

# Kinect v2 センサを用いた 3D Viewer の非接触閲覧システムの開発

正会員 永井 孝志郎\* 正会員 木村 元\*\*

A study of a non-contact 3D viewer system making use of Kinect v2 sensor  
by Koshiro Nagai, Member Hajime Kimura, Member

**Key Words:** Non-contact Input Device, 3D Viewer, IoT

## 1. 緒 言

造船所内の作業現場では、作業効率向上のため、これまで紙媒体を介して行われていた工程確認作業や図の閲覧などを端末を介した3D電子データのやりとり置き換える取組みが行われつつある。これにより設計-現場間の意思疎通が容易になり、迅速な作業指示や作業進捗のリアルタイム把握が可能となる。しかし従来の技術でデバイスを操作するには、現場作業者は操作に邪魔な装備を外さなければならないなどの問題があった。本研究では Kinect v2 を利用してジェスチャによる非接触の操作によって 3D モデルを閲覧し、必要なデータの出し入れが可能なシステムの構築を提案する。本研究の先行技術として Kinect を用いた医療向けの非接触画像閲覧システムがある[1][2]。本研究ではこれらのシステムとは差別化して造船所向けの開発を行い、作業現場での実用可能性について評価を行った。

## 2. 先行関連研究について

先行技術として、手術室向け非接触型画像操作システム Opect (株式会社ニチイ)<sup>1)</sup>や医療系非接触閲覧システム iKINESYS (株式会社ネクストシステム)<sup>2)</sup>などがある。これらは両方とも医療向けに開発されたシステムで以下のような特徴を有する：

- 完全非接触操作による衛生面の向上
- 手術時間短縮による患者の負担の軽減
- レントゲン写真や CT スキャンの画像を利用可能

清潔を保たなければならない手術室において非接触で操作を行うことができるのは衛生面に関して優れており、また執刀医が自らの意思で操作ができるため、補助の医者に操作を要求する手間が省ける点も魅力である。また、手術用手袋は手のラインが明瞭に浮かび上がる形となっているので Kinect により「手」として認識しやすい点で有利である。本研究では、遠隔かつ非接触で 3D-Viewer を操作するシステムを造船所に適した形で開発する。

## 3. 開発した現場向け閲覧システムの概要

### 3.1 ジェスチャ入力システム

\* サノヤス造船株式会社

\*\*九州大学 大学院工学研究院

原稿受付（学会にて記入します）

秋季講演会において講演（学会にて記入します）

©日本船舶海洋工学会

本システムはプログラミング言語として C# を用いて開発した。ライブラリとして KinectSDK ver2.0, WPF (Windows Presentation Foundation) を用いた。また、本システムでのクリック (タップ) 判定には MSDN (Microsoft Developer Network) の Inject Touch Input 関数を用いた。本システムはスマートフォンやタブレット PC のタッチパネルの代替として、ジェスチャ操作を用いる。PC のモニタ上部に Kinect v2 を配置し、モニタの前に手をかざすことでマウスポインタに見立てた灰色の四角のアイコンが移動する。Fig.1 は実際にモニタの前でジェスチャにより PC を操作している様子を示す。Fig.2 はディスプレイ上に表示されるポインタを表すアイコンの形状で、Fig.2 の左側の四角形は何も選択していない hover 状態と呼ばれる状態で、右側の丸は対象を選択あるいはクリック状態を示す interactive 状態を表している。

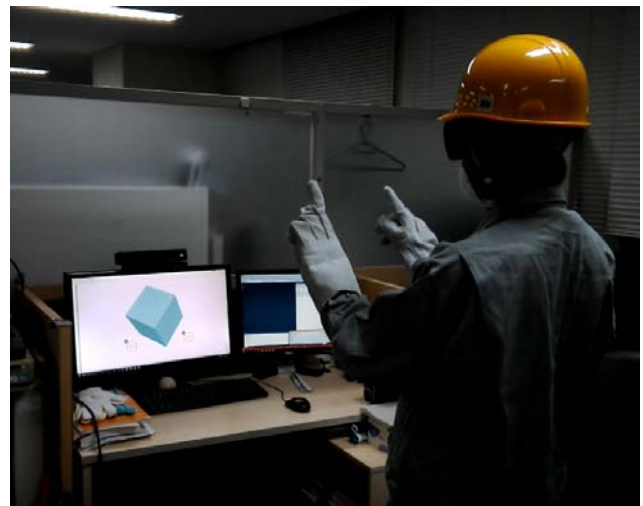


Fig.1 The gesture input system developed in our research.



Fig.2 Displayed icons as the pointers on the monitor.  
Left: Hover state, Right: Interactive state.

Table 1 Comparison of the standard mouse operation and the gesture operation of our system.

マウス操作	ジェスチャ操作
左クリック	手をパー (Open) の状態 (Hover) からチョキまたは人差し指を掲げる (Lasso) の状態 (Interactive) (Fig. 1 参照) にする
右クリック	定点保持で左クリックと同じ状態を数秒間続ける
ドラッグ	左クリックの状態を維持したまま手の座標を移動させる

Table 1 は通常のマウス操作と本システムのジェスチャ操作との比較を示す。ほぼマウスと同等の操作が可能だが、実際は両手を使用してタッチパネルをピンチイン・ピンチアウト・両手でドラッグなどの操作も可能なので、マウスよりも多様な操作となる。

手の開閉状態{グー (Closed) チョキ (Lasso) パー (Open)} は KinectSDK で提供されている API をそのまま利用して認識を行う。Hover 状態から Interactive 状態へとシフトする動作を当初は Open から Closed としていたが、手袋を着用した状態では、グー (Closed) の状態を認識するのが困難なケースが多かったため、最終的に Open から Lasso の状態へのシフトを Hover 状態から Interactive 状態へと設定することで認識率が向上し、実用的な操作が可能となった。

### 3.2 3D-Viewer

本システム用いての 3D モデル表示・操作を行う Viewer としては、一般的なブラウザを使用し、3D 表示のためにコンテンツやデータについて HTML と Javascript を利用して加工・作成した。3D モデルのファイルフォーマットとしては X3D を使用しており、3D モデルをブラウザに表示するためのオープンソースランタイムとして X3DOM を用いた。X3DOM とは XML 形式の 3D データフォーマットである X3D を WebGL でレンダリングするためのフレームワークである。HTML をブラウザで表示することを基本とするため、3D データに関連する文字情報や画像データを簡単に引き出せる利点がある。

本研究で開発した Viewer には以下の機能がある：

- PC 上のファイルシステムの X3D ファイルを開く
- Viewer 上の部材をクリック (タップ) することで ComponentNo.を表示する
- Component 毎に進捗状況を管理できる

Fig.3 に本研究の Viewer の画面例を挙げる。Fig.3 の左上にある [Open] ボタンをクリックすることで、ファイル選択ダイアログが開き、X3D フォーマットのファイルを選択することで、Viewer 上に X3D モデルが 3D 表示される。また Fig.3 の [Open] ボタン下部にある clicked component 欄に Viewer 上でクリック (タップ) した部材の Component No.が表示される。Fig.3 右側の Component 欄は開かれている X3D ファイルの <shape> タグの数によって決められ、○×の画像をクリック (タップ) することで画像中の対応する部材の色が切り替わり、作業の進捗状況を管理できるようになっている。

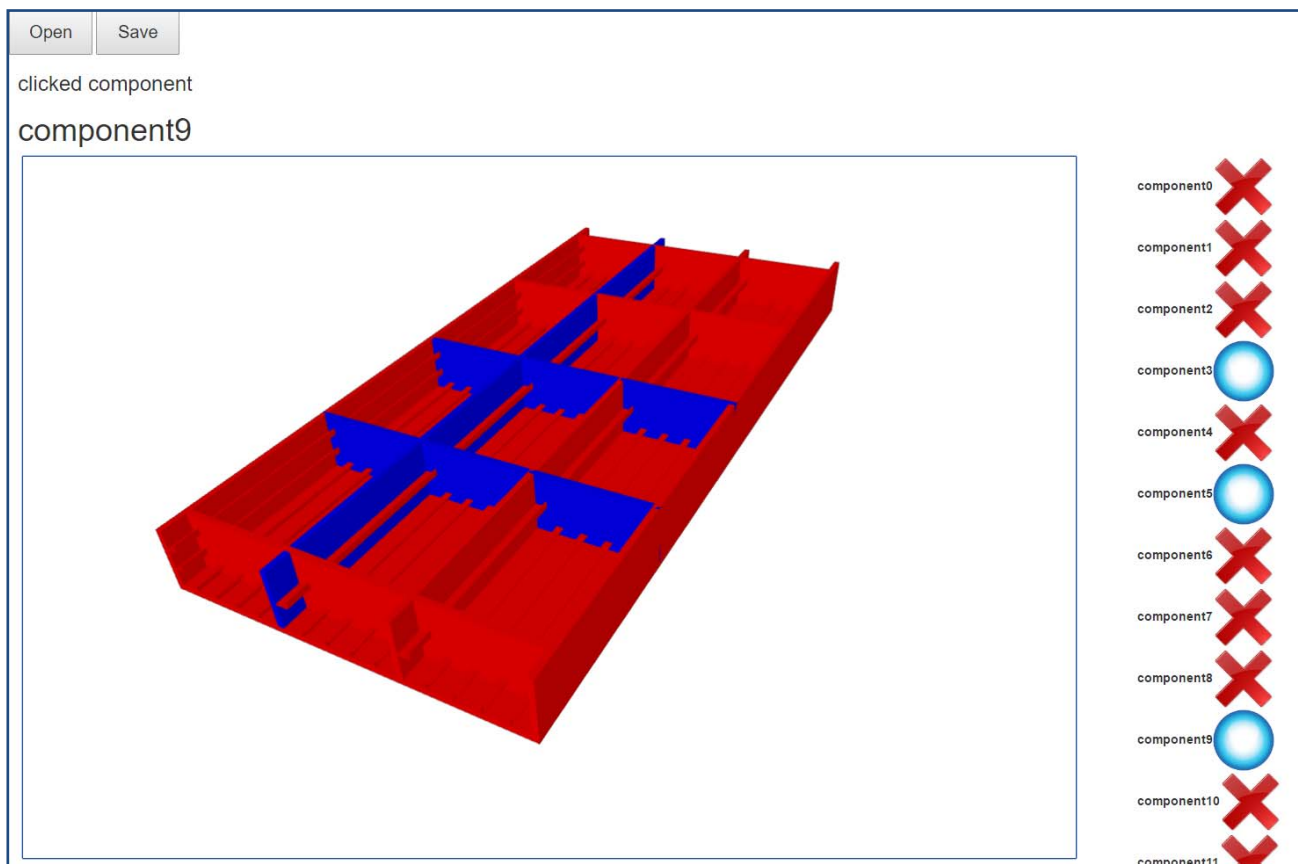


Fig.3 An example of a screenshot of our 3D-Viewer.

本研究の Viewer 上で 3D モデルを動かす際、その基本的な操作方法は一般的なタッチパネル操作に準ずるため、訓練無しで誰でも簡単に操作可能である。Table 2 に本システムによるモデルの操作方法をまとめる。

Table 2 Methods of 3D operations in the 3D-Viewer.

操作	操作方法
3D モデルの回転	片手でのドラッグ
3D モデルの拡大	両手でのピンチアウト
3D モデルの縮小	両手でのピンチイン
3D モデルの平行移動	両手でのドラッグ

#### 4. 実験による操作性の評価

##### 4.1 ジェスチャ入力時の服装と装備について

本研究では造船工場内の現場での PC 操作を行うことを目標としているため、Fig.4 に示す作業着にヘルメット、ゴーグル、防塵マスクに作業用の手袋を着用した状態で実験を行った。作業用の手袋としては、Fig.5 に示すように左から順に溶接用手袋（革製）、軍手、およびゴム手袋の 3 つを試した。

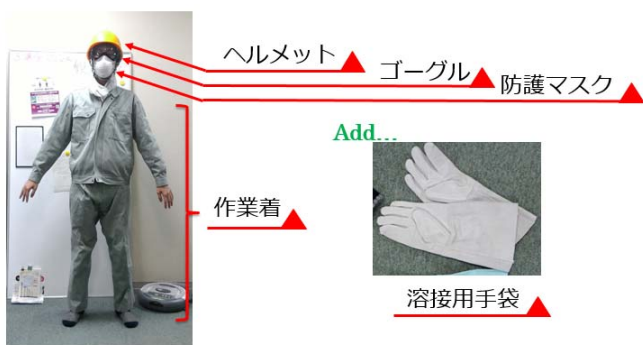


Fig.4 Clothes at the experiments.



Fig.5 Three kinds of gloves used at the time of experiments.

##### 4.2 操作性の評価

本研究で開発したジェスチャ入力システムの操作性を評価するため、以下の実験を行った：

- ① クリック実験
- ② 文字入力実験
- ③ ドラッグ&ドロップ実験

これらの実験にはブラウザのフリーゲームである「マウス検定試験」<sup>4)</sup>を利用した。

##### (1) クリック実験

上述したマウス検定試験の科目一：クリック速度、二：クリック正確さを検証するため、ブラウザ上に 9 箇所の中からランダムに出現するボタンを 10 秒間に何回クリック（タップ）することができるかを評価する実験を行った。本テスターは HTML と Javascript で作成した。Fig.6 に本実験において表示される画面の例を示す。



Fig.6 A screenshot of the click performance evaluation.

Fig.6 中に示された[P]ボタンをクリック(タップ)すると、そのボタンは消失し新たな[P]と表示されたボタンが出現する。右下にはカウンタが設けられ、クリック（タップ）した回数を記録する。これを 10 秒間に何回クリック（タップ）できるかの計測を、素手、溶接用手袋、軍手、ゴム手袋ごとに 10 回ずつ、そして比較用にマウスで 1 回行った。また、この実験とは別に短期間内での習熟度による精度向上を確認するため素手で 100 回実験を行い、10 回ずつ平均値や分散を算出した。Fig.7 に 3 セット分の計測結果のグラフを示す。手の形をパーからチョキに変えることでクリックするため、マウス操作のクリックほどの性能は出ない。また溶接用手袋は軍手などに比べると手の形状認識が難しく、クリック回数が 2 割程度劣る。

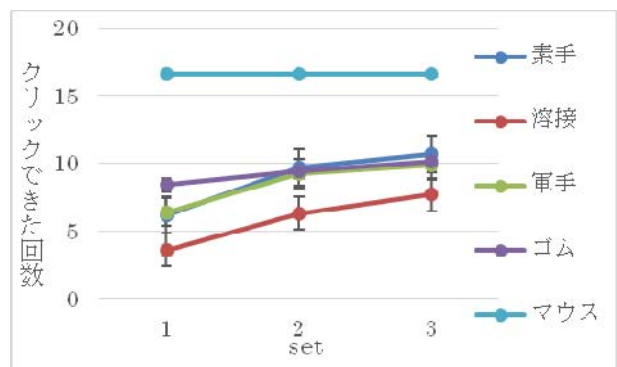


Fig.7 Results of the experiment of the click performance.

## (2) 文字入力実験

マウス検定試験の科目三：スライド速度，四：スライド正確さを検証するため，フリーソフト ForbiddenFruit のスクリーンキーボード FlickKeyboard<sup>4)</sup> を用いて文字入力実験を行った．FlickKeyboard での入力方式は，スマートフォンやタブレット PC 等でよく用いられるフリック形式が採用されている．図のように「ら」の付近のマスをタップすると「ら」を中心に上下左右「る」「ろ」「り」「れ」の文字が現れ，それぞれの方向にフリックすることでその文字が入力される．またこの FlickKeyboard には BackSpace キーや Space キー，TAB キーも標準装備されており，入力を間違えた時にも Delete することが出来るし，漢字変換なども可能である．文字入力実験では，素手，溶接用手袋，軍手，ゴム手袋ごとに 3 回ずつ，また比較用としてマウス入力とキーボード入力を 1 回ずつ行った．なお，この実験で入力した文字列は「国境の長いトンネルを抜けると雪国であった．夜の底が白くなった．信号所に汽車が止まった．向側の座席から娘が立って来て，島村の前のガラス窓を落した。」の 72 文字である．この入力にかかった時間（単位：Second）を計測した．Fig.8 に本実験で文字入力を行う様子の画面例を示す．

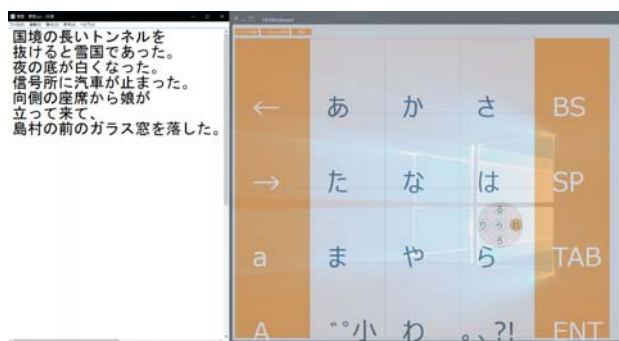


Fig.8 A screenshot of the text input performance evaluation.

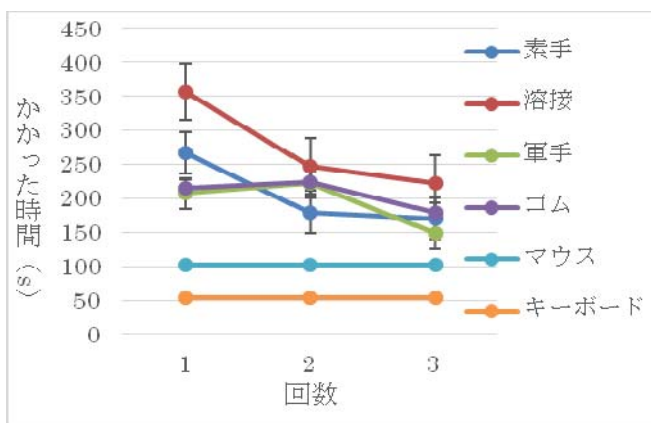


Fig.9 Results of the text input performance evaluation.

Fig.9 に文字入力実験の結果を示す．72 文字の入力はキーボードが約 50[sec]と最も短時間であり，マウス入力とその次に短く約 100[sec]を要している．ジェスチャによる文字入力はさらにその倍の 200[sec]以上を要しており，習熟すればもう少し時間の短縮は可能と考えられるが，あまり長い文字列の入力には適さない．

## (3) ドラッグ&ドロップ実験

マウス検定試験の科目五：ドラッグ&ドロップ速度/正確さを検証するため，デスクトップ上に敷き詰めたテキストファイルのアイコンを 30 秒間に何個ゴミ箱に移すことができるかという実験を行った．この実験も実験 (1) のクリック実験と同様に素手，溶接用手袋，軍手，ゴム手袋と，比較用にマウス操作で実験を行った．Fig.10 に本実験で用いたゴミ箱アイコンとテキストファイルのアイコンのデスクトップ上の配置を示す．



Fig.10 A screenshot of the drag & drop performance evaluation.

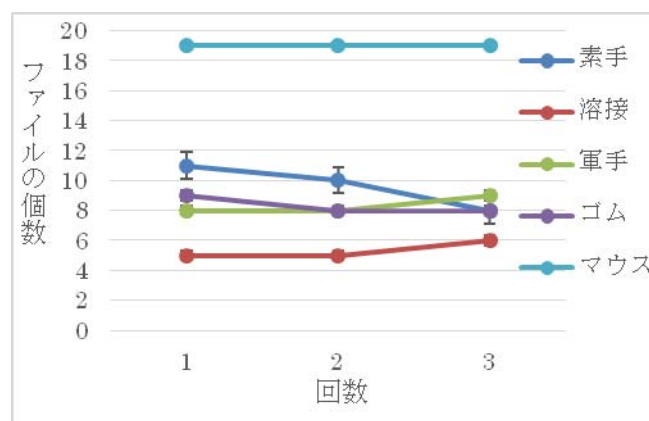


Fig.11 Results of the drag & drop performance evaluation.

Fig.11 にドラッグ&ドロップ実験の結果を示す．ジェスチャによる操作はマウス操作で行ったドラッグ&ドロップファイル数の半分程度であった．これはジェスチャによるボタンクリック動作が遅いことやジェスチャによる正確な移動がやや難しいためと考えられる．

## 5. 考察

まず，手袋ごとに着目した場合，軍手とゴム手袋は素手に劣らぬ結果を出し，溶接用手袋が少し劣る結果となった．これは溶接用手袋が軍手やゴム手袋と比較して手の形が明確に浮かび上がるようなデザインではないためと考えられる．手のラインが曖昧になると Kinect 側も手の開閉状態を認識しづらいためと考えられる．

次に，回数ごとに着目すると，ドラッグ&ドロップ実験を除く実験結果において回数を重ねるごとにパフォーマンスが上昇する傾向が見られた．これは実験を行っていくうちに習熟によって良結果を出すようになったためと思われる．この習熟度を検証した実験では 1set 目の平均値が 8.5 回に始まり，そこから右肩上がりに上昇し，7set

目頃から 10.5 回前後で収束した。

ドラッグ&ドロップ実験において上記の手袋差，習熟によるパフォーマンス向上の傾向があまり見られなかったのは，システムの扱いに慣れすぎたためと思われる．よって初期ではパフォーマンスは手袋差によって変動するが，その差は習熟によって縮まっていくことが分かった．

## 6. 今後の展望

昨今，IoT やビッグデータの有用性が指摘される中，Kinect の安易な入力デバイスとしての性能を活かし，現場から作業者の姿勢判定を行うことで，大量の作業者の姿勢データをログとして得ることが期待される．この膨大なデータをビッグデータとして扱い，作業者の動きを把握・解析することで作業者の安全管理や負担の軽減，より効率的な作業指示の検討を行うことができる可能性がある．Fig.12 は KinectSDK の Discrete Gesture Basics で骨格情報から人が座っているかどうかを判定する様子である．予備実験の結果，Kinect v2 装置から 3[m]程度の範囲内であれば，かなりの高精度で着座しているか立っているかの判別が可能であった．

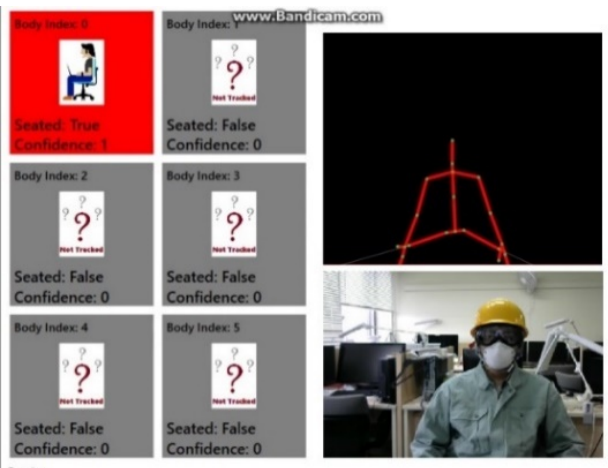


Fig.12 Judging by the KinectSDK whether he is sitting or not.

## 7. 結 言

本論文では，造船所内の工作現場においてジェスチャーによって PC から情報を出し入れするためのシステムについて検討を行い，kinect v2 を使用してタッチパネル入力として使用可能なシステムを構築し，標準的なブラウザ上で実験を行うことにより実用可能性について評価を行った．マウスによる入力と比較するとほぼ倍の時間を要するが，手袋などの装備を外すことなく操作可能な利点と，操作の習熟によってさらに時間短縮が可能である点を考慮すると実用レベルではないかと考えられる．

また，ブラウザを利用して 3D モデルを表示し，そのモデルを介してデータを出し入れするために HTML と Javascript, および X3DOM を利用したシステムを構築し，上記のジェスチャー入力システムと組み合わせて，目標とする操作をジェスチャーによって実現できることを示した．

## 参 考 文 献

- 1) 千葉慎二 他：「手術室向け非接触型画像操作システム：Opect」，Journal of Japan society of computer aided surgery, 2012, 14 号, pp150--151 (2012).
- 2) 医療系非接触閲覧システム iKINESYS, [http://ikinesys.com/about\\_ikinesys/](http://ikinesys.com/about_ikinesys/)
- 3) マウス検定試験 <http://portal.nifty.com/special05/02/05/>
- 4) フリーソフト ForbiddenFruit のスクリーンキーボード FlickKeyboard : <http://likehen.ninpou.jp/>