行手順を次に示す. まず, 次の手順で物体が移動する空間領域を求める. 求めた領域空間を図 7 に示す.

手順 1-1 物体移動情報テーブルから, 物体 A が時刻 t_B から時刻 t_C の間に通過するノードの ID id_X を取得.

手順 1-2 ノードの ID id_X とノード情報テーブルから, 各ノードの位置座標を取得.

手順 1-3 点 P_B と点 P_C との線分を水平軸として, 手順 1-2 から求めた各ノードの位置座標を内包する最小の矩形を算出.

次に, 手順 1-1 から 1-3 で求めた領域の拡大縮小を行うフィルタ関数を求める.

手順 2-1 入力 4 より, 以下の式 1 を満たす変換の中心点 P_M を求める.

$$|\overrightarrow{P_MP_B}| \cdot |\overrightarrow{P_MP_D}| = |\overrightarrow{P_MP_C}| \cdot |\overrightarrow{P_MP_E}| \quad (1)$$

手順 2-2 中心点 P_M から, 拡大縮小した後の領域を求める.

手順 2-3 手順 2-2 で求めた領域に変換することで不連続となる部分を補完する領域を設定する. (図 8)

手順 2-4 領域の変換する関数と, その逆変換となる関数を算出する.

上記の手順により任意の点 P_X に対して変換後の点 P_Y との関係式が得られる. 点 P_R , P_S , P_T は点 P_M , P_X を含む直線の P_X 側の線分と変換前領域, 変換後領域, 連続性を確保する領域のそれぞれの境界との交点である.

- 物体が移動する空間領域

$$\overrightarrow{P_MP_Y} = \overrightarrow{P_MP_X} \cdot \frac{|\overrightarrow{P_BP_C}|}{|\overrightarrow{P_DP_E}|} \quad (2)$$

- 連続性を補完する領域

$$\overrightarrow{P_MP_Y} = \overrightarrow{P_MP_S} + \overrightarrow{P_RP_X} \cdot \frac{|\overrightarrow{P_RP_X}| |\overrightarrow{P_SP_T}|}{|\overrightarrow{P_RP_T}|} \quad (3)$$

最後に, 空間変換を行う関数を用いて, 背景空間を示す画像や, 物体の位置座標, 移動量, ノードの位置座標の変換を行ない, それらに関する VM コードを再生成し, もとの VM コードと置き換える.

4.2 時間変換

時間変換機能では, 次の項目を入力パラメータとして, 物体 A の ID id_A から以下を得る.

入力 1 物体 A の移動軌跡上のノードを通過する時刻 t_B

入力 2 時刻 t_B とは異なる, 物体 A の移動軌跡上のノードを通過する時刻 t_C

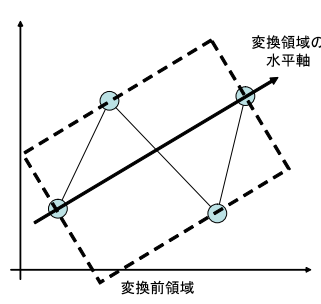


図 7 物体が移動する空間領域

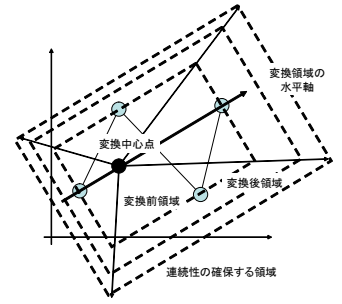


図 8 変形後の領域と補完領域

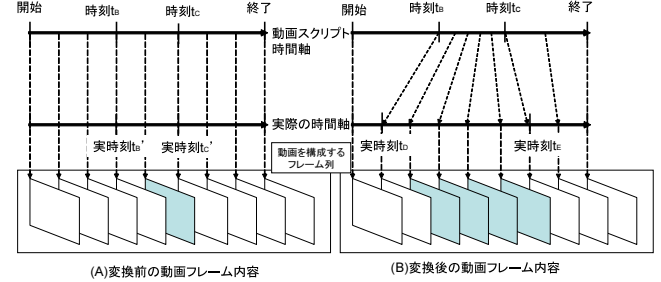


図 9 時間変換によるフレームの内容変更

入力 3 時刻 t_B , t_C の変更先の実際の時刻 t_D , t_E

入力 4 変換率 α

時間変換機能では, 動画スクリプトの時間軸と実際の時間軸との変換率を変更する. 変更された各区間での変換率から動画生成エンジン機能から, 部分で再生速度が異なる動画を作成する.

実行手順を次に示す. まず, 動画の時間を指定した物体の移動情報から分割する.

手順 1-1 物体移動情報テーブルから, 物体 A がノードを通過する時刻を求める.

手順 1-2 手順 1 で求めた時刻を境界とした区間に時間を分解する.

手順 1-3 各区間に初期値として変換率 α を設定する.

次に, 各区間の変換率を調整する.

手順 2-1 各区間の変換率から, 入力 2, 3 の時刻に対応する実際の時刻 t'_B , t'_C を求める.

手順 2-2 時刻 t'_B , t'_C 間の時間に対する時刻 t'_D , t'_E 間の時間の倍率 x を求める.

手順 2-3 時刻 t_B , t_C 間にあるすべての区間の変換率を x 倍する.

手順 2-4 全体の時間が一定になるように, 残りの区間を等倍率で変更する.

4.3 時空間の自動変換

時空間操作機能では, 時間変換機能と空間変換機能を利用して利用者の変換に合わせた動画の生成を行う. この機能により本システムの特徴を実現する. 時空間の自動変換機能では, 次の項目を入力とする.

入力 1 動画上を移動する物体 A

入力 2 物体 A の移動軌跡上の点 P_B , および P_B を通過する時刻 t_B

入力 3 点 P_B とは異なる, 物体 A の移動軌跡上の点 P_C , お

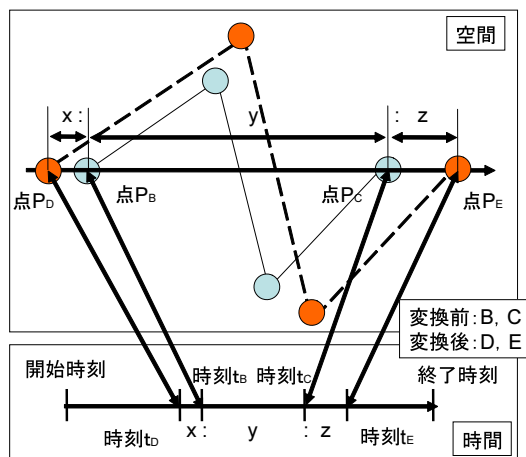


図 10 時間と空間の変換前後の位置関係

表 6 自動変換ルール			
移動項目	距離	時間	速さ
状態	可変	追従	固定

および P_C を通過する時刻 t_C

入力 4 点 P_B, P_C , およびそれらの点を通過する実際の時刻 t_B, t_C の変換先の点 P_D, P_E , 実際の時刻 t_D, t_E

入力 5 変換率 α

時空間の自動変換機能では、上記の入力に対して、空間か時間の何れか一方のみの変換を行ったものを入力とする。時空間の自動変換機能では、表 6 に示す自動変換ルールに基づいて次の変換を行う。表 6 の項目には以下の意味がある。

- **可変** 利用者により変更される項目
- **追従** 可変項目の変動に連動して変更される項目
- **固定** 変更されない項目

まず、自動変換ルールの内、空間が可変で時間が追従であった場合の手順を示す。

手順 A-1 点 P_B, P_C, P_D, P_E の位置関係と比例するように、実際の時刻 t_B, t_C から実際の時刻 t_D, t_E を求める。

手順 A-2 時間変換機能に時刻 t_B, t_C と実際の時刻 t_D, t_E と変換率 α を入力して時間を変換する。

次に、自動変換ルールの内、時間が可変で空間が追従であった場合の手順を示す。

手順 B-1 実際の時刻 t_B, t_C, t_D, t_E の時間間隔と比例するように、点 P_B, P_C を含む直線上の点 P_D, P_E を求める。

手順 B-2 空間変換機能に点 P_B, P_C, P_D, P_E を入力して空間を変換する。

自動変換ルールに従い、図 10 に示す時間と空間の関係を保持するために上記の手順を実行することで、時間と空間を自動的に変換する。この機能により物体動作空間または物体動作時間のいずれか一方を変更するだけで、目的に応じた動画が直感的かつ少ない手順で生成可能にする。

5. 評価実験

提案システムの評価のためにプロトタイプシステムを構築した。このプロトタイプシステムを用いて、設定した状況におい

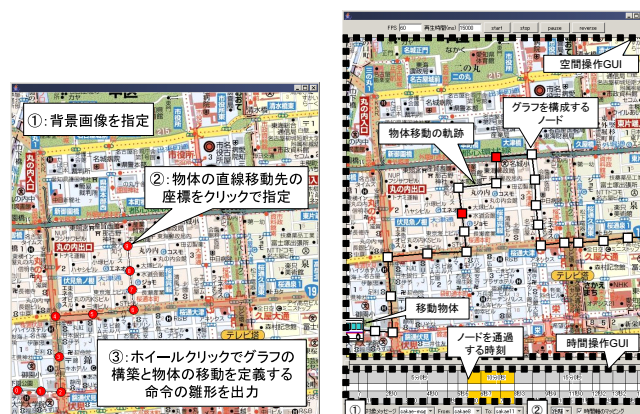


図 11 動画スクリプト作成 GUI

図 12 時空間変換 GUI

て提案システムの機能の妥当性を確認する。

5.1 プロトタイプシステム

プロトタイプシステムでは、動画記述命令として、表 1 に示す命令を実装し、動画スクリプトを作成するために図 11 に示す GUI を実装した。また、動画の時間や空間を変更を行うために図 12 に示す GUI を実装した。

物体動作の定義は GUI 上に表示した背景画像上で物体が移動する経路を直接指定する。物体動作の指定を終えると標準出力から動画スクリプトの雛形が出力される。この出力された雛形を基に物体の移動開始時刻や移動時間等の設定を書き換えることで、物体動作の詳細な定義を行う。

動画の時間や空間を変換を行う GUI では、先の動画スクリプトを入力とし、次の操作を行うことで変形を行う対象を選択する。

操作 1-1 GUI 上の 1 の ComboBox から対象とする移動物体を選択する。

操作 1-2 GUI 上から変換対象とする物体の移動軌跡を移動系軌跡上の始点と終点となるノードをダブルクリック (DClick) することで選択する。

操作 1-3 GUI 上の 2 の ComboBox から固定にする項目 (距離, 時間, 速度) を選択する。

次に、空間の部分変更を行う操作を示す。

操作 2-a 操作 1 で選んだ始点と終点となるノードをドラッグする。ドラッグした座標が変換先の座標となる。

最後に、時間の部分変更を行う操作を示す。

操作 2-b 操作 1 で選んだ始点と終点となるノード上を移動する時刻が GUI 上の時間を操作する空間に縦棒として表示されているので、その棒をドラッグする。ドラッグした座標が変換先の時刻となる。

上記の操作により、物体の移動軌跡上の一部の空間及び時間の変更を行う。

5.2 適用実験

事象の説明の一つである、地図を用いたナビゲーションをプロトタイプシステムに適用する。システム利用対象として、名古屋城から南方の地域のバスのナビゲーションを考える。このナビゲーションでは以下の 3 つの状況での変化を想定する。

状況 1 空間の部分的拡大として、曲がるべき交差点の周りだけ地図を拡大することで詳しく説明したい。

状況 2 曲がるべき交差点の周りを通過するときだけゆっくり再生することで曲がる順番を詳しく説明したい。

状況 3 空間の部分的拡大として、曲がるべき交差点の周りだけ地図を拡大して、さらに物体の移動速度を変えずに再生することで詳しく説明したい。

上記の状況に対して利用者が次の操作を行うと、各状況に応じた動画スクリプトを生成する。まず、バスのナビゲーションを行う基本的な動画を図 13 の様に作成した後、記述した動画スクリプトを入力することで、プロトタイプシステムの初期画面は図 14 に示す内容と成る。各状況の共通の操作として、プロトタイプシステムの操作 1-1 から 1-2 の操作を行うことで、バスの移動軌跡上で詳しく説明したい区間を選択する。そして、各状況により次の操作を行う。

- 状況 1-操作 操作 1-3 で、固定項目を時間に指定。操作 2-a により、軌跡の始点終点を拡大したい領域までドラッグ。
- 状況 2-操作 操作 1-3 で、固定項目を距離に指定。操作 2-b により、軌跡の始点終点時刻を拡大したい時刻までドラッグ。
- 状況 3-操作 操作 1-3 で、固定項目を速度に指定。操作 2-a により、軌跡の始点終点を拡大したい領域までドラッグ。

上記の各状況に対する操作により生成された動画の画面は図 15～17 に示す結果となった。また、各状況で生成された物体の移動速度及び画面の状態は表 7 に示す結果となった。さらに、各状況下での動画の部分的な変更にかかった手数と VM コードの変更量は表 8 に示す結果となった。

プロトタイプシステムにより生成された動画について考察する。図 15 では、指定したバスの移動区間に対して、地図の拡大が行われている。また、バスの画面上の移動も地図の拡大に合わせて変更されている。この動画を再生すると、バスが一部拡大した地図上を動くので、バスが曲がるべき交差点の情報を詳しく説明することが可能である。さらに、地図の連続性を保ちながら拡大しているので、他の区間とのつながりも示すことが可能である。図 16 では、指定した移動区間をバスが移動する時の移動速度が遅くなっている。また、全体の再生に掛かる時間は一定になるように各区間で再生速度が調整されている。この動画を再生すると、バスが指定区間をゆっくり移動するので、利用者はバスの移動に合わせて交差点に関する情報をより多く説明することが可能である。また、動画を一定時間で終わらせることが可能である。図 17 では、指定したバスの移動区間に対して、地図の拡大が行われている。また、バスの画面上の移動も地図の拡大に合わせて変更されており、バスの移動速度を一定にするように再生速度も変更されている。この動画を再生すると、バスが一部拡大した地図を拡大前の速度のまま移動する。これにより、利用者はバスが曲がるべき交差点の情報を拡大した地図上で示し、遅くなったバスの移動に合わせてより多くの情報を伝えることが可能である。

実験結果より次の結論が得られる。表 7 の内容より、各状況において目的とする動画を生成できたことを確認できた。また、生成に掛かる手数も表 8 より、GUI 上の物体の選択によるク

0:create(背景, Plane)

0:set(背景, ..., image="script/sakae.JPG", visible=true)

: 背景の配置

0:createNode(sakae0, 13, 648)

...

0:createNode(sakae21, 663, 443)

: ノードの配置

0:connect(sakae0, sakae1)

...

0:connect(sakae20, sakae21)

: ノード間にリンクを生成

0:set(sakae0_sakae1, thickness=3, visible=true)

...

0:set(sakae20_sakae21, thickness=3, visible=true)

: リンクの表示設定

0:set(sakae0, visible=true)

...

0:set(sakae21, visible=true)

: ノードの表示設定

グラフと背景の構築

物体の生成とノード上の移動

0:createMsg(sakae-msg, sakae0, 50, 50, RECT, 白, "")

0:set(sakae-msg_rect, image="script/bus_w.gif")

10:moveMsg(sakae-msg, sakae1, ..., sakae21, 1000)

図 13 名古屋城南方の地域をバスが移動する動画スクリプト

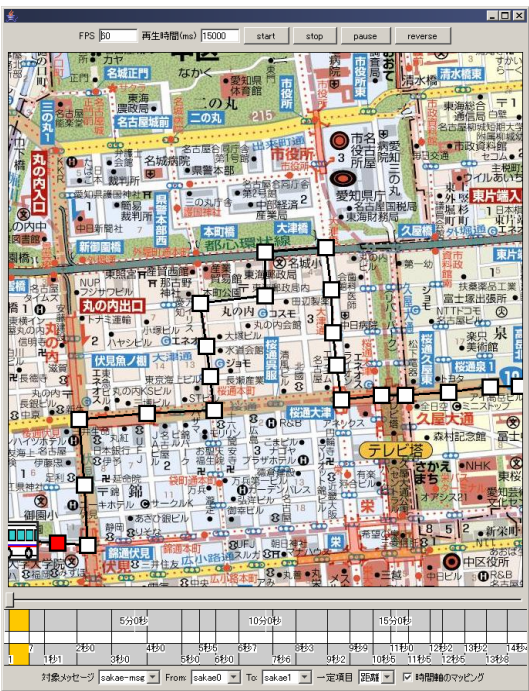


図 14 初期画面

表 7 各状況下での空間と時間の変更結果

状況	空間変形	移動速度
状況 1	指定領域での空間の拡大を確認	指定領域で速度が速くなるのを確認
状況 2	変化なし	指定領域で速度が遅くなるのを確認
状況 3	指定領域での空間の拡大を確認	変化なし

リック、注目経路の選択によるダブルクリック、画面所の 2 点または 1 点の地点に対するドラッグのみであることがわかる。また、動画の部分的な変更による VM コードの変換量は表 8 により、VM コードを直接書き換えるより簡単な少ない手数で部分的な変更が出来ることを確認した。さらに、固定項目を変更することで、ほぼ同じ操作手順により異なる動画を生成することが可能であることを確認できた。

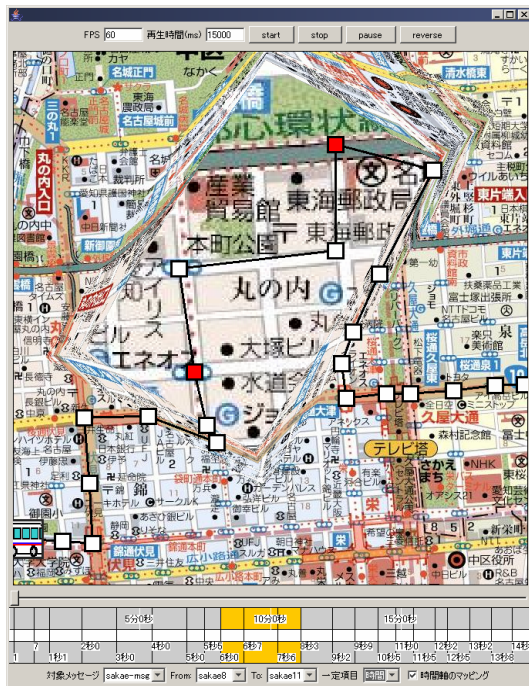


図 15 時間を固定した状況 1 の実行結果画面



図 17 速度を固定した状況 3 の実行結果画面

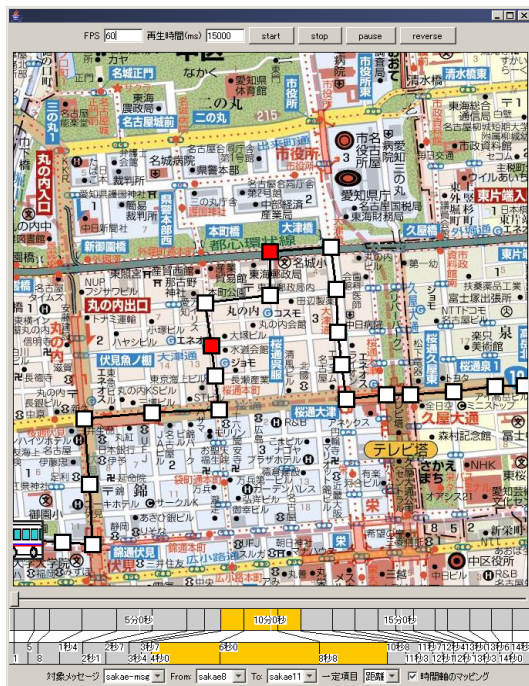


図 16 距離を固定した状況 2 の実行結果画面

表 8 各状況下での操作回数と VM コード変更量

状況	操作回数	VM コード変更量 (命令)
状況 1	Click2 回, DClick2 回, Drag2 回	35
状況 2	Click2 回, DClick2 回, Drag2 回	44
状況 3	Click2 回, DClick2 回, Drag2 回	82

入力記述の VM コード量: グラフの構築-174 命令, 物体の移動-49 命令

6. おわりに

本稿では、空間や時間の変更を容易に行えるシステムとして、背景と物体の移動を連携した動画生成システムの提案と実現を

行った。そして、プロトタイプシステムを作成し、次の 3 状況において提案システムを適用した。(1) 空間の部分的な拡大。(2) 再生時間の部分的な変更。(3) 物体の移動速度を一定に保ったままの状態での空間の部分的な拡大。その結果、各状況において要求を満たす動画の作成を確認し、生成に掛かる手数も少なく、共通の操作で出来る事を確認した。

今後の課題としては、空間の変形により一部の線形関係が崩れており、その部分に対して物体の移動と背景空間の対応が変動してしまっている。その部分を考慮した物体の移動内容を変更する機能が必要であると考えられる。また、提案システムの適応範囲の確認も課題の一つである。

謝辞 本研究の一部は人工知能研究振興財団の研究助成によるものである。

文 献

- [1] 小池 英樹: bit 別冊 ビジュアルインタフェース -ポスト GUI を目指して-, 平川, 安村編, 第 2. 1 章, pp. 24-44, 共立出版, 1996.
- [2] 亀田恒彦, 山下雅史: 分散アルゴリズム, 近代科学社, 1994.
- [3] Microsoft Office PowerPoint2003
<http://www.microsoft.com/japan/office/powerpoint/prodinfo/default.msp>
- [4] G. W. Furnas. Generalized fisheye views. CHI'86, 1986.
- [5] M. H. Brown M. Sarkar. Graphical fisheye views of graphs. 1992.
- [6] 木村恒希, 猿渡孝志, 細川宜秀, 高橋直久: “ゴム伸縮メタファによる認知地図再生システムの実現方式”, 信学技報 HIP2003-12, pp.65-70, 2003.
- [7] 佐々木敏晃, 片山喜章, 高橋直久, “グラフ上の動画作成を可能にする動画記述言語と動画生成エンジンの設計および実現”, FIT2004 情報技術レター, LK-019, pp.299-302, 2004.
- [8] 高嶋章雄, 山本恭裕, 中小路久美代: “検索的データ分析のための時間的な概観と詳細の表現およびインタラクションに関する研究”, 情報処理学会論文誌, Vol. 44, No. 11, pp. 2767-2777, 2003.