Vol.2017-GN-100 No.19
Vol.2017-CDS-18 No.19
Vol.2017-DCC-15 No.19
2017/1/20

音声認識・合成技術を用いた平仮名学習コンテンツ

荒木千尋[†1]　松河剛司[†1]

概要：筆者らは大語彙連続音声認識エンジン Julius と音声合成技術である VOCALOID を使用した平仮名学習コンテンツを開発した. このコンテンツでは, ユーザーが発生した日本語(単語)を Julius によって PC に入力し, 単語を 3DCG で平仮名 1 文字ずつ立体化し, モーションキャプチャデバイスを用いて 3DCG 平仮名モデルに擬似的に触れることで操作し組み合わせることで別の単語を作り, 組み合わされた単語が VOCALOID によって読み上げられる. このコンテンツを使用することで発声した単語が立体化されることで文字の形を学習でき, 読み上げられた単語を聞くことで文字の読み方を学習することができる.

Developing The Hiragana Learning Content by using Speech Recognition and Speech Synthesis

CHIHIRO ARAKI[†1]　TSUYOSHI MATSUKAWA[†1]

1. はじめに

iOS の Siri や Google の OK Google のようにスマートフォンでも音声認識技術が扱えるようになり，音声認識は私たちにとって身近な技術になりつつある．オープンソースで公開されている大語彙連続音声認識エンジン Julius[1]も Siri や OK Google と同様にスマートフォンに実装できるほど軽量かつ高性能な音声認識エンジンである. 人間から PC への入力技術である音声認識の技術開発が進む一方で，PC から人間の耳への出力技術となる音声合成技術も YAMAHA の Vocaloid[2]の登場と You tube やニコニコ動画などでの Vocaloid を使用した作品の公開で身近なものとなっている．筆者らはこれらの音声の入力，音声の出力技術を用いて新しい日本語学習コンテンツは制作できないかと考え, 本稿の内容である平仮名学習コンテンツを開発した.

提案コンテンツではユーザーは音声認識技術により「喋る」を行い，音声合成技術により「聞く」を行い，VR 機器を使用することにより「見る」「組み合わせる」ことができるコンテンツとなっている．VR 機器を使用することにより，体験が現実に近いものとなり学習効果が上がることが期待できる.

2. 平仮名学習コンテンツの概要

本論文で提案する平仮名コンテンツは，処理用の PC とユーザーが使用する HMD とワイヤレスコントローラー，ユーザーの位置と動きを取得する 2 基のベースステーションによって構成される.

ユーザーは VR 機器の機能であるモーションキャプチャによって頭の位置やコントローラーを操作し，図 1 に示す

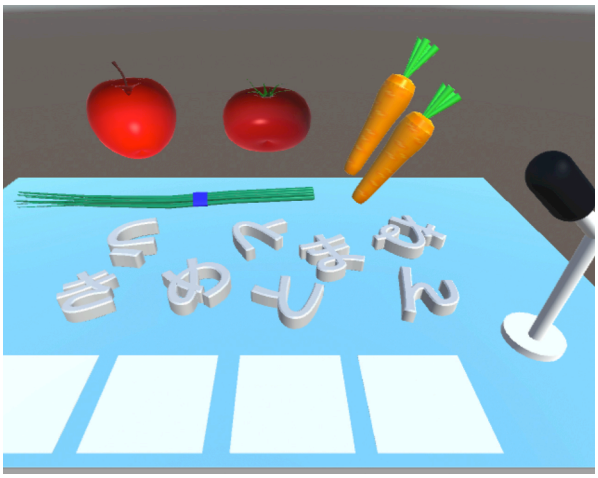

図 1　提案する平仮名学習コンテンツ

画面内の食べ物や動物などのカテゴリから場にある平仮名モデルを組み合わせて指定されたターゲットモデルの名前を作る. 場にある食べ物や動物などのターゲットモデルと，組み合わされた名前の単語を比較し，指定されたターゲットモデルの名前と一致していれば，一致したターゲットモデルはクリアになる.

場にある平仮名モデルとターゲットモデルは，コントローラーで操作して接触することによりその平仮名モデルとターゲットモデルの名前が読み上げられるため，どの平仮名モデル同士を組み合わせるか，どのターゲットモデルに関連した平仮名モデルか，など音によって考えるヒントが用意されている.

場にある平仮名モデルが減少し組み合わせることができなくなった場合は，ユーザーが任意で発声した単語を分解することによって，使用する平仮名モデルを増やすことができる. ユーザーはマイクに向かって発声し，処理用の PC

†1　愛知工業大学大学院経営情報科学研究科

Vol.2017-GN-100 No.19
Vol.2017-CDS-18 No.19
Vol.2017-DCC-15 No.19
2017/1/20

がユーザーの発声した単語を認識する.認識された単語を，登録された平仮名の情報と比較し，単語に含まれる平仮名と該当する平仮名モデルを場に生成する．ユーザーは，生成された平仮名モデルを分解して既に場にあった平仮名モデルと組み合わせて新しい単語を作ることができる．

3. 実現手法

提案する平仮名コンテンツでは，音声認識技術，音声合成，VR 技術（ヘッドマウントディスプレイとモーションキャプチャ）を用いて実現している．

3.1 音声認識

ユーザーが発声した単語の認識は, Julius[1]が行う. Julius とは，オープンソースの汎用大語彙連続音声認識エンジンのことであり，単語辞書や言語モデル・音響モデルなどの音声認識の各モジュールを組み替えることで，小語彙の音声対話システムからディクテーションまで様々な幅広い用途に応用できる．

しかし，Julius は発声された「たまご」などの単語は認識されやすいが,「あ」などの一文字だけの発声では誤認識が多く認識がされにくい．そこで本研究では，認識されやすい単語から認識し，分解して一文字ずつ関連付けられた平仮名モデルの表示処理を行っている．

さらに，Julius からの結果を平仮名で取得するために，認識される単語は優先的に平仮名で表示させる必要がある．例えば，ユーザーが発声した「りんご」という単語を Julius が認識した場合,「林檎」と表示され，単語中に該当する平仮名がないため平仮名モデルの出力ができない．そこで，平仮名表記の単語の優先度を高く登録することで，漢字表記やカタカナ表記での結果取得を防いでいる．本研究では Unity 用に公開されている Julius-Client-for-Unity[3]を用いた．

3.2 音声合成

ユーザーが操作するアバターの手が平仮名モデルに触れた場合や，ターゲットモデルに触れた場合に PC のスピーカーから合成音声が発生される．合成音声の実装には VOCALOID SDK for Unity[4]を使用し，音声ライブラリにはランタイム版ライブラリ unity-chan!を使用した．

平仮名モデルにはそれぞれの文字に対応する発生内容を割り当てており,「あ」の平仮名モデルに触れると，触れている間は「あ」を発声し続ける．その際「あ」「あ」「あ」と区切って連続で発声するのではなく「あー」と音を伸ばして発声する．ターゲットオブジェクトに触れた際は，そのターゲットオブジェクトの日本語読みを 1 度だけ発声する．例えば「とまと」のターゲットオブジェクトに触れた際は,「とまと」と 1 度だけ発声する．触れたアバターの手をターゲットオブジェクトから離し，繰り返し触れることで何度でもターゲットオブジェクトの日本語発声を確認することができる．VOCALOID SDK for Unity では音程や発声の強弱も設定できるため，ターゲットオブジェクトを触れた際に発生される単語は出来る限り日本人が発声する音程や強弱に近くなるように単語ごとに設定を行っている．

3.3 VR 機器（HMD・モーションキャプチャ）

ユーザーの操作および位置情報の取得には HTC Vive[5]を使用している. Vive とは HTC 社から発売されている VR・モーションキャプチャデバイスであり，ヘッドマウントディスプレイ，左右それぞれ 1 台ずつのコントローラー，2 基のベースステーションで構成されている．ベースステーションにはからは赤外用 LED が搭載されており, その LED パターン信号をヘッドマウントディスプレイやコントローラーに搭載されているトラッキングセンサが受信することでヘッドマウントディスプレイやコントローラーの位置情報や回転情報を取得している．また 2 台のベースステーションを利用しているため，ベースステーション間（約 2〜3m 四方）であれば非常に高い精度で位置情報・回転情報の取得が行える．またヘッドマウントディスプレイは立体視にも対応している．図 2 に使用した HTC Vive の機材を示す．

本研究ではユーザーはヘッドマウントディスプレイを装着し，左右の手それぞれにコントローラーを持ってコンテンツを体験する．ヘッドマウントディスプレイおよびコントローラーは前述した Vive の位置情報取得機能を用いてそれぞれの位置や回転の情報を取得している．ヘッドマウントディスプレイが立体視に対応している為, 3D で作成し

図 2　使用した VR 機器（HTC Vive）
左奥：ヘッドマウントディスプレイ
右奥：ベースステーション
手前：コントローラー

ている平仮名モデルやターゲットオブジェクトは立体感を

Vol.2017-GN-100 No.19
Vol.2017-CDS-18 No.19
Vol.2017-DCC-15 No.19
2017/1/20

持った状態で表示することができる．平仮名モデルやターゲットオブジェクトに触れる場合はコントローラーをその対象となる平仮名モデルやターゲットオブジェクトに触れるだけでイベント処理が行われ，発声が行われる．平仮名モデルを掴んで動かしたい場合はコントローラーのトリガーボタンを押すことで掴むことができ，トリガーボタンが押され続けている間は平仮名モデルを掴んだ状態が継続する．

3.4 CG の表示

本研究で使用される平仮名モデルは Blender で作成しており，食べ物や動物などのターゲットモデルは Autodesk Maya を使用して作成している．ターゲットモデルの例を図 3 に示す．「や行」と「わ行」の「い」「え」および鼻濁音を除き，濁音・半濁音を含む 76 個の平仮名モデルを登録している．平仮名モデルは平仮名の情報と関連付けられており，分解された一文字ずつの平仮名と比較し，文字同士が一致した場合に平仮名モデルが呼び出される．平仮名のモデルの例を図 4 に示す．

ユーザーが組み合わせた単語は，解答枠に配置することで単語として結合される．平仮名モデルに関連付けられている一文字ずつの平仮名の情報から，一文字ずつの平仮名を結合し単語にする．結合された単語を正しい解答に設定されている単語と比較し，正解の場合は，ユーザーが解答枠に配置した平仮名モデルを消去し，ターゲットモデル上に読み仮名の書かれたラベルが表示され，クリアとなる．

4. 実験

提案コンテンツを実装し，以下の環境で実験を行った．使用した機材は VR 用ノート型 PC（マウスコンピュータ社製，NextGear-Notei5702PA2，Windows8.1 64bit，IntelCorei7，RAM 32G，NVIDIA GeforceGTX970M），HTC Vive である．

被験者には椅子に座った状態で HTC Vive のヘッドマウントディスプレイを頭に装着し，左右の手にコントローラーを持たせた．Vive のベースステーションは被験者の左前方 1mの場所に 1 つ，右後ろ 1mの場所にもうひとつを設置した．事前に Vive のキャリブレーションを実施し，実験空間内でヘッドマウントディスプレイとコントローラーを正しく認識することを確認して実験を行った．腕を休ませることができるように被験者の前には机を設置した．実験の様子を図 5 に示す．

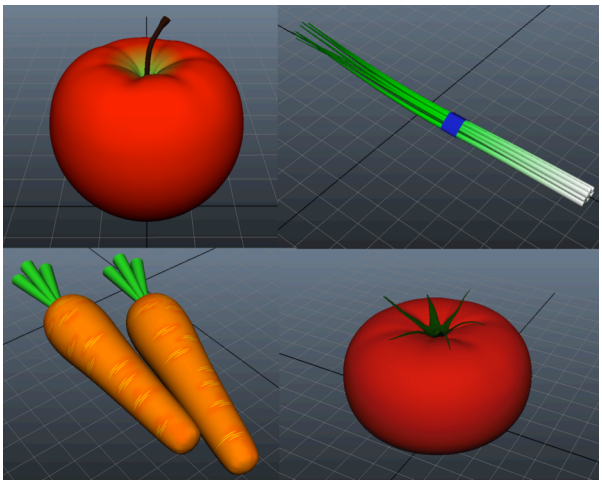

図 3　ターゲットモデルの例
（りんご，ねぎ，にんじん，とまと）

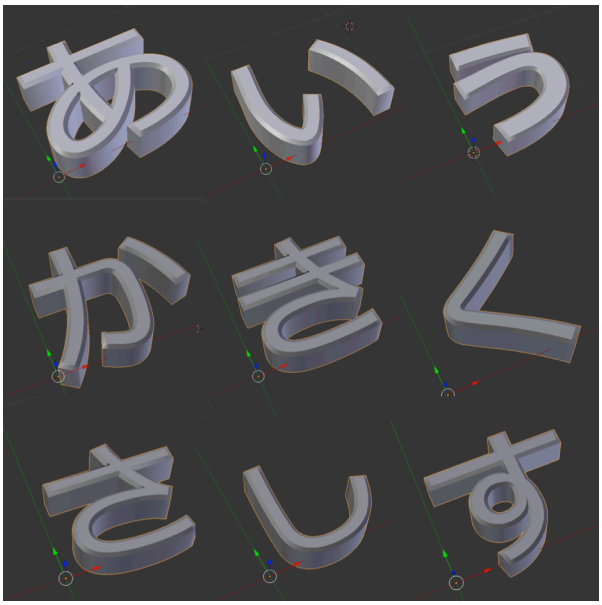

図 4　平仮名モデルの例

図 5　実験の様子

Vol.2017-GN-100 No.19
Vol.2017-CDS-18 No.19
Vol.2017-DCC-15 No.19
2017/1/20

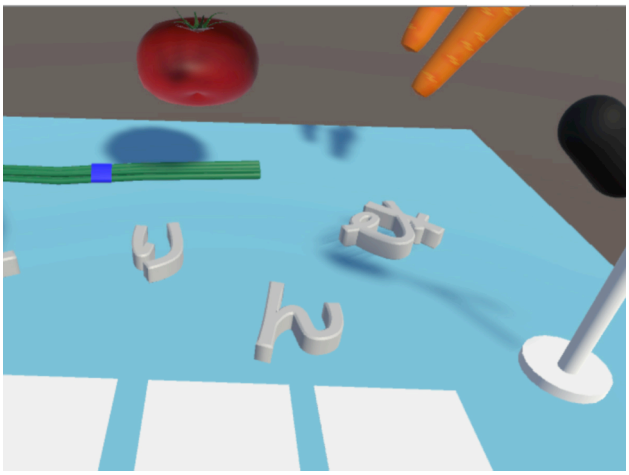

図 6　被験者の見ている画面

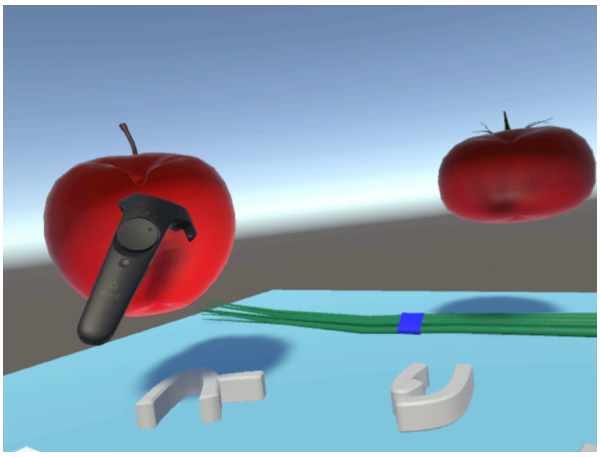

図 7　「りんご」モデルに触れている様子

コンテンツが実施されると被験者の見ている画面にはターゲットオブジェクトである「りんご」「とまと」「ねぎ」が浮かんだ状態で表示されており，またすぐ前方には机モデルがあり，その上に平仮名モデルである「り」「ん」「と」「ね」が配置されている．被験者の見ている画面を図 6 に示す．

更にマイクモデルと解答する際に文字を配置するための解答枠が見て分かる状態になっている．

被験者はこの状態で，「ターゲットオブジェクトに触れる」，「平仮名オブジェクトに触れる」，「平仮名オブジェクトを掴む」，「マイクに触れる」の 4 つの行動をすることができる．

実験ではまずターゲットモデルのひとつである「りんご」モデルに触れることを行った．「りんご」モデルに触れるために左手に持ったコントローラーを VR 空間上の「りんご」モデルに近づけて，モデルにコントローラーが触れるように操作を行った．コントローラーが「りんご」モデルに触れると，コントローラーモデルが「りんご」モデルを透過し，「りんご」モデルは特に動くことはなかったが，合成音声で「りんご」と発声が行われた．図 7 に「りんご」モデルに触れている様子を示す．

次に平仮名モデルの「り」モデルにコントローラーで触れてみた．「り」モデルに触れると「り」モデルはコントローラーモデルと接触し，「り」モデルは押されるようにして動いた．またターゲットモデルである「りんご」モデルに触れたときとは違い，コントローラーが「り」モデルを透過することはなかった．「り」モデルにコントローラーが接触している間，「りー」と合成音声が発声された．図 8 に「り」モデルに触れている様子を示す．

「り」モデルに触れている状態のまま，触れている側の手に持っているコントローラーのトリガーボタンを押してみた．すると「り」モデルはコントローラーの動きに追従

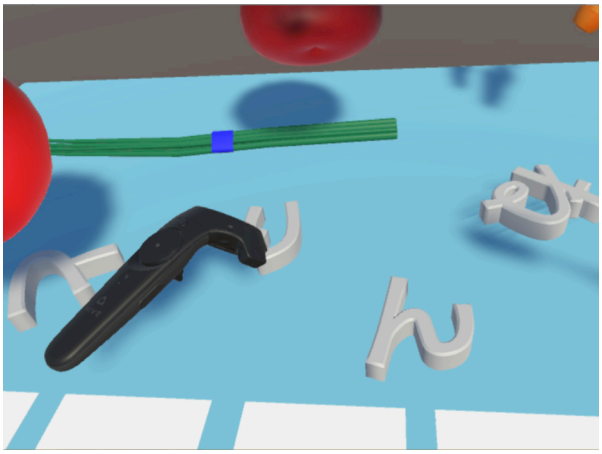

図 8　「り」モデルに触れている様子

図 9　「ん」を配置している様子

Vol.2017-GN-100 No.19
Vol.2017-CDS-18 No.19
Vol.2017-DCC-15 No.19
2017/1/20

するようになり，掴んでいる状態となった．「り」モデルを掴んでいる間，「りー」と合成音声が発声された．「り」モデルを掴んだまま，解答枠の一番左側に配置し，トリガーボタンを離した．同様に「ん」モデルを掴み，解答用の文字置き場の左から 2 番目の位置に配置した．「ん」モデルを掴んでいる間，「んー」と発声されていたが「ん」と発声していることが認識できる声であった．図 9 に配置している様子を示す．

次にマイクモデルに触れた．マイクモデルに触れた状態で「たまご」と自分の声で発声すると，前方の机の上方に「た」「ま」「ご」のモデルが現れ，そのまま机の上に落ちた．図 10 に発声から文字が表示される流れを示した．

「ご」のモデルを掴み，解答枠の左から 3 番目の位置に配置すると，「りんご」と合成音声の発生が行われ，ターゲットモデルの「りんご」モデルの前に「りんご」と文字が表示された．図 11 に解答後のターゲットモデルを示す．

ひとつのターゲットモデルの解答を示すと，そのターゲットモデルへの解答，今回の実験の場合「りんご」は，以後，解答枠に置いても正解とされず，特に変化はおきないようになっている．

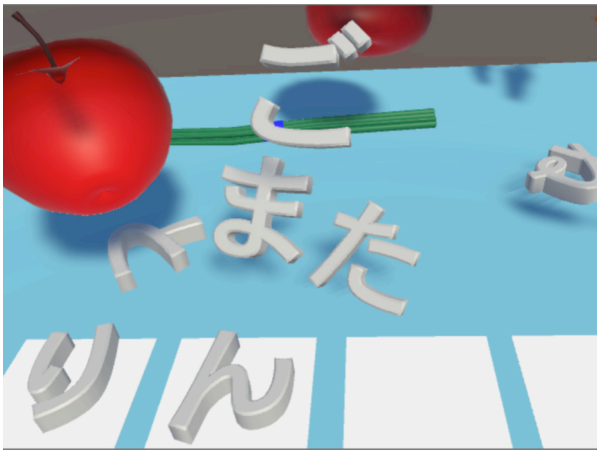

図 10　発声から文字が表示される様子

図 11　解答後のターゲットモデル

5. まとめ

本論文では，音声認識・音声合成技術および VR を利用した平仮名学習コンテンツを開発した．

このコンテンツでは，指定されたターゲットモデルの名前を，場に提示された平仮名モデルをひとつひとつ組み合わせて単語を作っていく．単語を組み合わせる過程で，ターゲットモデルや平仮名モデルに触れることで合成音声による文字や単語を「聞く」こと，足りない文字を増やすためにマイクに向かって日本語を発声することで「喋る」こと，VR 空間上に表示される平仮名モデルを「見る」，「組み合わせる」ことができ，文字の形や読み方を学習することができる．

しかしながら多くの HMD のガイドライン上では，年齢制限により低年齢を対象に HMD を使用した実現が困難である．そこで，今後は本コンテンツの改良とともに，VR コンテンツは留学生の勉強用に開発し，別途，未就学児童や小学校低学年を対象としたコンテンツとして机上へのプロジェクションとハンドキャプチャーを行える Leap Motion を利用し代替可能なシステムを開発予定である．

[1] “Julius“. http://julius.osdn.jp/, (参照 2016-10-13).
[2] “Vocaloid“. https://net.vocaloid.com/, (参照 2016-10-13).
[3] “Julius-Client-for-Unity“. https://github.com/SavantCat/Julius-Client-for-Unity, (参照 2016-02-20).
[4] “Unity with VOCALOID “. http://business.vocaloid.com/unitysdk/, (参照 2016-10-13).
[5] “HTC Vive”. https://www.vive.com/jp/, (参照 2016-10-13).