

Web ブラウザ上で MMI システムを実行可能にする JavaScript ライブラリ MMI.js の提案

菊地泰己^{†1} 桂田浩一^{†1} 入部百合絵^{†1} 新田恒雄^{†1†2}

本論文では、一般的な Web ブラウザ上でマルチモーダル対話システムを可能にするための JavaScript ライブラリ MMI.js を提案する。MMI.js は、逐次的、同時的、択一的なマルチモーダル入出力の制御、入出力のタイミング制御、エージェントによるプレゼンテーションの各機能をサポートしている。MMI.js の有用性を確認するために、本研究室で開発している英語の発音訓練ソフトにマルチモーダルインターフェースを組み込んだ。これにより複数の入出力手段を組み合わせさせた複雑な対話を容易に記述できることを確認した。

MMI.js: A Proposal of JavaScript library enables to execute MMI system on Web browsers

TAIKI KIKUCHI^{†1} KOUICHI KATSURADA^{†1}
YURIE IRIBE^{†1} TSUNEO NITTA^{†1†2}

This paper proposes a JavaScript library called “MMI.js” which enables us to use multiple modalities on web browsers. This library supports sequential multimodal inputs/outputs, simultaneous multimodal inputs/outputs, alternative multimodal inputs/outputs, synchronization of multimodal inputs/outputs and gestures given by the dialogue agents. To show usefulness of this library, we embedded multimodal interaction into a pronunciation training application for Japanese students. Through the development of this application, we confirmed the library makes it easy to describe combination of multiple inputs/outputs appearing in complicated interaction.

1. はじめに

Siri[1]やしゃべってコンシェル[2]といった電子秘書サービスの登場により、対話エージェントの利用が活発化している。ユーザは対話エージェントを通して様々なサービスを利用することができる。しかしこれらのサービスは、特定の端末のみにおいて提供されており、コンテンツを開発するための環境も一般には公開されていない。

そこで本論文では、一般の技術者が慣れ親しんだ環境でこうしたアプリケーションを開発できるよう、JavaScript 用マルチモーダル対話記述ライブラリ MMI.js を提案する。JavaScript は一般的な Web ブラウザ上で動作可能なため、開発者はユーザの利用環境を考慮することなくアプリケーションを開発することができる。MMI.js は、マルチモーダル入出力機能、入出力のタイミング制御機能、対話エージェントによる様々なジェスチャーを行う機能を提供しており、複数のモダリティやタイミング制御を組み合わせさせた複雑な対話を記述可能にする。以下、関連技術、ライブラリの概要、サンプルアプリケーションについて述べる。

2. 関連技術

まず、本研究に関連する技術とその機能を表 1 に示す。表 1 の各技術のうち XISL[3], SMIL[4], MPML[5], TVML[6] は XML ベースの記述言語であり、他のものは JavaScript を用いた技術である。XML ベースの記述言語には特定の

機能に特化したものが多く、XISL ではマルチモーダル対話に頻繁に現れる逐次的、同時的、択一的なマルチモーダル入出力を記述することが可能である。SMIL はマルチメディアコンテンツの同期制御をサポートする。指定したコンテンツを指定したタイミングで再生することが可能である。MPML, TVML を利用すると、エージェントを用いた出力を記述することができる。エージェントの様々な動作を用意することで効果的なプレゼンが可能になる。

一方、JavaScript を用いた技術のうち Microsoft Agent[7] は Web 上でのエージェントのアニメーションを制御するための技術である。また、Google Speech API[8]や w3voice システム[9]は Web 上での音声入力インターフェースを実現する。JavaScript を用いることの利点は、開発者が新たな言語を習得する必要がないことである。また、一般的なブラウザで動作可能なため、ユーザがソフトウェアをインストールすることなく利用できるという利点も持つ。

3. ライブラリの概要

前節で述べた関連技術がサポートする機能は、マルチモーダル対話の記述を実現する上で非常に重要な機能である。しかし、表 1 に示す通りこれらすべての機能をサポートしているものは無い。本研究では、それらの機能を全て満たす JavaScript ライブラリを提供するために、1. 音声を含むマルチモーダル入出力機能、2. メディアの同期制御機能、3. エージェントの様々な動作アニメーション表示機能を満たすライブラリを提供する。

まず 1. の機能を提供するために、XISL を参考に、逐次的、同時的、択一的な入力、逐次的、同時的な出力を行

^{†1} 豊橋技術科学大学
Toyohashi University of Technology
^{†2} 早稲田大学
Waseda University

表 1 関連技術がサポートする機能の一覧
Table 1 Features supported by related technologies

機能	XISL	SMIL	MPML	TVML	Microsoft Agent	Google Speech API	w3voice
マルチモーダル入力	○	×	△	×	○	○	○
マルチモーダル出力	○	○	○	○	○	×	×
メディア同期機能	△	○	○	○	×	×	×
エージェントの動作	○	×	○	○	○	×	△
インストールが不要	○	×	×	×	×	○	○
JavaScript での記述	×	×	×	×	○	○	○

○：対応 △：一部制限あり ×：非対応

表 2 ライブラリの概要

Table 2 Outline of the library

分類	関数名	処理内容
マルチモーダル 入力	mmi.seqInput	逐次的な入力を受付ける
	mmi.parInput	同時的な入力を受付ける
	mmi.altInput	択一的な入力を受付ける
マルチモーダル 出力	mmi.seqOutput	逐次的な出力を行う
	mmi.parOutput	同時的な出力を行う

図 1 入出力の記述例

Figure 1 Descriptive examples of inputs/outputs

<pre>//Alternative input mmi.altInput({ "type": "click", "match": "agent", "options": { "begin": 1000, "dur": 10000 } }, { "type": "speech", "match": ". /grammar", "options": { "begin": 1000, "dur": 10000 } });</pre>	<pre>//Parallel output mmi.parOutput({ "type": "agent", "event": "speech", "text": "商品を選択してください" }, { "type": "agent", "event": "gesture", "gesture": "point" });</pre>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

う関数 mmi.seqInput, mmi.parInput, mmi.altInput, mmi.seqOutput, mmi.parOutput を用意した。表 2 にその一覧を示す。次に 2. の機能を提供するために、SMIL のメディア同期機能を参考にして入出力の開始や終了等のタイミング制御を行うための属性 17 種類を用意した。更に、3. の機能を提供するために、MPML を参考にして“お辞儀”や“指さし”等のエージェントの動作 62 種類を用意した。2. および 3. の機能は表 2 の関数への引数として記述可能である。入出力の記述例を図 1 に示す。

4. サンプルアプリケーション

提案したライブラリの有用性を確認するために、本研究室で開発している発音訓練ソフト[10]にマルチモーダル対話機能を組み込んだ。図 2 にスクリーンショットを示す。ユーザは、音声またはマウス入力によりアプリケーションを操作することができる。図 2 画面中央に表示されている対話エージェントは、ユーザの入力や英単語の発音の結果に合わせて発話やジェスチャーを行う。提案したライブラリを用いることで、従来は難しかったアプリケーションの動作結果に応じたエージェントの振る舞いや、タイミング



図 2 発音訓練ソフトのスクリーンショット

Figure 2 Screenshot of pronunciation training application

を指定したメディアの同期出力を容易に組み込むことが可能なことを確認した。

5. まとめと今後の課題

本稿では、Web ブラウザ上でマルチモーダル対話を可能にする JavaScript ライブラリ MMI.js を提案した。また、発音訓練ソフトにライブラリを組み込むことにより、アプリケーションと連動したエージェントの振る舞いや、タイミングを制御したメディアの同期出力を容易に記述できることを確認した。今後はスマートフォンへの対応に取り組むと共に、エージェントを可変にする等の拡張を検討したい。なお、本ライブラリは Web 上での公開を予定している。

参考文献

- 1) <http://www.apple.com/jp/ios/siri/>
- 2) http://www.nttdocomo.co.jp/service/information/shabette_concier/
- 3) 桂田浩一, 中村有作, 山田 真, 山田博文, 小林 聡, 新田恒雄: MMI 記述言語 XISL の提案, 情報処理学会論文誌, Vol. 44, No. 11, pp. 2681-2689 (2003).
- 4) <http://www.w3.org/AudioVideo/>
- 5) 筒井貴之, 石塚 満, キャラクタエージェント制御機能を有するマルチモーダル・プレゼンテーション記述言語 MPML, 情報処理学会論文誌, Vol. 41, No. 4, pp. 1124-1133 (2000).
- 6) Hayashi, M., Ueda, H. and Kurihara, T.: TVML (TV program Making Language) - Automatic TV Program Generation from Text-based Script -, ACM Multimedia'97 State of the Art Demos (1997).
- 7) <http://www.microsoft.com/products/msagent/main.aspx>
- 8) <http://www.google.com/intl/ja/chrome/browser/>
- 9) 西村竜一, 三宅純平, 河原英紀, 入野俊夫: 音声入力・認識機能を有する Web システム w3voice の開発と運用, IPSJ SIG Technical Report, Vol. 2007-SLP-68-3, pp. 13-18 (2007).
- 10) 森 拓郎, 入部百合絵, 桂田浩一, 新田恒雄: 発音訓練のための調音特徴に基づく IPA 母音図へのリアルタイム表示, IPSJ SIG Technical Report, Vol. 2011-SLP-89, No. 15, pp. 1-6 (2011).