VR空間での音声認識を利用した2変数関数描画システム VVPlotの開発とその評価

新井 崇之[1]　大西 建輔[1]

概要：一般に多変数関数の概形は, その数式を見ただけで理解できるものではない. 専用の数式処理ソフト等を用いれば 2 変数関数のグラフを描画できるが, これらのソフトは有料であったり独自の書き方を学ぶ必要がある. そこで, 近年普及しつつあるヴァーチャルリアリティ機器と音声認識を用いて, 数学が苦手な人でも直感的な理解を促す 2 変数関数のグラフを描画するシステム “VVPlot”を作成した. 本システムは授業の理解を深めたり, 数学が苦手な人に興味を持ってもらうきっかけとなることを目指して作成している. 本稿では VVPlot の概要を説明すると共に, ユーザによる評価について述べる.

1. はじめに

ヴァーチャルリアリティ (以下 VR と呼ぶ) は, コンピュータとシミュレーション技術を用いてコンピュータによって作られた 3 次元空間を, 視覚や聴覚などの感覚を通じてあたかも現実のように体験できる技術である. VR はここ数年でよく耳にする単語になったが, VR 技術が実用段階となったのは 1990 年頃に NASA が開発した VR システムである. このシステムは宇宙空間を作り, 訓練できるように開発されている [1]. 2000 年台になると, クリスティ・デジタル・システムズ社による HoloStage[*1]という新しい形の VR システムが登場する. これは床面・正面・側面に設置されたスクリーンに複数の特殊なプロジェクターで画面を投映し, ユーザーがそのスクリーン上に立つ事で VR 空間に入り込み, 没入感を得られる機器である. これらの VR 機器は企業や大学などしか持てない高額かつ大型なものであった. 2012 年に, パルマー・ラッキーがクラウドファンディングにより資金を集め, ヘッドマウント型 VR デバイス Oculus Rift[*2]を開発した. それに続き HTC VIVE[*3]や PlayStation VR[*4]などの VR 機器が続々と登場した. これらの価格は, 発売当初は 10 万円以上のものであったが, 徐々に値下げをおこない, 個人でも購入できるようになってきている. また, スマートフォンを利用して VR 体験ができる Daydream[*5]などもあり, 今後はより VR 環境が普及していくことが考えられる.

我々はこの VR を教材として利用することを考えた. 教材として作られた VR アプリケーションとして, VR 空間で天体を間近に観察することができるアプリケーション [2] がある. また, 我々は物理の授業向けに放物運動が体感できるアプリケーション [3] を開発した. 次に, 我々は, VR 空間で 2 変数関数を描画することを考えた. 1 変数関数の場合は, 増減表を用いてグラフの概形を知ることができ, 関数のグラフを描くことができる. 一般に多変数関数の概形は, 数式を見るだけで理解できるものではない. Mathematica, Maple のような数式処理システムや, GeoGebra, gnuplot のようなグラフ描画ソフトを用いると, 数式を入力し 2 変数関数のグラフを描画できる. これらのソフトウェアは有料であったり, 専用の書き方を学ぶ必要があり, 初学者や数学を苦手とする人にとっては扱うのが難しい.

本研究では入力として音声認識を用いることで初学者や数学が苦手な人にも優しいシステムを作成した. 本システムを用いることで数学の授業の理解だけでなく, そういった関数を知らない学生や生徒に体験させる事で, 数学に興味を持ってもらう事を目標としている. また, VR や音声認識などの新しい技術にも触れてもらい, これらにも関心を持ってもらうことも目的の 1 つである.

2. 2 変数関数描画システム:VVPlot

2.1 概要

Voice VR Plot(以下 VVPlot と呼ぶ) は Unity[*6]で開発

[1] 東海大学 理学部 情報数理学科
Department of Mathmatical Sciences, Tokai University

[*1] https://www.christiedigital.com/en-us/3d/virtual-simulation/virtual-reality-systems/holostage
[*2] https://www.oculus.com/rift/
[*3] https://www.vive.com/jp/
[*4] https://www.jp.playstation.com/psvr/
[*5] https://vr.google.com/intl/ja_jp/daydream/smartphonevr/
[*6] https://unity3d.com/jp

© 2019 Information Processing Society of Japan

をおこない, そのプラットフォームは HTC VIVE (以下 VIVE と呼ぶ) とした. VVPlot のシステム図を図 1 に表す. VVPlot は音声を入力としている. ユーザーが数式を発声すると コンピュータ上に音声データが生成される. このデータを IBM Watson の音声認識システム Speech to Text に送り, 認識された文字列を得る. この文字列を VVPlot で処理し, 数式とする. この数式を元に VVPlot は描画をおこなう. VIVE コントローラーでの操作は次のようにおこなう. トラックパッドでユーザーの見ている方向を基準として前後左右の移動をおこなう. トリガーで上下の移動をおこなう. グリップボタンで, VR 画面のボタンをクリックをする. メニューボタンを押すと初期位置に戻る. 図 2 は実際の初期画面である.

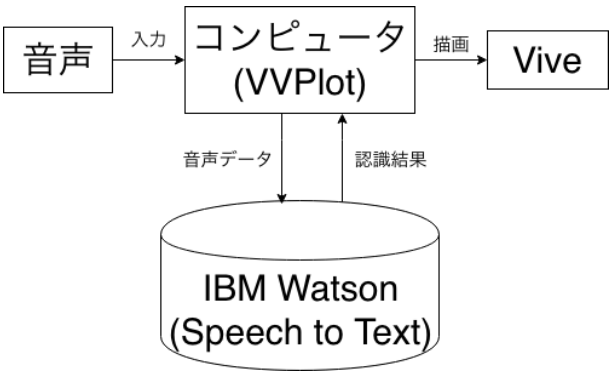

図 1 VVPlot のシステム図

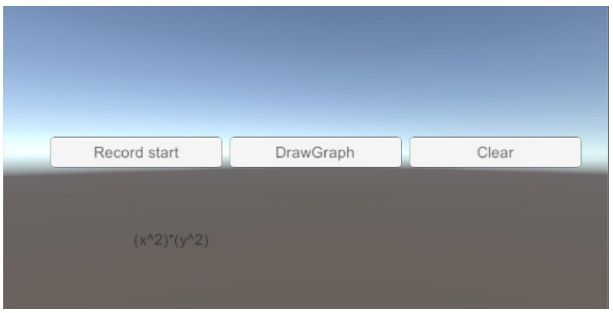

図 2 VVPlot の初期位置

初期画面では, 左から `Record start`, `Draw graph`, `Clear` の 3 つのボタンがある. さらに数式を表示する部分が左下にある. `Record start` ボタンをクリックし, 1 秒ほど待つと音声認識が開始されるので数式を発声する. この際, 括弧の始まりや終わり (括弧閉じ) や, 数字と変数の間の掛ける (∗) などを明示的に発言する必要がある. その後 `Record start` ボタンの下に認識結果が出力される. 正しくなければもう 1 度入力をおこなう. 次にグラフの描画に移る. 真ん中にある `Draw graph` ボタンをクリックする事でグラフが描画される. 現在, 使うことのできる数学記号は +(たす, プラス),−(ひく, マイナス),∗(かける),/(わる),^2(にじょう),^3(さんじょう),$^{\frac{1}{2}}$(にぶんのいちじょう) で変数は x(エックス),y(ワイ) が使える. 例として $z = x^2 + y^2$ のグラフと $z = (-x^3 - y^3)/(x^2 + y^2)$ の描画結果をそれぞれ図 3, 図 4 に示す. グラフは $[-5, 5] \times [-5, 5]$ の範囲に描画されている.

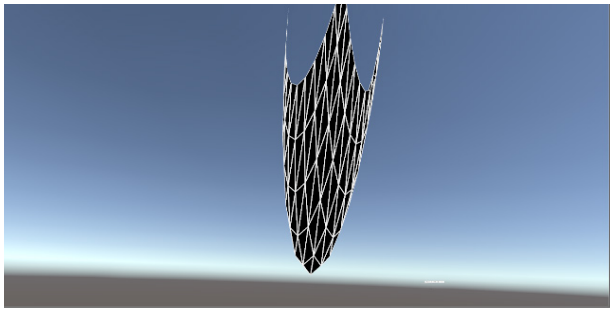

図 3 $z = x^2 + y^2$ のグラフ

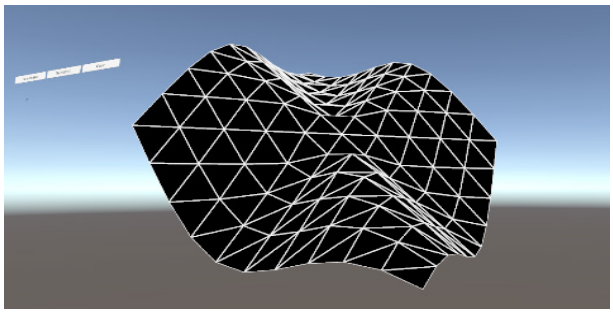

図 4 $z = (-x^3 - y^3)/(x^2 + y^2)$ のグラフ

2.2 数式の入力

本アプリケーションでは数式の入力として, 音声による入力を用いている. これは次の 2 点を考慮したためである. まず, VR 空間でグラフを見るためにヘッドマウントディスプレイ (以下 HMD と呼ぶ) を装着する. そのため, 現実にある機器の操作が難しいと考えたためである. 次に現在は HTC VIVE をプラットフォームとしているが, 最終的には大多数の人が持っているスマートフォンでの利用を考えている. そのため, コンピュータ, スマートフォンの両方で利用できる入力方法を採用した.

音声認識は IBM Watson の Speech to Text[*7] という API を用いておこなっている. しかし, この API は日常的な会話の認識を目的としているため, 非日常的な数式は正しく認識されなかった. 例えば, x^2(エックスニジョウ) と発声すると “x^2” と認識されてほしいが, “x2 畳” と認識されてしまう. そこで, 認識した発声内容を一般的な形で出力するのではなく, こちらが指定した形で出力することができるカスタム言語モデルという機能を利用した. カスタム言語モデルとは Speech to Text の機能の一つで, 認識した音声

[*7] https://www.ibm.com/watson/jp-ja/developercloud/speech-to-text.html

© 2019 Information Processing Society of Japan

表 1　VVPlot の評価結果

項目番号	項目	加重平均	評価値/%			
			4	3	2	1
1	音声入力のしやすさ	2.73	20.0	33.3	46.7	0.0
2	移動のしやすさ	3.13	26.7	60.0	13.3	0.0
3	グラフの大きさ	3.27	26.7	73.3	0.0	0.0
4	グラフの範囲	3.07	20	66.7	13.3	0.0

を数字や漢字に変換する際に, 一般的な形ではない略語などを認識させたいときに使う. 使い方は, まず cURL というツールを用いて Speech to Text のサーバーに自身の ID とパスワードを送信する. cURL とは様々なプロトコルを用いてデータの転送をおこなうことができるコマンドラインツールである. cURL を用いることで, カスタム言語モデルの登録に必要な ID を Speech to Text のサーバーから取得することができる. 次に, ソースコード 1 の形式の json ファイルを作成し, 先ほど取得した ID 共に再びサーバーにファイルを送信する. これらの作業は, アプリケーションの作成時に事前におこない, 先ほど取得したカスタム言語モデルの ID を, アプリケーション内に埋め込むことで, 通常の音声認識の結果に対してカスタム言語モデルの設定内容を反映させることができる.

ソースコード 1 json ファイルの一部

```
"words":[
  {
    "word": "2 乗",
    "sounds_like":["ニジョウ"],
    "display_as": "^2"
  },
  {
    "word": "1/2 乗",
    "sounds_like":["ニブンノイチジョウ"],
    "display_as": "^(1/2)"
  },
  {
    "word": "3 乗",
    "sounds_like":["サンジョウ"],
    "display_as": "^3"
  },
```

ソースコード 1 の"words"の中にある各中括弧が出力を変更する文字のラベルである. "word"はそのラベルの名前, "sound_like"は認識される音声, "display_as"がこちらが指定する出力の形である. この機能を使用することによって例で示した “x2 畳” を “x^2” と表示したり, “+”(たす, プラス) や “/”(わる) といった記号も数式として使える形で出力することができた.

2.3　VR 空間での描画

認識された数式 $f(x,y)$ を用いて, VR 空間での描画をおこなう. まず数式 $f(x,y)$ に (x,y) の値を代入し, 3 次元の点 $(x,y,f(x,y))$ を求める. 点集合 $S=\{x,y,f(x,y) \mid x=-5,-4,\cdots,5 \quad y=-5,-4,\cdots,5\}$ の点から曲面を生成している. $f(x,y)$ の値を求めるために, eval 関数という文字列を数式として評価する関数を利用したい. しかし, Unity の開発言語である C#には eval 関数に相当するものが標準ライブラリに実装されていない. そこでクラスの形で公開されている Calc[4] を利用した. これにより数式から点集合 S を求めることができた. 次に $f(x,y)$ の面に当たる部分を, Unity の Mesh クラスを用いて描画した. Mesh クラスでは任意の 3 点を指定するとそれらを頂点とする三角形のポリゴンを生成することができる. ポリゴンは点集合 S の 1 点に対して正の値方向に隣接する点を用いて生成している. つまり, 点 (x,y) であれば, 点 $(x,y+1)$ と点 $(x+1,y)$ を用いて三角形のポリゴンを作成する. これを $x=-5,y=-5$ から $x=4,y=4$ までおこなっている. これによりグラフのポリゴンが生成される. しかし, このポリゴンは点を結んだ順番が時計回りになる面しか見ることができない. このため, 裏側も見えるようにするために点の順番を変えて, もう一度三角形のポリゴンを生成している. これにより, 目的の立体的な描画を得ることができる.

3.　アンケートによる VVPlot の評価

本システムを 15 名に体験してもらい, このアプリケーションの使用感のアンケートをとった. 評価は 4 が最もよく, 1 が最も悪い評価である. 表にある加重平均とは, 各変量の重みを加味した平均で, 今回の場合はアンケートの各回答結果の評価値を重みとして加重平均を使用した. 表 1 がそのアンケート結果である. 表 1, 項目 1(図 5) は音声入力のしやすさである. 20%のユーザーがとても入力しやすい, 33.3%のユーザーが入力しやすい, 46.7%のユーザーが入力しづらいという結果になった. これはまだ音声による文字の入力が一般的に広まってなく, 慣れていないことが考えられる. また, 普段口にする数式をそのまま発声するのではなく, 括弧の始まりと終わりや数式であれば省略できる乗算の記号も発声しないといけない事も原因の 1 つと考えられる. これについては事前の説明をより丁寧におこなったり, 内部の処理でユーザーが明示的に発言せずとも

© 2019 Information Processing Society of Japan

目的の数式が表示できるようにする方法があると考えられる．また，数式の入力方式も音声入力のみではなく，ソフトウェアキーボードや手書き入力のような画面上で入力できるものも用意すれば，入力のしやすさは向上すると考えられる．

表 1，項目 2(図 6) は移動のしやすさである．26.7%のユーザーがとても操作しやすい，60.0%のユーザーが操作しやすい，13.3%のユーザーが操作しづらいという結果になった．この結果を見ると移動のしやすさについては概ね問題ないと考えられる．一部のユーザーから，首だけ右または左に向けて前進すると，おかしな感覚になるといった評価があった．これは現実で首を右または左に向けて前進するとなると，現実での前進の基準は体の向きであるが，本アプリケーションでは HMD の向き，つまり頭の向きが基準になっているためであると考えられる．これに関しては事前に説明をおこない，ユーザーに頭だけでなく体の向きも変えて移動するように説明する必要がある．表 1，項目 3(図 7) のグラフの大きさや項目 4(図 8) のグラフの範囲については特に否定的な意見がなかったため現在のままで良いと考えられる．

また，アンケートの最後の自由記述の欄に “数学は苦手だが，楽しかった．微積分などに興味を持つきっかけになった．”という記述があった．VVPlot を体験してもらったユーザーのうち数学系の研究室に所属している人は様々な形のグラフを出力して楽しめてもらえていた．また，数学が苦手なユーザでも楽しめてもらい，さらに興味も持ってもらえた．そのため，数学に苦手意識を持っている人に対しても興味を持ってもらえるきっかけとなることができると考えられる．

4. 今後の展望

現在，アンケートの今後追加して欲しい機能にあった三角関数を実装している．本アプリケーションで使用している eval 関数では三角関数が使えないため，三角関数の部分を抜き出して本体とは別に計算している．他にも対数やネイピア数が入った数式のグラフも対応して欲しい，1 つのグラフしか描画できていないので同時に複数のグラフを描画できるようにして欲しいなどの声もあったため，順次対応していきたいと考えている．また，今日はパソコンを使ってのアプリケーションの開発となったが，最終的にはより手軽にこの VR 体験が行えるようにスマートフォンでも使えるようにしていきたいと考えている．

参考文献

[1] The Link Trainer Flight Simulator, https://www.nasa.gov/ames/spinoff/new_continent_of_ideas/, 最終アクセス 2018 年 12 月 17 日.
[2] 田尻 圭佑ら, 3D 仮想モデル操作による没入型天体教材の開発に関する一検討, 日本科学教育学会研究会研究報告, Vol. 32, No. 5, pp. 97-100 (2017).
[3] 藤井 美珠妃, 大西 建輔, 北林 照幸, 藤城 武彦, 高橋 修司, 堀内 翔: 放物線を体感できる VR アプリの試作とその教材としての評価, 第 17 回情報科学技術フォーラム　第 4 分冊, pp. 289-290, 2018.
[4] C# class Calc-Pastebim, https://pastebin.com/wLjD7RjB, 最終アクセス 2018 年 11 月 22 日.

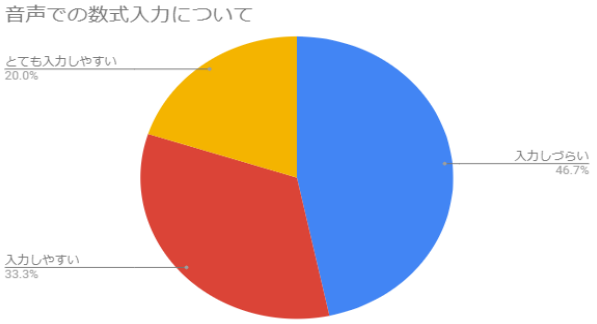

図 5　音声入力のしやすさ

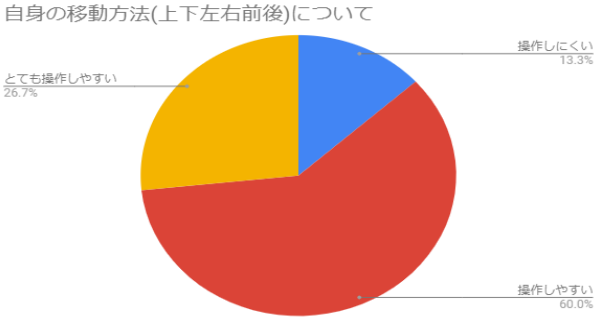

図 6　移動のしやすさ

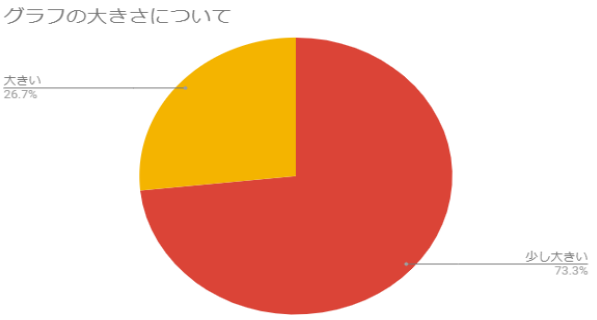

図 7　グラフの大きさ

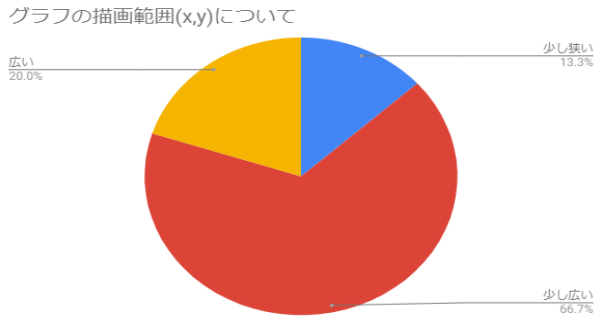

図 8　グラフの x, y の範囲

© 2019 Information Processing Society of Japan