ジェスチャー認識装置を用いた人体位置検出と工程作業動作分析への応用

Applying Gesture Recognition Technology to Industrial Engineering and the Development of Web-Based System

熊　谷　卓　也*
Takuya KUMAGAI

要旨

NUI（Natural User Interface）と呼ばれる，人の動きやジェスチャーを認識する技術の発展はめまぐるしい。安価に入手できるジェスチャー認識デバイスが増え，エンターテインメント以外でも，様々な分野で応用法が研究されている。本研究ではIE（Industrial Engineering）への応用に着目し，組立工場の改善活動を加速させることを目的としている。

IEによる分析を効率的に行えるように，NUI技術をはじめ，最新のWeb技術やビッグデータ技術を融合した，クロスプラットフォームかつスケーラブルで使いやすいシステムを開発した。これによりIEによる分析を効率的に行うことが期待される。

本稿では，システムの中核であるNUI技術のIEへの応用と，それを取り巻く技術を紹介する。

Abstract

Natural user interfaces (NUIs) exemplified in gesture recognition devices like Microsoft's Kinect sensor has gained public attention. Because gesture recognition devices provide an easy-to-use interface with cutting-edge technology, NUIs are now found in the amusement, entertainment, and video game industries.

NUIs, as seen in popular gesture recognition devices, are indicative of the future of human machine interaction (HMI). NUIs interpret the natural movements of a person such as gestures, which allows people to operate computers more interactively. NUIs' applicability in a host of fields has the potential to create a paradigm shift in HMI.

This study focuses on the application of gesture recognition technology to industrial engineering (IE) and especially to Kaizen, the practice or the philosophy of methodically improving manufacturing processes. The application of gesture recognition technology can enhance the vital value and power of IE and, ultimately, expand an enterprise's value by optimizing manufacturing processes. This study provides an avenue to the achievement of a system that incorporates new technologies within the advancement of manufacturing methodologies.

A system that provides a simple and clean interface to cutting-edge technologies is the key to obtaining competitive advantage. To gain competitiveness, rather than develop standalone systems, it is better to develop such systems as Web-based cross-platform applications, since such applications would allow trends in tablet computers and smartphones to be incorporated in the system. This paper covers the application of gesture recognition technology and the development of a system comprised of Web, big data, and machine learning technologies, which machine learning technology automatically analyzes the tremendous amount of data gathered from gesture recognizing devices.

*生産統括部　生産改革部

1　はじめに

昨今，NUI技術が発達し，様々な環境で直感的に操作することができるシステムが普及してきた。中でも従来のGUIに取って代わる技術としてジェスチャー認識がある。本研究ではジェスチャー認識装置を用いて人体の様々な動きを取り込み, IE（Industrial Engineering）手法を用いて動作分析作業を行うWork Analysisシステムを開発した。本稿ではWork Analysisシステムで採用した技術及び搭載機能を紹介する。

2　技術紹介

2. 1　NUI

人の動き（ジェスチャー）認識装置の普及でHMI（Human Machine Interaction）に革新的な技術が増え，人体をコントローラーとして操作できるアプリケーションがゲーム業界を筆頭に応用されてきた。NUI（Natural User Interface）はこうした技術の総称で，直感的な操作を実現することを目的としている。ゲーム業界だけではなく，医療現場では手術中の医師が手を汚さずにその場でレントゲン写真を確認するシステムや，開発現場での応用も進んできている。映画のように直感的な動作（ジェスチャー等）でシステムを操作することを目的とした研究も多くある[1)]。

2. 1. 1　Kinect

人体の動きを取得するためには，関節などにマーカー（標識）を装着し, 3次元座標をカメラで測定する方法が主流だったが, Microsoft Kinectセンサーは非接触かつ非侵襲に関節位置の3次元座標を得ることができる。KinectにはRGBカメラ，赤外線プロジェクター，赤外線カメラ，アレイマイクという4つのセンサーが搭載されている。

人体の位置検出は内蔵API（Application Programming Interface）が行う。赤外線プロジェクターはランダムパターンを照射し，赤外線カメラがパターンの歪みを読み込む。歪から距離（深度）を認識し，APIにより人体の関節位置20か所の座標を認識することができる。

Fig. 1　Microsoft's Kinect has four sensors: a camera, an infrared projector which emits random patterns to recognize the depth of an object, an infrared camera which captures the patterns, and an array microphone to capture voices.

また，通常のカメラと同様に画像を取り込むことができ，深度データをピクセル単位でマッピングできる。Kinectの利点はこれらの複雑な処理を全てAPIが包括している点にある。APIを呼び出すだけで深度画像や人体のスケルトン情報を取得できるという簡便なしくみによって，開発者はアプリケーションに集中して取り組むことができる。

開発時点のSDK（System Development Kit）バージョンではC++やC#, Visual Basicを通してAPIを呼び出すことができ, IDE（Integrated Development Environment）にVisual Studioを用いる。本稿執筆中にSDKが1.8にバージョンアップし，後述するHTML5/JavaScriptでの開発環境をサポートしている。

2. 2　Web技術

近年スマートフォンやタブレット等の普及や，モノのインターネット（Internet of Things）という概念の一般化で，電化製品をはじめとする様々な製品がインターネットに繋がるようになり，Web技術は急激に発展してきた。これまで，単一のプラットフォーム上で動作することが暗黙的な共通認識だった従来のアプリケーションも，OS（Operating System）等に捉われないWeb技術の応用によって，クロスプラットフォームアプリケーションとして動作することが求められてくる。本節ではシステム開発に適用した技術等を紹介する。

2. 2. 1　HTML5とJavaScript

HTML5（Hypertext Markup Language 5）はWebの基幹的役割を持つ技術の5世代目となるバージョンである。HTML5は，リッチなコンテンツを提供するプロプライエタリ（Proprietary）なプラグイン（Adobe Flash, JavaFX, Microsoft Silverlight, ActiveX等）を置き換えるマルチメディア要素等を取り込んでいるため，クロスプラットフォームアプリケーションを開発する際には妥当な選択となる[2)]。

JavaScriptはWebページの為に開発されたクライアント・サイドのプログラミング言語であり，jQueryやPrototype等の多種多様なライブラリも魅力のうちの一つである。現在ではモノのインターネットの牽引役としてHTML5と共に広く普及している。

HTML5とJavaScriptを組み合わせることで，高機能かつ動作環境に捕らわれないクロスプラットフォームアプリケーションを開発することができる。

2. 2. 2　Node.js

Node.jsはV8 JavaScript Engineと呼ばれるGoogleが開発したJavaScriptを高速に実行できる環境上に構築されたプラットフォームで, 高速でスケーラブルなWebアプリケーションを簡単に構築することができる。

イベント駆動とノンブロッキングI/Oというモデルに基づいており，接続するノードが増えてもサーバーへの負荷が増えにくいという，効率的に分散されたデバイスに向けたリアルタイムアプリケーションを構築できる[3)]。

Node.jsによりバックエンドの開発言語にクライアント・サイドと同じJavaScriptを利用できる為，効率的に

開発をすることができる。また，アプリケーションプラットフォームのExpress.jsを用いることで，HTMLテンプレートエンジンやCSSフレームワークを手軽に利用でき，またSocket.ioを読み込むことにより，WebSocket機能を用いたリアルタイムWebを容易に構築出来るなど，有用なプラグインが多いことも特徴である。

2. 3　ビッグデータ

ビッグデータという言葉は昨今様々な業界で注目されている。一般的に量（Volume），速度（Velocity），多様性（Variety）という3つ性質を持ち，多種多様な大量のデータをインプットし，高速・リアルタイムな処理を行い，得られた結果を企業活動の改善や公共・公益の増進に繋げるパラダイムのことである。

ビッグデータの中核をなす技術にHadoopに代表されるNoSQL（Not Only SQL）と呼ばれるデータベースの枠組みがある。様々なソースから発生する大量のデータは正規化が難しく，リレーショナルデータベースに格納するには非効率だが，NoSQLの導入でそれらを高速かつ効率的に処理することができる。

2. 3. 1　MongoDB

MongoDBはスキーマレスで，データをキーと値のセットとで扱うKVS（Key-Value Store）型データベースの長所と，リレーショナルデータベースのテーブル概念を兼ね備えた，ドキュメントという単位で扱うドキュメント指向データベースである。豊かな表現力を持ち，階層型のデータ構造を表現できるので，大規模でスケーラブルなWebアプリケーションの為の汎用的なソリューションである[4]。

MongoDBがリレーショナルデータベースの多様なクエリの強力さをほぼそのまま保ち，クエリや関数は全てJavaScriptで記述することができる。これによって，クライアント開発や，サーバー開発と同様に，データベースを同じプログラミング言語で扱うことができる。

Kinectから取得できるデータ構造をFig. 2のように定義し，MongoDBに格納することで，画像データと全関節位置の3次元座標データを単一のIDで指定することができるようになる。

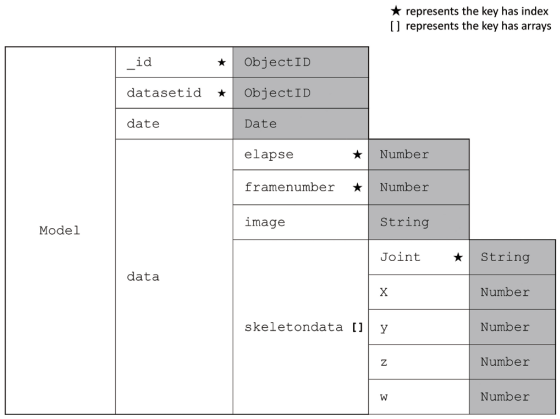

Fig. 2　Kinect data structure. Each dataset has a key and a value so as to compose a whole document.

3　システム開発

本章では前述した技術をどのように応用してシステムを構築したかについて述べる。具体的には，Kinectから取得したデータを格納し，可視化するプロセスの実装方法とUI（User Interface）設計を紹介する。

3. 1　人体位置検出と座標データの取得

人体の位置検出と座標データの取得はKinectを用いて行う。開発時点のSDKではJavaScriptからAPIを呼び出すことができなかったため，Kinect周りは全てC++で実装することとした。Kinectは内蔵されているAPIを通すことで，深度画像から「人物」らしき物体を識別し，手や頭，足など全身20箇所の3次元座標を最大30fps（frame per second）で取得することができる。距離データは16bit値として得られるが，APIを通すことで，Fig. 3のように3次元座標に変換することができるが，ミリメートル単位の細かな動作は認識できないことが多い。

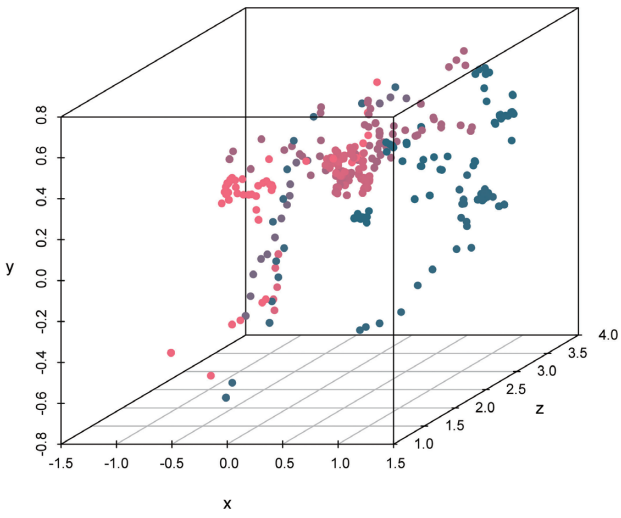

Fig. 3　Scatter plot of left hand position with respect to time, where hue represents density.

3. 2　撮影データの蓄積と転送方法

Kinectから取得できるデータをデータベースへ格納するにはFig. 2で定義したデータモデルに変換する。画像データはバイナリーデータをBase64でエンコードし，その他の情報と合わせて格納するが，データ容量の観点からは，撮影画像を1枚ごとに格納するのではなく，動画形式にエンコードしたものを格納する方が効率的である。しかし，本システムでは正確性に重点を起き，座標データと撮影画像を一塊で格納することとした。このためFig. 4のようにサーバーへTCP/IPソケット通信で送信することで，データベースへの格納と同時に撮影中のデータをクライアント端末からリアルタイムに確認できる。

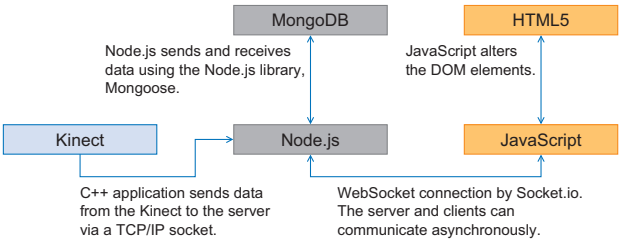

Fig. 4 System structure and data flow. The server broadcasts Kinect's data to the clients via WebSocket API to provide a real time Web feature.

3. 3　データの取得と可視化

データを蓄積したままでは，十分に活用することはできない。この節ではクライアント端末から蓄積したデータへアクセスする方法と，可視化について紹介する。

3. 3. 1　蓄積したデータの抽出

MongoDBに蓄積されたデータはFig. 2 の構造を持っているため，一連の撮影データを特定するために，キーとなる値を指定する必要がある。各ドキュメントには区分を示すdatasetidを持たせているため，クライアント端末からこの値を指定できるようにUIを設計した。IDが指定されると，サーバーがMongoDBから該当のデータセットを特定し，データをクライアント端末に送信することができる。クライアント端末は受信データを内部ストレージに一時的にストックすることで，後述する可視化や分析を行う。

3. 3. 2　座標データの可視化

可視化方法は様々だが，本システムでは取得した座標データをFig. 5 のようにグラフ化した。特に各関節の移動量に着目し，時間軸の変化により，移動量推移を折れ線グラフ，円グラフ，ヒートマップで表示し，任意の関節の動線を撮影画像上にマッピングした。これによりどの部位の移動量が多いか，ムダな動きをしていないかを視覚的に確認することができるため次章で紹介する分析作業を行いやすくなる。

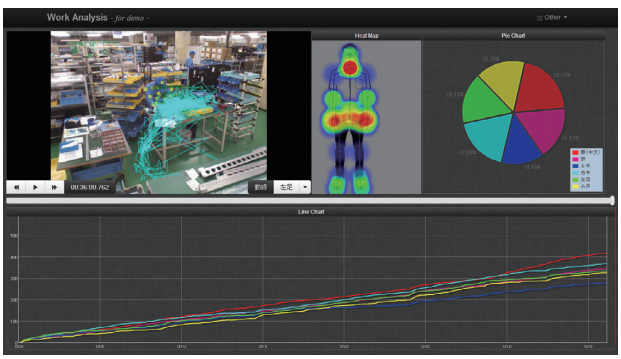

Fig. 5 User interface of the system. Images from the Kinect are displayed in the top-left area along with flow lines, and position data converted into movement are visualized as a heat map, a pie chart, and a line chart.

3. 4　UI（User Interface）設計

Node.jsのWebアプリケーションフレームワークとしてExpress.jsを用い，HTMLテンプレートにEJS，CSSフレームワークにはTwitter Bootstrapを適用した。またクライアントとサーバー間ではWebSocketの導入でハンドシェイク手続きにより双方向通信を実現するために，Socket.ioをバックエンドに設置する。

Twitter BootstrapはCSSフレームワークに加えてjQueryプラグインのコンポーネントを備えており，Fig. 5 のようにHTML5をベースとしたWebページを短時間で制作できる。システム全体のレイアウトは動画編集ソフトで多く用いられるようなレイアウトを踏襲し，比較的操作しやすい設計とした。

4　動作分析

人体の3次元座標から，“動き”を可視化し，収集データから直接的に導き出せる情報を得ることができた。しかし，本システムの目的は冒頭で述べたとおり，IE手法を用い動作分析を行うことにある。ここまで収集した情報は量という側面から分析することはできるが，IE手法では時間という側面から分析することが重要である。従来であれば，人手を介さなければ難しく，撮影データを細かく確認をしながら，対象作業者の作業時間を計測し集計するため，分析に膨大な工数を要する。本章ではシステムに実装した動作分析をサポートするしくみを紹介する。また今後の発展として取り込んだデータから動作素を自動的に解析する手法としてニューラルネットワークを紹介する。

4. 1　IEによる分析をサポートするしくみ

本システムでは効率良く分類を行う為に，Fig. 6 のように同一画面上にスペースを取り，簡単なマウス操作だけで分類を行えるようなUIを開発した。

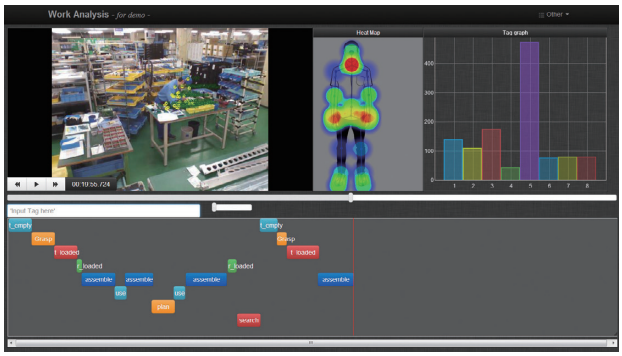

Fig. 6 User interface for data analysis. The work space below enables the user to set therblig units.

作業を分析するために，ビデオカメラで撮影する方法をIEではVTR法と言うが，本システムも広義ではVTR法だと言える。一般的に，VTR法では撮影した映像を再

生と停止を繰り返しながら，作業者の動きを分類し，それぞれの所要時間を集計する。改善活動は分析データを元に行うため，この作業をできるだけ正確に行うことが望ましいが，工数とのトレードオフの関係にある。しかしながら前述のFig. 6に示すようなUIにより，単純な作業工程の分類の他，サーブリッグ分析（微動作分析）のような詳細なユニットであっても簡単に設定できるようになる。

4. 2　自動分析技術（動作素解析）

Kinectを用いた本システムでは作業者の各関節位置座標データという従来の方法では測定が難しかった情報を得ることができるため，この情報を元に同様の解析―すなわち，座標データから被写体の作業をクラスタリングし，各作業の所要時間を求める―を自動的にある程度の精度で行える可能性がある。ここで，分類したい作業の動作を動作素（クラスター）と呼び，複数の入力値から動作素をクラスタリングする方法について紹介する。具体的には，大量のデータの中から有意データを抽出又は算出し，それらを入力値としてクラスタリングを行うニューラルネットワークを構築する。

4. 2. 1　膨大なデータ量に対応する技術（MapReduce）

解析対象に全撮影データを対象としてしまうと，計算量が撮影時間に比例して増大してしまうため，データをある程度集約する前処理を構築する必要がある。今回はビッグデータ分析の手法の一つであるMapReduceモデルを用い，MongoDB上にFig. 7のような処理を作成した。

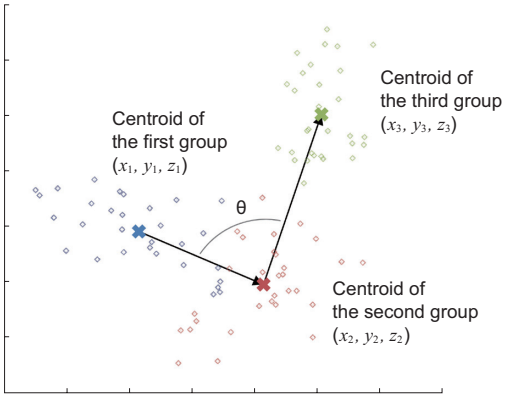

Fig. 7 Results obtained from the MapReduce function, which calculates centroids or geometric centers for each group and an angle composed of two vectors.

この処理は1秒間の各関節位置の3次元座標の重心と標準偏差を算出し，連続する2点の重心のベクトルを求め，更に2つのベクトルのなす角を算出する。得られる結果は，3秒間の座標データとなる90フレーム分の集約として3点の重心座標とそれぞれの標準偏差，3点からなるベクトルがなす角の7項目である。この処理により，データ量はFig. 2のデータ構造をそのまま処理するのに比べ，大幅に圧縮できることが期待される。

4. 2. 2　ニューラルネットワークによる機械学習

MapReduce処理によって得られた結果が，どのクラスターに当たるかを判断するために，ニューラルネットワークを用いて，推測することとする。ニューラルネットワークは生体システムにおける情報処理，すなわち脳の機能を数学的に表現する試みのことだが，統計的パターン認識の効率的なモデルの1つとして知られている。

ネットワークはFig. 8のようにインプットノードに対して任意数のアウトプットノードが存在し，アウトプットを算出するための中間ノードがある。インプットに対してある係数で重み付けをすると，アウトプットが算出されるしくみである[5]。

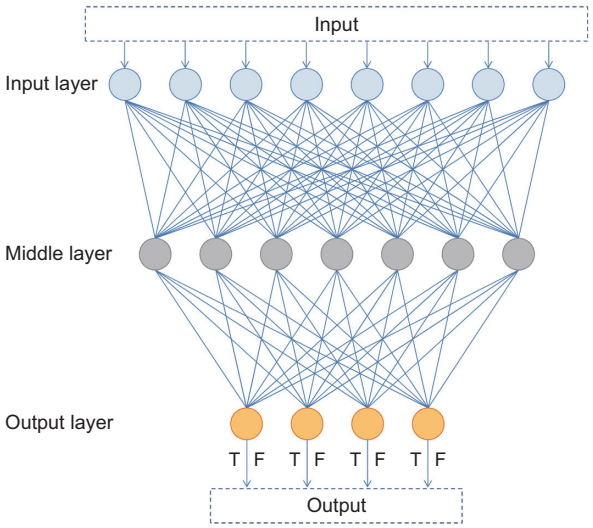

Fig. 8 The neural network has input nodes and output nodes. The nodes in the middle layer calculate weights for input nodes to express outputs.

重み係数を修正する学習アルゴリズムはバック・プロパゲーション（誤差逆伝播）を用いることで，中間層に隠れノードが存在していても，一定の精度で結果を得ることが出来る[6]。

ニューラルネットワークは，統計解析環境のR言語(R)を用い，バッチ的に実行する構成とした。現時点では本システムとRがシームレスに接続されていないため，オンラインでの実行ができないが，このネットワークをシステムに取り込むことで，分析作業を大幅に削減できる。

5　課題

実際の組立工程で作業を2日間で合計9時間撮影し，システムの有効性を検証した。Kinectを用いることで，従来では計測することが難しかった人体の3次元座標を得られ，また各部位の移動量を比較できる為，分析箇所の絞り込みに有効であることが分かった。また，分析ツールとしても，ブラウザ上で動作するため，インターネット接続可能なデバイスであれば，どこからでも簡易に分析作業を行えることが確認できた。

しかしながら，後ろ向きでの作業や，しゃがみ込んでの作業を行っていると，座標データを正確に認識できず，Kinectの内蔵APIにより，推測位置を取得してしまう。場合によっては，作業机等を人体と誤認識してしまうこともあり，座標データからの分析が全くできない場合もあることを確認した。これらの多くは撮影シーンに依存してしまう。人体認識は本研究で採用したKinectのAPIに依存しているため，Kinectの精度向上や複数台を用いて死角をなくすなどの対策が必要になる。

また，ニューラルネットワークが本システムと構造的に切り離されているため，効率的に機能していないことが挙げられる。現時点ではRを通してバッチ的に処理している部分を，将来的には本システムに取り込んでいく必要がある。分析精度という面では訓練データが少なく十分な精度でクラスタリングできないため，訓練データの作成とネットワークの訓練が必要である。

6　まとめ

現状ではKinectを工程作業分析の根幹に据えるには，認識技術を向上させる必要があることが確認できた。しかし，本システムはどんなデバイス上であっても動作するクロスプラットフォームアプリケーションとして構築し，これを取り巻く技術は多くの優位な特徴を有すると考えている。

NUIをはじめとして，Web技術やビッグデータ技術の発展は目覚ましいため，生産現場における業務支援に貢献できる新たなソリューションを提供し続けることを目的に，継続して技術動向に着目していき，更に使いやすいシステムの構築に役立てていきたい。

●参考文献

1) Oblong, G-SPEAK, October 02, 2013. <http://www.oblong.com/g-speak/>
2) W3C. HTML5. August 06, 2013. October 02, 2013. <http://www.w3.org/TR/html5/>
3) Joyent. node.js. October 02, 2013. <http://nodejs.org>
4) BankerKyle. MongoDB イン・アクション. 訳 玉川竜司. 東京: 株式会社オライリー・ジャパン
5) BlaisAndrew, MertzDavid. “An Introduction to neural networks.” July 01, 2001. IBM Developer Works. October 02. 2013. <http://www.ibm.com/developerworks/opensource/library/l-neural/>
6) Bishop M. Christopher. パターン認識と機械学習. 訳 元田浩 ほか. 東京: 丸善出版株式会社, 2012.