

ブログからロボットの動作コンテンツを生成するシステム TENORIの提案

大隅 俊宏*¹

Toshihiro Osumi

大澤 博隆*¹

Hiroataka Osawa

藤本 健太*¹

Kenta Fujimoto

篠沢 一彦*²

Kazuhiko Shinozawa

桑山 裕基*¹

Yuki Kuwayama

今井 倫太*³

Michita Imai

野田 誠人*¹

Masato Noda

*¹慶應義塾大学大学院 理工学研究科

Graduate School of Science and Technology, Keio University

*²株式会社 国際電気通信基礎技術研究所

Advanced Telecommunications Research Institute International

*³慶應義塾大学 理工学部

Faculty of Science and Technology, Keio University

We propose a contents generation system from blogs named "TENORI" for a portable robot named "Blog Robot". Blog Robot presents blog contents by using verbal and non-verbal expression. TENORI has introductory essay generation function via the category informations and authoring tool which can embed gestures. Using TENORI, Blog Robot can communicate the blog entry to readers by speaking and using gestures, so it can present blog entry with realistic sensation.

1. はじめに

携帯電話やPDAに内蔵されたGPSを利用したサービスの普及によって、位置ベースのコンテンツの提示インタフェースの必要性が高まっている。しかし、[Ittemia 07] や [Rsstogps 07] といった位置情報を利用したコンテンツを提示するインタフェースは多く存在するが、コンテンツを作った人がその場で説明してくれているような臨場感のある形でのコンテンツの再生についての研究はほとんどなされていない。

そこで我々は、身体を用いて人間と同じように感情表現ができるロボットに着目した。ロボットを提示インタフェースに用いることによって、人間が説明してくれているような臨場感のある形でのコンテンツ再生を行うことができる。本研究ではコンテンツにブログを選び、ブログの記事をロボットがジェスチャと発話を用いて能動的にユーザに伝えることが可能となるTENORIシステムを開発した。ブログとは記事が時系列順に並び、それぞれの記事に対してコメントをつけることができるWEBサイトのことであり、身の回りの出来事や自分の意見を気軽に紹介できるコンピュータネットワーク上の媒体としてブログに対する一般ユーザの関心が高まっている。現在、ブログはインターネット上に非常に多く存在するため、新しくコンテンツを用意しなくて良いという利点がある。

ロボットにブログ記事の内容を読ませる際に、ただロボットがブログの内容を喋るだけでは読み手に臨場感を与られない。読み手に臨場感を与えるためには、ロボットが適切なタイミングで書き手の意図したジェスチャを行う必要がある。しかし、現在のブログ記事はロボットで再生されることを考慮されていないため、ブログの内容に相応しいジェスチャをロボットがすることができない。また、音声による記事再生だけでは、再生中に注意を十分に実世界に向けることができず、結果として臨場感が損なわれてしまう。

そこで本論文では、ブログオーサリングツールによって従来の顔文字と同じ感覚でロボットのジェスチャタグをブログの文章中に埋め込む。また動作コンテンツ生成システム TENORI

連絡先:

大隅 俊宏,

慶應義塾大学理工学部,

tosihiro@ayu.ics.keio.ac.jp

によって、ジェスチャタグからのジェスチャ生成とブログ記事の簡単な紹介文生成を行う。TENORIを用いることで、外観は普通のブログのままで記事にロボットのジェスチャを組み込むことが可能となる。

2. ロボットによるブログ記事の再生

2.1 ブログの再生にロボットを用いる有効性

ジェスチャは音声言語の理解を高めるのに効果的であり、[Berger 71]の研究によれば理解の正確性が約2倍になった。つまり、ジェスチャを交えた会話は文章を読むよりも状況を伝えやすく、書き手がその場で説明してくれているような臨場感を生むと考えられる。

また、[Shinozawa 05]の研究によれば、実世界上で動くロボットは画面内のCGエージェントに比べて実世界上に関する情報提示を行う際に、ユーザに意図を伝えるのに有利であるとされる。そのため、実世界の事柄について書かれたブログ記事の提示に関しても画面内でブログ記事を提示するよりも、ロボットを用いた方が読み手の注意を実世界に向けるには効果的であると考えられる。

2.2 ブログにおける臨場感

本研究のブログにおける臨場感は、書き手が書いた実世界のブログの記事内容を書き手がその場で説明してくれているような感覚とする。臨場感が高まれば、読み手がその実世界とブログの記事内容についてより興味を抱き、印象に残る可能性が高くなると考えられる。

ブログにおける臨場感を高めるには、ブログの適切なシチュエーションでのブログの再生の提案と書き手が意図した通りのタイミングでのジェスチャが必要であると考えられる。

2.3 BlogRobot

BlogRobotはブログの記事を再生するための携帯型小型ロボット(図1)である。BlogRobotはGPS機能を搭載したPDAと頭部と両手を持つ。頭部と両手の各ユニットはPDAからBluetoothによってコントロールされる。頭部と両手はそれぞれサーボモータによって動作し、頭部は6の自由度を持ち、両手はそれぞれ3の自由度を持つ。

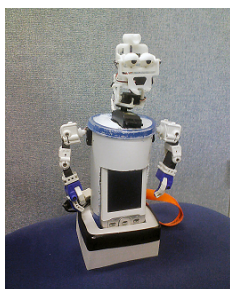


図 1: BlogRobot

BlogRobot は一定周期で GPS 情報を更新し、ブログ記事に含まれる jpeg 画像の GPS 情報と比較する。BlogRobot の GPS 情報に近い GPS 情報を持つ jpeg 画像をもつブログ記事があれば、読み手にブログ記事の存在を伝えて紹介し、記事の再生を開始する (図 2)。



図 2: BlogRobot 動作の流れ

2.3.1 BlogRobot とジェスチャ

BlogRobot はジェスチャコマンドを読み込むことで各種ジェスチャを行うことができる。BlogRobot が行えるジェスチャと対応するジェスチャコマンドを以下に示した。

- !!hatena ⇒ 首をかしげる (図 3)
- !!ojigi ⇒ おじぎ (図 4)
- !!kubihuri ⇒ 首を振る (図 5)
- !!namida ⇒ 悲しむ (図 6)
- !!warai ⇒ 喜ぶ (図 7)
- !!play: ⇒ 同意を求める (図 8)

3. TENORI による動作コンテンツの生成

我々はブログからロボットの動作コンテンツを生成するシステム TENORI を提案する。TENORI は次の 2 つの機能によってロボットの動作コンテンツを生成する。

- ジェスチャタグを埋め込み可能なオーサリングツールを持つブログシステム
- 埋め込まれたジェスチャタグからジェスチャコマンドを生成する機能

ジェスチャタグとは図 9 に示したようなタグであり、パソコン上で見るとジェスチャを示すアイコンが付与される以外は普通の文章として表示される。このジェスチャタグを TENORI を用いて変換し、ジェスチャコマンドを生成する。



図 3: 首をかしげるジェスチャ



図 4: おじぎジェスチャ



図 5: 首を振るジェスチャ



図 6: 悲しむジェスチャ



図 7: 喜ぶジェスチャ



図 8: 同意を求めるジェスチャ

ジェスチャタグ:
`どーも、大隅です`


パソコン上で見ると: どーも、大隅です

図 9: ジェスチャタグの例

図 9 に示したジェスチャタグを TENORI で変換すると図 10 のようなジェスチャコマンドが生成され、BlogRobot がおじぎジェスチャをしながら「どーも、大隅です」と発話する。これによりジェスチャとジェスチャに対応する文章の発話を同時に実行することができる (図 11)。

4. TENORI のシステム構成

TENORI の構成を図 12 に示す。ブログの記事はブログモジュール内のオーサリングツールによって生成される。ブログモジュールは単体では通常のブログとして動作しており、WEB 上で閲覧することも可能である。TENORI は BlogRobot から GPS 情報を受け取ると、BlogRobot の GPS 情報と近い GPS 情報を持つブログ記事を探し、記事があれば該当する記事を BlogRobot で再生するのに適した形に変換してブログ記事の紹介文を付加して BlogRobot へ送り返す。

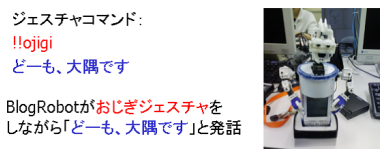


図 10: ジェスチャコマンドの例

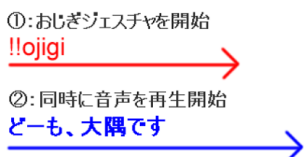


図 11: ジェスチャと発話のタイムライン

4.1 ブログモジュールの設計

ブログモジュールは他のモジュールから独立しており、単体でも通常のブログとして動作する。ブログモジュール内のオーサリングツールを用いることによって、ジェスチャタグを組み込んだブログ記事を生成する。

4.1.1 オーサリングツール

TENORI のオーサリングツールは従来のオーサリングツールで文字の強調、文字の色の変更といった機能を持たせていたボタンの欄に、同様の方法でジェスチャタグを埋め込むことが可能なボタンを追加したものである。書き手は自分の作成した文章中で、ジェスチャタグを組み込みたい部分を選択し、ジェスチャを追加するボタンを押すだけでブログ記事にジェスチャタグを組み込むことが可能となる。

4.2 TENORI の設計

TENORI は以下のモジュールを内部に持つ。

- GPS 取得モジュール
- 記事選択モジュール
- 前処理モジュール
- F-TOPIC モジュール

記事選択モジュールは BlogRobot から GPS 情報を受け取ると、GPS 取得モジュールを呼び出す。GPS 取得モジュール

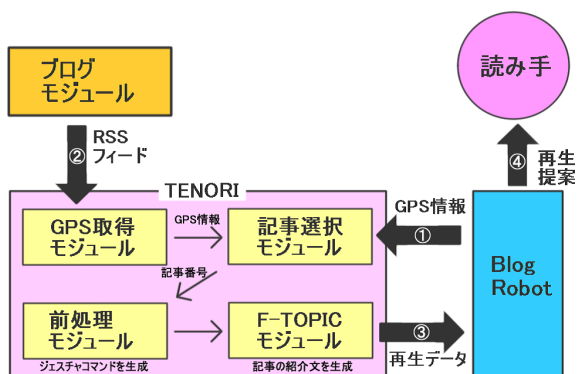


図 12: TENORI のシステム図

は登録されているブログの RSS フィードを取得し、ブログの記事に含まれる jpeg 画像から GPS 情報を取得する。記事選択モジュールは BlogRobot から受け取った GPS 情報と GPS 取得モジュールから得られたブログ記事の GPS 情報を比較する。BlogRobot の GPS 情報に近い GPS 情報を持つブログ記事があれば、前処理モジュールにブログの記事番号を送信する。前処理モジュールは送られてきた記事番号のブログ記事を取得し、ブログ記事の本文に含まれる読み上げとジェスチャに不要なタグの除去とジェスチャタグを利用したジェスチャコマンドの生成を行う。F-TOPIC モジュールでは、前処理部で処理されなかった日付、カテゴリ、編集者のデータからブログの簡単な紹介文を生成し、紹介文と本文を BlogRobot へ送信する。

5. 実装

5.1 ブログモジュールの実装

本研究ではブログモジュールに WordPress [WordPress](図 13) を使用する。WordPress はオープンソースのブログ/CMS プラットフォームである。ブログのオーサリングツールを図 14 に示した。オーサリングツールでは文章の範囲を選択し、ジェスチャタグ挿入ボタンを押すことでジェスチャタグが挿入される。



図 13: WordPress で作られた TENORI ブログ

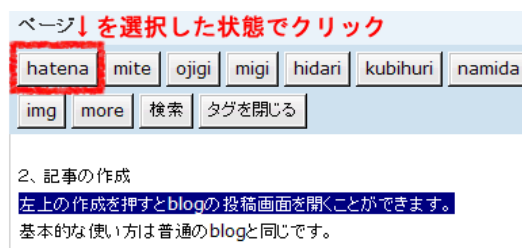


図 14: オーサリングツールとジェスチャタグの挿入方法

5.2 TENORI の実装

5.2.1 GPS 取得モジュール

RSS フィードから得られたブログ記事の本文の img タグが含まれる行を検索し、画像の URL を抽出する。抽出した画像の Exif 情報を解析し、GPS 情報を取得する。

5.2.2 前処理モジュール

前処理モジュールではRSS フィードで取得してきたブログ記事の本文に含まれるジェスチャタグをジェスチャコマンドに変換する。加えて、ジェスチャタグの終了と不要なタグを除去することにより、ジェスチャの開始ポイントと本文のみのデータが生成される。ジェスチャは種類ごとにジェスチャ時間が定められているため、タグの終了位置は BlogRobot の動作に影響を与えない。

5.2.3 F-TOPIC モジュール

(1) いつ, (2) 何について, (3) 誰が, (4) どのような記事を書いたかが分かるような紹介文を生成する。それぞれ RSS フィードの次の部分に対応させる。カテゴリが天気であった場合のみ、今日の天気についての文章を加える。

- 日付 ⇒ いつ (when)
- カテゴリ ⇒ 何について (what)
- 編集者 ⇒ 誰が (who)
- タイトル ⇒ どのような (how)

F-TOPIC モジュールによる紹介文の生成例を図 15 に示した。

```

日付:2009/01/18
カテゴリ:お店
編集者:Osumi
タイトル:たんたん

再生時の日付:2009/01/20
「一昨日、この近くのお店について
Osumi さんがたんたんって記事を書いてるよ。
再生してみる？」
    
```

図 15: 紹介文の生成例

6. ブログ記事の再生検証

図 13 の記事を再生した際の本文部分のタイムラインを図 16 に示す。図 16 より、ジェスチャとジェスチャに対応した文章の発話が同時に実行できていることが確認された。

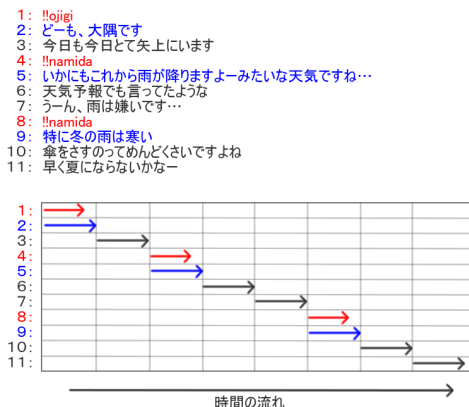


図 16: 動作のタイムライン

7. 今後の課題

7.1 一般的なブログ記事の再生

本論文では専用に設計されたオーサリングツールを用いてブログ記事にジェスチャタグを埋め込んだ。そのため、一般的なブログからはジェスチャコマンドを生成することができない。今後、一般的なブログ記事に対してもジェスチャと発話による再生を行うために、顔文字や感嘆詞といったタグからジェスチャコマンドを生成できるようにする必要がある。

7.2 読み手とのインタラクション

本論文の BlogRobot は TENORI から与えられた記事を再生することしかできない。今後、読み手の反応によって挙動を変えたり、読み手の感想をブログにフィードバックする機能を実装する必要がある。

8. まとめ

本研究では身体表現を伴う発話を用いてブログ記事を読み手に伝える小型ロボット BlogRobot のためのブログからロボットの動作コンテンツを生成するシステム TENORI を提案した。BlogRobot でブログ記事を再生して読み手に臨場感を与えるためには書き手が意図したタイミングで BlogRobot のジェスチャを生成する機能が必要であった。TENORI では (1) 書き手が簡単にジェスチャタグをブログに組み込むことが可能となるオーサリングツールと (2) ジェスチャタグからジェスチャを生成する機能を実装することによってこれらの問題解決を図った。TENORI を実装したシステムの動作確認を行い、オーサリングツールによってブログ記事に簡単に BlogRobot のジェスチャタグを組み込めることを確認した。次に、TENORI によってジェスチャタグが埋め込まれたブログ記事からジェスチャコマンドを含んだ記事データへ変換できることを確認した。最後に、BlogRobot が TENORI から送られた記事データを発話とジェスチャを用いて再生できることを確認した。

参考文献

[xypoint 00] xypoint: Nomad,
<http://www.xypoint.com/> (2000).

[AU 06] AU: EZ メール読み上げ,
http://www.au.kddi.com/ez_mail_yomiage/ (2006).

[Ittemia 07] 日立製作所: Ittemia,
<http://ittemia.jp/> (2007).

[Rsstogps 07] アイティア: RSStoGPS,
<http://www.aitia.co.jp/solution.html> (2007).

[WordPress] WordPress: <http://ja.wordpress.org/>.

[Shinozawa 05] Kazuhiko Shinozawa, Futoshi Naya, Junji Yamato and Kiyoshi Kogure: Differences in Effect of Robot and Screen Agent Recommendations on Human Decision-Making. In *International Journal of Human-Computer Studies*, 267/279 (2005).

[Berger 71] Kenneth W.Berger and Gerald R.Popelka: Extra-Facial Gestures in Relation to Speechreading. In *Journal of Communication Disorders*, 302/308 (1971).