平成 28 年度学位論文

題目

深度センサを用いた
室内環境の再構築に関する研究

報告者

羽田　遼

徳島大学　大学院　先端技術科学教育部

博士後期課程　システム創生工学専攻

目次

序論 1

1.1 研究背景 1

1.2 研究の目的 3

1.3 本論文の構成 4

2 次元情報に対する処理 5

2.1 FAST 特徴点 5

2.1.1 特徴点検出 5

2.1.2 スケール検出 7

2.2 BRISK 特徴量 8

2.2.1 特徴量の記述 8

2.2.2 特徴量の比較 10

3 次元情報に対する処理 12

3.1 重心距離を用いた特徴点検出 12

3.1.1 ボクセルフィルタ 12

3.1.2 特徴点検出 14

3.1.3 極値検出 15

3.2 FPFH 特徴量 16

3 次元モデルの作成 19

4.1 対応点の統合 19

4.1 RANSAC 19

4.2 データの重ね合わせ 23

実験 25

5.1 実験 26

5.1.1 シーン 1 26

5.1.2 シーン 2 29

5.1.3 シーン 3 32

5.1.4 シーン 4 36

5.1.5　シーン 5 40
5.2　考察 44
結論 46
参考文献 48
謝辞 50

第 1 章

序論

1.1 研究背景

近年，ドローンや VR/AR などの市場が大きく拡大しており[1] [2]，それに伴い 3 次元に関する需要が大きくなっている[3]．3 次元モデルの利点として，通常の 2 次元画像の場合，一度作成してしまうと構図の変更が行えない．しかし，3 次元モデルの場合，一度作成した後でも構図の変更が容易に行え，何度でも利用することができる．また，2 次元画像と比べて 3 次元モデルなら実際の形状のイメージが容易である．そのため，3 次元モデルを利用することで，事故や災害時の被害予測や製品の完成結果の評価が可能になる[4]．しかし，3 次元モデルの作成には，3D モデリングの技術と測定する立体物に対する知識が必要であり，3 次元モデルを初心者が作成することは，非常に困難である．

そこで，上記のような問題を解決するために，本論文では，安価な Kinect センサを手で動かしながら被写体の複数方向の 3 次元データを取得し，それらを重ね合わせることで，単一の 3 次元モデルを生成する手法を提案する．手で Kinect センサを移動させるため，撮影の方向や距離がデータごとに異なるため，3 次元データ間での位置合わせをする必要がある．3 次元データの位置合わせの代表的な手法として，ICP(Iterative Closest Point)アルゴリズム[5]が挙げられる．これは 3 次元データ間での対応付けと対応結果から変換パラメータの推定を繰り返し計算することで，データ間の位置合わせを行う手法である．しかし 3 次

元データの点数が多いと，計算コストが膨大になる問題や初期値として与える値によっては正しい結果が得られないといった問題がある．この問題を解決するために，3 次元形状変化の周期性に着目し，3 次元位相限定相関法を用いたデータの高速・高精度な位置合わせ手法[6]を提案したが，データが x，y，z のそれぞれの軸で同時に回転した際に，正しく変化を推定できない問題がある．また，これらの手法の多くは 3 次元形状の変化を位置合わせに用いるため，凹凸変化の少ないシーンにおいて精度が低下する問題もある．

そこで，本論文では 3 次元情報による重ね合わせに加えて，カメラから取得した 2 次元画像をもとに，複数方向のデータを合成することで，3 次元形状変化の少ないシーンに対応した，3 次元モデルを生成する手法を提案する．まず Kinect センサを用いて 3 次元情報と 2 次元情報を取得する（図 1.1）．取得した情報に対して，それぞれ特徴点の検出と特徴量を算出し，特徴点間で対応付けを行う．そして，得られた情報から変換パラメータを推定することでデータの位置合わせを行う．2 次元特徴量には，処理速度を考慮して FAST 特徴点[7]と BRISK 特徴量[8]を用いる．3 次元特徴量には，すでに 2 次元情報から特徴点検出を行っているため，テクスチャを用いない特徴量が望ましい．また，3 次元特徴点算出は特徴点の算出に 3 次元情報の形状評価を行うため，形状評価の計算コストが高く処理時間に問題がある．そこで，注目点の周辺点群の占有率を計算することで特徴点の検出を行う手法[9]を参考にした，周辺点群の重心距離による特徴点検出を用いる．検出された特徴点には FPFH 特徴量[10]を付与する．

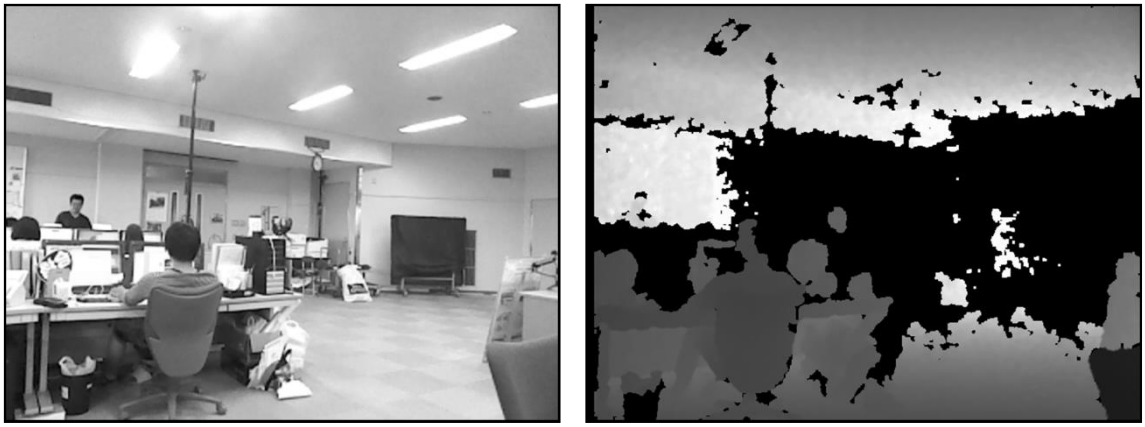
1.2　研究の目的

本研究では，Kinect センサで室内環境を撮影し，撮影したデータから室内環境を 3 次元モデルとして再現することを目的としている．室内データは，手で直接 Kinect センサを持ち，移動しながら撮影するため，被写体の撮影方向や被写体までの距離がデータごとに異なる．そのため，Kinect センサから取得した 2 次元画像と 3 次元情報に対して，特徴点検出と特徴量算出を行い，特徴点間で対応付けを行う．得られた対応情報から RANSAC[11]を用いて誤対応点の除去を行い，変換パラメータの推定を行う．得られた変換パラメータをもとに複数のデータを重ね合わせることで 3 次元モデルをする．

しかし，Kinect センサから得られる情報の中には，3 次元形状変化の少ないシーンやテクスチャ変化の少ないシーンが存在する．また，Kinect センサは赤外線パターンを投光し，そのパターンの歪みから対象の深度情報を取得するセンサであり，その撮影範囲も 0.8～4.0m と限られているため，図 1.1 からわかるように，Kinect センサから得られる 3 次元情報には，金属などの赤外線を反射する材質や黒髪といった吸収率が高い材質では距離が正しく測れない，被写体までの距離の影響で情報が存在しない領域がある．そのため，3 次元情報の精度が低い．2 次元画像のみの対応付けでは処理速度と処理精度は高いが，得られた対応点の座標に 3 次元データが存在せず，使えない可能性がある．そのため，得られた対応情報を組み合わせることで，3 次元形状変化が少ないシーンに対してはテクスチャ情報，テクスチャ変化の少ないシーンに対しては 3 次元形状による対応が得られるため，安定した 3 次元モデルの作成が行える．

すなわち本手法は，特殊な装置を使ったり厳密なキャリブレーションをせずに，安価な 3 次元センサを用いて連続して取得された複数方向の 2 次元画像と 3

次元情報から，複数次元の特徴点と特徴量を用いて単一の 3 次元モデルへと重ね合わせる，高精度な 3 次元位置合わせを用いた 3 次元モデリング手法である．



(a): カメラ画像　　(b): 深度情報

図 1.1：Kinect センサから得られる情報

1.3　本論文の構成

本論文の構成は，本章を含め 6 章により構成される．本章では，研究の背景と目的について述べ，本研究の動機について説明した．以下，第 2 章では，本研究に用いた 2 次元特徴点検出手法と特徴量について述べる．第 3 章では，本研究に用いた 3 次元特徴点検出手法と特徴量について述べる．第 4 章では，RANSAC を用いた誤対応情報の除去，データの重ね合わせの手法について述べる．第 5 章では，本研究の有効性の確認のために行った実験の一部を示し，考察を述べる．第 6 章では，本論文のまとめ，本研究の成果，今後の課題について述べる．

第 2 章

2 次元情報に対する処理

本章では，Kinect センサから取得した 2 次元画像間の対応付けに必要な特徴量の取得手法について説明する．まず，取得した画像に対して FAST を用いた高速特徴点検出を行う. その後, 各特徴点の BRISK 特徴量を求める. 本章では, まず FAST による高速特徴点算出の方法について述べ，BRISK 特徴量の計算方法, BRISK 特徴量の比較方法について説明する.

2.1 FAST 特徴点

2.1.1 特徴点検出

特徴点検出の有名な手法として，Scale-Invariant Feature Transform(以下 SIFT)[12]に用いられている Difference of Gaussian がある．これは，複数のガウシアン画像の差分を取ることでスケール変化に不変な特徴点を検出する方法である．しかし，この方法は多数の平滑化画像を作成する必要があるため，処理が低速であることが問題として挙げられる．これは，多くの情報に対して対応付けを行うにあたって大きな問題となる.

そこで, 処理が高速な特徴点検出手法の Features from Accelerated Segment Test(以下 FAST)特徴点検出[7]を用いる．FAST は図 2.1 のように注目画素 P と P を中心とする周囲 16 画素を比較することで高速にコーナーを検出する手法で

ある．周囲 16 画素の輝度が注目画素 P より T 以上輝度値が明るい，または暗い画素が N 個以上連続する場合，注目画素 P をコーナーとして検出する．

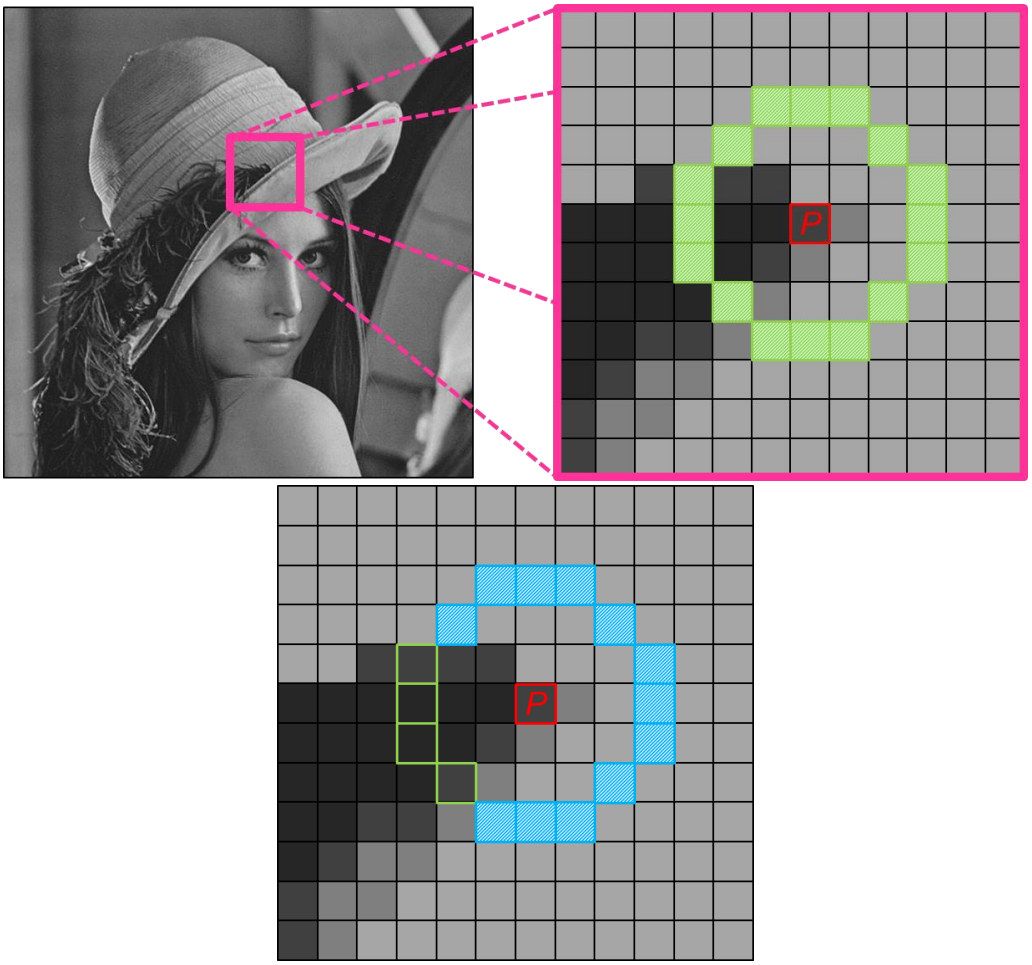


図 2.1：FAST によるコーナー検出

しかし，16 画素全てを参照すると，コーナー検出に時間がかかる．そこで，注目画素 P の周囲 16 画素の対角画素を比較することで非コーナーを高速に判定することができる．例として N=8 の場合，注目画素 P の上下の画素を比較したとき，最低でもどちらか一つの画素が注目画素 P よりも T 以上明るいまたは暗くなければ，8 個以上連続することはない．他の対角画素に対しても同様の比較を行う．そして，この条件を満たした場合のみ，改めて周囲の 16 画素を参照し，コーナー判定をすることで高速にコーナーを検出することができる．

2.1.2　スケール検出

上記で述べた FAST 特徴点の検出に加えて，特徴点のスケールを検出することで，スケールの異なる類似特徴点間の誤マッチングを防ぐことができる．また，スケールを計算することで，撮影距離の変化による画像上の対象物の大きさ変化に合わせた特徴量計算が可能となる．特徴点のスケールを検出するために，まず図 2.2 に示すように，画像を多段階に縮小したピラミッド画像を作成する．作成したピラミッド画像それぞれに対して FAST を適用し，特徴点検出を行うことで，特徴点のスケールを求める．

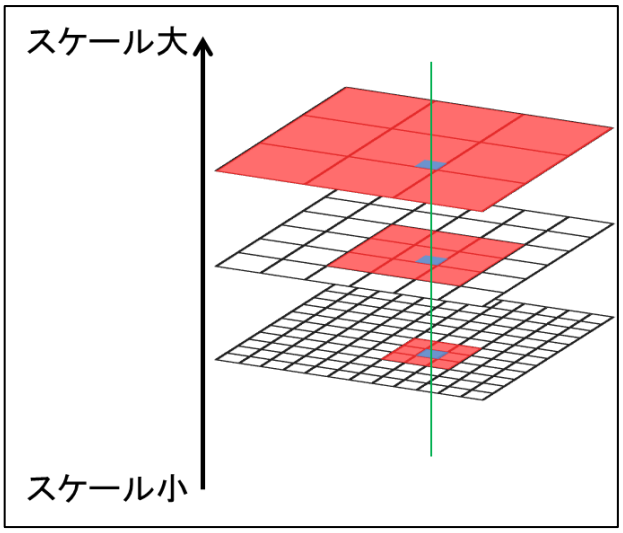


図 2.2：ピラミッド画像と特徴点スケールの関係

$$\mathrm{B} = \sum_{x:S_{p\to x}=brighter} \left|I_{p\to x} - I_p\right| - t$$
$$\mathrm{D} = \sum_{x:S_{p\to x}=darker} \left|I_p - I_{p\to x}\right| - t \qquad (2.1)$$
$$\text{レスポンス値 V} = \max(V, D)$$

しかし, ピラミッド画像それぞれに対して FAST による特徴点検出を行うと, スケールの異なる特徴点が同一座標に対して複数検出される．そのため，これらの特徴点からノイズなどの影響を受けにくい，適切なスケールの特徴点を求める．適切な特徴点を求めるために，FAST を適用する際に，式(2.1)を用いて FAST のレスポンス値 V を計算する．計算したレスポンス値 V を x，y，スケール方向の 3 次元空間で比較し，極大となる点を特徴点として採用する．図 2.3 に FAST 特徴点検出を行った例を示す.

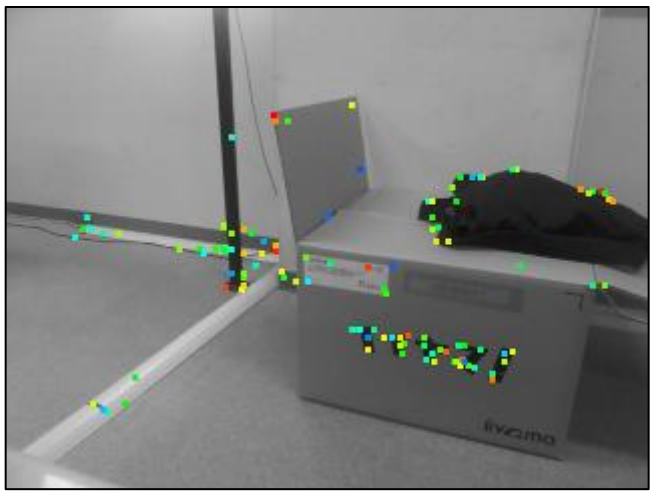


図 2.3：FAST 特徴点検出結果

2.2 BRISK 特徴量

2.2.1 特徴量の記述

FAST で検出した特徴点間で対応付けを行うために, 特徴点に対して特徴量の記述を行う．本研究では手で Kinect センサを持ったまま移動することでデータ

の取得を行うため，画像の回転変化に強い特徴量が必要となる．また，特徴点間の対応付けとして，数多く用いられている SIFT はこれらの問題に強い特徴量を取得することができるが，特徴量の比較において時間が掛かるといった問題がある．そこで，SIFT と同様に回転変化に頑健かつ高速な対応付けが可能な特徴量が必要となる．

そこで上記の問題に対応する特徴量として Binary Robust Invariant Scalable Keypoints(以下 BRISK)特徴量[8]の記述を行う．BRISK 特徴量は図 2.4 に示すように，中心に存在する特徴点周辺から等間隔にサンプリングされた 60 か所の画素値を比較することで特徴量を算出する手法である．

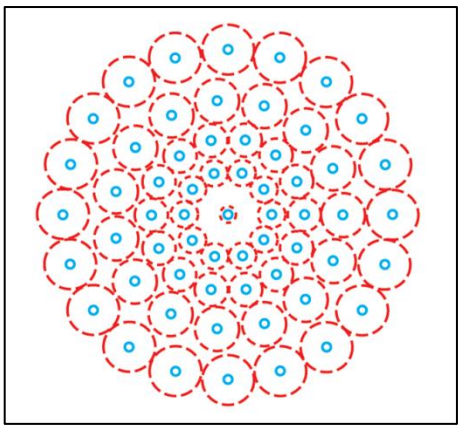


図 2.4：BRISK 特徴量のサンプリング間隔

まず，サンプリング点間の距離がδmax以下になるペアの集合を S ,δmin以上となるペアの集合を L とする．そして，集合 L のペア(P_i, P_j)から式(2.2)を用いて平均勾配ベクトルを求める．求めた平均勾配ベクトルを特徴点のオリエンテーション α とし， α に従って集合 S を回転させることで集合 S'を得る．

$$\mathrm{g}(P_i, P_j) = (P_j - P_i)\frac{I(P_j, \sigma_j) - I(P_i, \sigma_i)}{\|P_j - P_i\|^2}$$
$$\mathrm{g} = \begin{pmatrix} g_x \\ g_y \end{pmatrix} = \frac{1}{|L|}\sum_{(P_i, P_j)\in L} \mathrm{g}(P_i, P_j) \tag{2.2}$$
$$\alpha = \arctan2(g_y, g_x)$$

集合 S'のペア(P_i', P_j')から式(2.3)を用いて，画素値の大小関係からバイナリコードを作成する．作成されたバイナリコードが BRISK 特徴量となる．オリエンテーションの推定を行っているため，回転変化に強く，特徴量として画素値の大小関係を用いているため，照明変化に強い特徴量であることが知られている．また, FAST で検出したスケールに応じて特徴量の算出範囲を増減させることで, ある程度センサと対象物までの距離が変化し対象の大きさが変化してもマッチングが可能である．

$$\mathrm{b} = \begin{cases} 1 & I(P_j') > I(P_i') \\ 0 & otherwise \end{cases} \quad (P_i', P_j') \in S' \tag{2.3}$$

2.2.2 特徴量の比較

BRISK 特徴量は各特徴点につき，512 個ずつ存在するため非常に特徴量の要素数が多い．しかし，特徴量をバイナリコードで記述しているため，特徴量の類似度をハミング距離により求めることができる．ハミング距離は，2 つのバイナリコードの排他的論理和を求め，‘1’が立っているビットの数を数えること

で得られるため，高速に計算することができる．図 2.5 に算出した特徴量に比較結果から対応付けを行った例を示す．

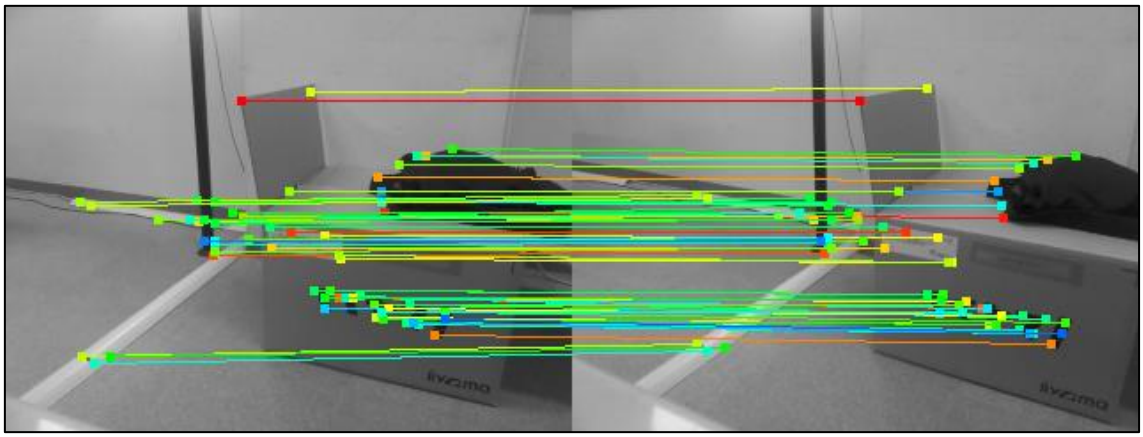


図 2.5：BRISK 特徴量による対応付け結果

第 3 章

3 次元情報に対する処理

本章では，Kinect センサから取得した 3 次元情報間の対応付けに必要な特徴量の取得手法について説明する．まず，取得した 3 次元情報に対して注目点周辺点群の重心距離を用いた特徴点検出を行う．その後，各特徴点の FPFH 特徴量を求める．本章では，まず注目点周辺の重心距離による特徴点算出の方法について述べ，FPFH 特徴量の計算方法について説明する．

3.1　重心距離を用いた特徴点検出

3.1.1　ボクセルフィルタ

FAST で検出した特徴点間で対応付けを行うために, 特徴点に対して特徴量の記述を行う．一般的に 3 次元情報から特徴点を取得する方法は，注目点周辺の 3 次元の形状評価を行うことで特徴点を取得する．しかし，3 次元点群の座標は実数値であるため，注目点の近傍点算出や形状評価の計算コストが非常に高く，処理に多くの時間が掛かる問題がある．そこで，3 次元特徴点の検出速度と検出位置の再現性を考慮して，注目点周辺の点の空間的な広がりを用いた特徴点算出手法を用いる. まず取得した 3 次元データに対してボクセルフィルタを掛け，$N_1 \times N_2 \times N_3$ ボクセルのデータに変換する．ボクセルフィルタとは，3 次元点群処理ライブラリである PCL に実装されているダウンサンプリング処理の一つで, 図 3.1 に示すように，空間上に一定のスケールで立方体のボクセルを配置し，そ

の各ボクセル内に存在する 3 次元点群を点群の重心座標に置き換える処理である．ボクセルフィルタを掛けることで，ボクセルのスケールに応じて点を削減することができるため高速化につながる．またボクセルフィルタにより，注目点の近傍点が注目インデックスの近傍インデックスに配置されることで，近傍点が高速に算出できる．本研究では，各ボクセルのスケールは，3 次元点群が各ボクセルに 1 点含まれる程度のスケールを目安に設定している．ボクセルフィルタの処理の内容を図 3.1 に示す．図 3.1 では青色の格子でボクセルを，赤色の点で 3 次元点群を，各ボクセル内に存在する 3 次元点群の重心を緑色の点で表している．

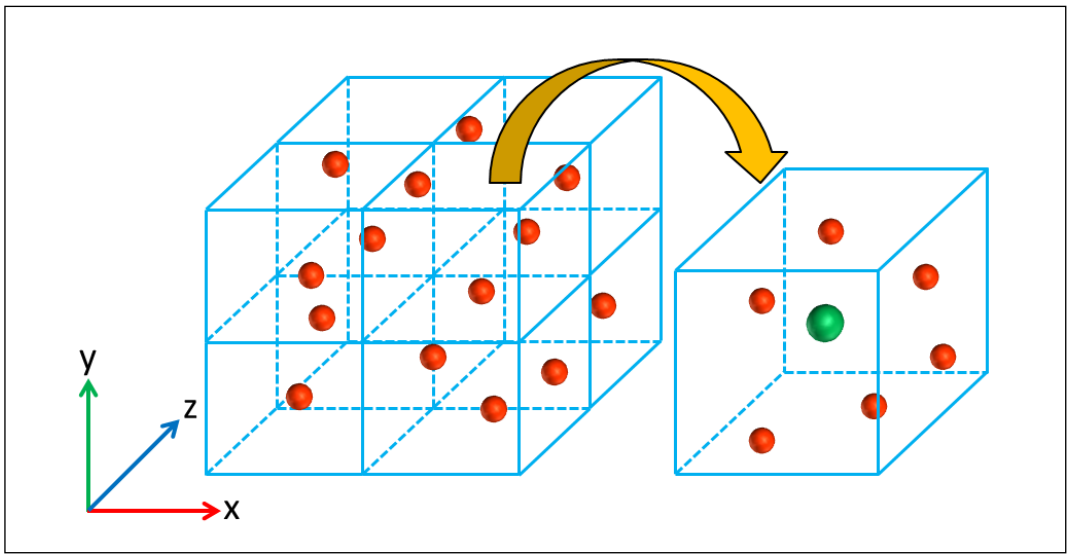


図 3.1：ボクセルグリッドマップの例

3.1.2　特徴点検出

次に占有率を用いて特徴点の検出を行う手法[9]について説明する．まず，ボクセル v_i の周囲を半径 r の球状にサンプリングする．このとき，球の外周上のボクセルのみを用いることで処理を高速化する．サンプリングしたボクセル内の点数を数えて，その割合（占有率）から特徴点の算出を行う．このようにデータ周辺の点の数を数えるだけの処理であるため，非常に高速な特徴点検出が行える．検出する特徴点としては，オクルージョンなどの影響を考慮して，様々な視点から見えやすい凸形状であることが重要であると考えられている．図 3.2 に示すように，凸形状は平面形状と比較して注目点付近の点群の占有率が低いことがわかるため，得られた占有率が設定した閾値以下の場合特徴点とすることで，凸形状の先端を特徴点として検出する．

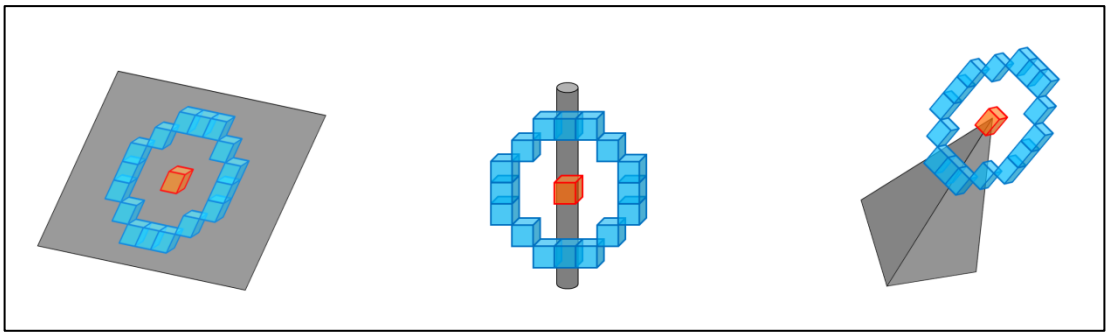


図 3.2：様々な形状に対する占有率の例

しかし，図 3.2 の例でも示しているように，単純に占有率を計算すると凸形状の先端以外にも，棒形状の物体からも特徴点を検出してしまう問題がある．棒形状から特徴点を検出してしまうと，開口問題から誤マッチングしてしまう恐れがある．そこで，本研究では特徴点の検出に占有率ではなく，近傍点群の重心距離を用いる．球の外周上ボクセル内の点の重心 G_i を算出し，ボクセル vi

から重心 G_i までの距離から特徴点の算出を行う．重心までの距離を用いることで，点が散らばっている場合や棒形状の場合に距離が近くなり，凸形状の先端では距離が遠くなることから，より適切な特徴点検出が行える．また，占有率と比較しても重心までの距離を計算するだけなので高速な処理が可能である．

3.1.3 極値検出

重心距離を用いて検出した特徴点からより高精度な点のみを得るために，特徴点の絞り込みを行う．まず，特徴点らしさの指標である重心距離を，注目点から x, y, z 方向に隣接する 26 ボクセルに存在する点に対しても同様に算出し，その値を比較し極大ならば特徴点として検出する．また，3 次元点群の密度はセンサから被写体までの距離に応じて変化するため，センサから離れている領域は点の密度が非常に小さく，隣接ボクセルに点が存在しないことがある．そのような孤立点が特徴点として検出されると精度に影響があるため，隣接 26 ボクセル内に特徴値が計算できなったボクセル，すなわち特徴値が 0 であるボクセルが 26 ボクセルの半数を占める場合は，注目点は孤立点であると判断して特徴点検出を行わない．図 3.3 に重心距離を用いて特徴点の検出を行った例を示す．

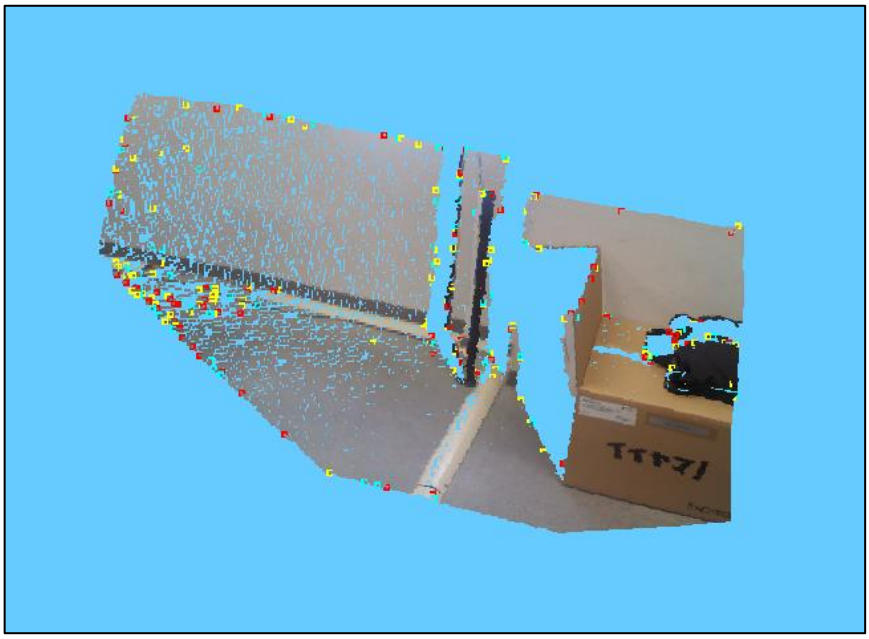
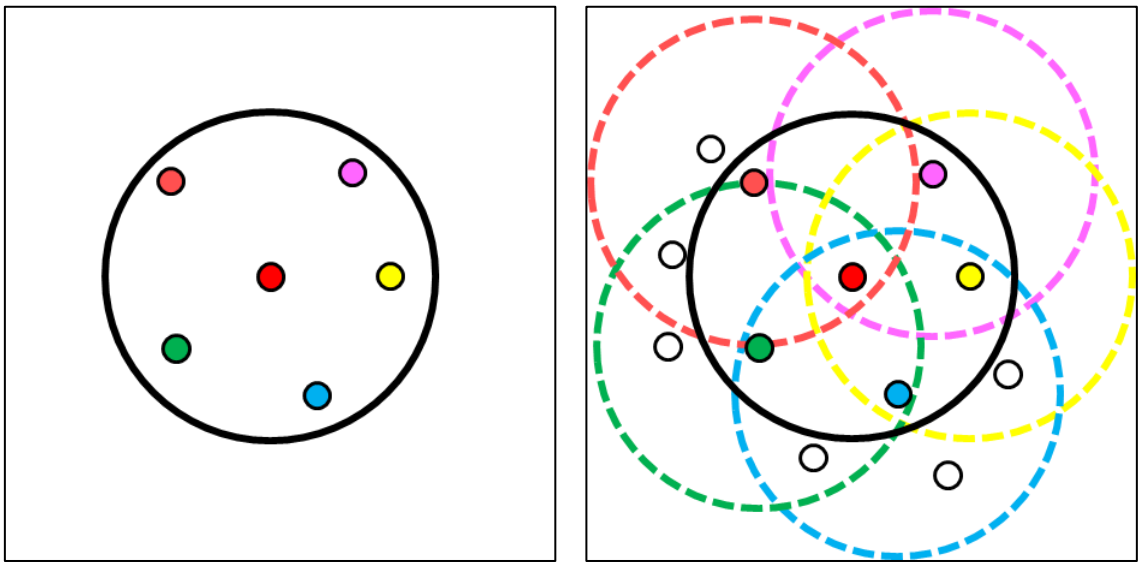


図 3.3：重心距離を用いた特徴点検出結果

3.2　FPFH 特徴量

抽出した 3 次元特徴点に対して, Fast Point Feature Histogram (以下 FPFH) 特徴量[10]の記述を行う．FPFH 特徴量は図 3.4(a)に示すように，特徴点から一定の範囲に存在する点を抽出し, 次に図 3.4(b)に示すように抽出した点の周囲に存在する近傍点を抽出する．抽出した近傍点群から図 3.5 に示すように，2 点間の法線ベクトルのばらつきを計算する．抽出点ごとに計算した値を統合することで，特徴点の特徴量を取得する手法である．



(a)：注目点周辺の近傍点検出　　(b)：近傍点群周辺の近傍点検出

図 3.4：FPFH 特徴量の計算範囲

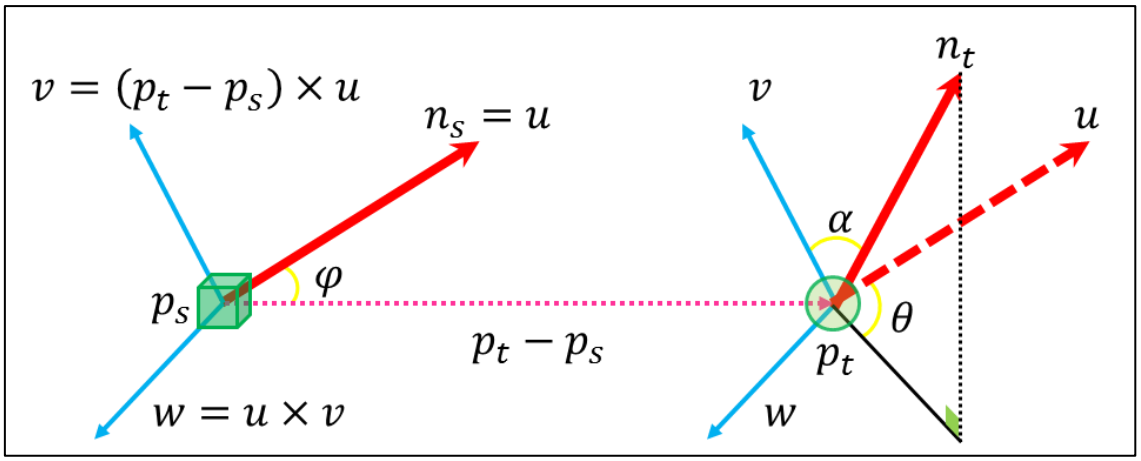


図 3.5：法線ベクトルのばらつき

FPFH 特徴量は注目点と近傍点群を組み合わせて，複数の 2 点組を作成し，その 2 点組から α，θ，ϕ，p_t-p_s を計算し，ヒストグラム化することで特徴量を算出している．ヒストグラム特徴量であるため，ある程度の外乱変化に強く，注目点周辺の法線ベクトルを特徴量としているため，回転変化などにも強い特

徴量である．また PFH 特徴量[13]と比べて，作成する 2 点組の組み合わせ数を削減しており，高速に特徴量計算できることでも知られている．図 3.6 に FPFH 特徴量を用いて対応付けを行った結果の例を示す．

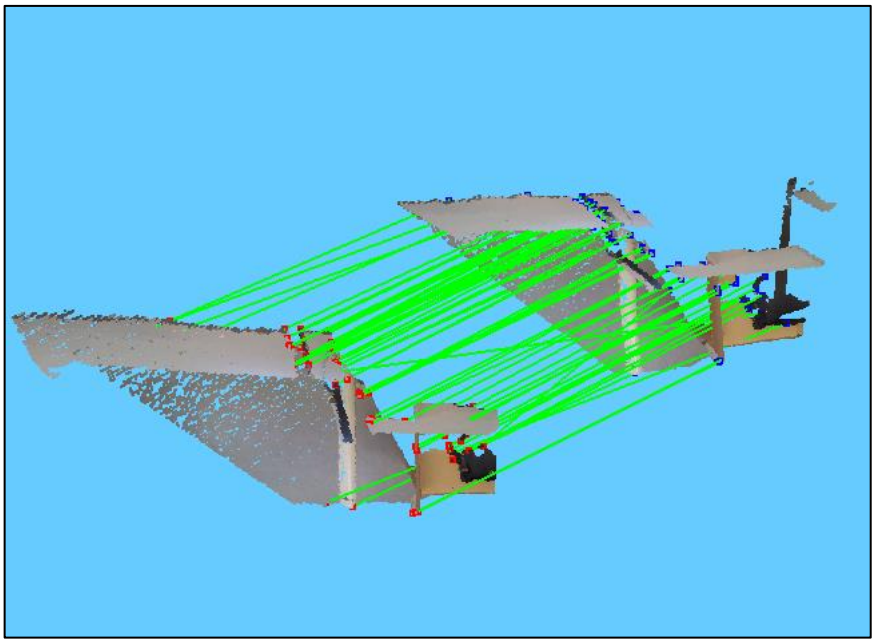


図 3.6：FPFH 特徴量による対応付け結果

また，本論文では Kinect センサを手で持ち移動させながら撮影しデータを取得するが，連続して取得したデータ間でマッチングを行うため，連続したデータ間ではデータの大きさや向きが大きく変化することはないと考えている．そのため，3 次元特徴量でのマッチングには，撮影距離の変化による密度変化を考慮せずに対応付けを行う．

第 4 章

3 次元モデルの作成

本章では，2 次元画像と 3 次元情報に対する対応付け結果をもとに，3 次元モデルの作成を行う方法について説明する．まず特徴量を用いた対応付けの結果には，多くの誤対応点が存在するため，RANSAC[11]を用いて誤対応情報の除去を行う．その後，対応情報をもとにデータ間の変換パラメータを計算し，1 フレーム前のデータに重ね合わせる．本章では，まず RANSAC による誤対応情報の除去方法について述べ，次にデータの重ね合わせ手法について説明する．

4.1　対応点の統合

2 次元特徴量と 3 次元特徴量，それぞれの対応付け結果の統合を行う．Kinect ならば 2 次元画像と 3 次元情報を同時に得ることができるため，2 次元対応点が存在する座標インデックスと同じインデックスに存在する 3 次元座標に変換することで 2 次元対応点を 3 次元対応点に変換し，FPFH 特徴量により取得した対応点情報と統合する．

4.1　RANSAC

特徴量による対応付けには，類似の特徴点間での誤対応が存在し，誤った対応情報から変換パラメータの計算を行うと，正しくデータを重ね合わせることができず，作成されるモデルの精度が低下する原因となる．そのため，各特徴量

での対応付け結果から誤対応結果の除去をする必要がある．測定値から外れ値を除去するロバスト推定手法では，最小二乗法が多く用いられている．最小二乗法は測定値と近似結果の二乗誤差の和が最小となるように係数を決定する手法である．しかし，最小二乗法では図 4.1 のように測定結果に外れ値が多く含まれている場合や大きい外れ値が存在するとき，正しい推定結果を得ることができない．

そこで，RANSAC を用いて測定結果から外れ値の削除を行う．RANSAC とは，ロバスト推定の一つである．以下に外れ値を含んだデータに対して，直線検出を行う例を示す．まず，測定結果からランダムで n 個のデータを抽出してパラメータの計算を行う．次に，求めたパラメータと抽出したデータ以外のデータとの誤差を計算し，誤差が閾値以内にあるデータの数を求めたパラメータに投票する．これを k 回繰り返し，図 4.2 のように投票数が最大となったものを仮パラメータとする．

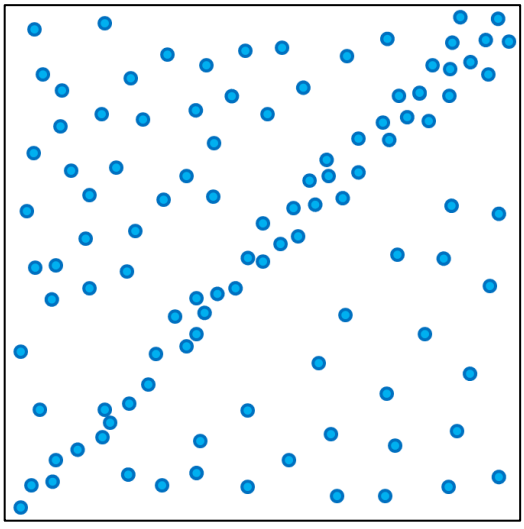


図 4.1：測定データの例

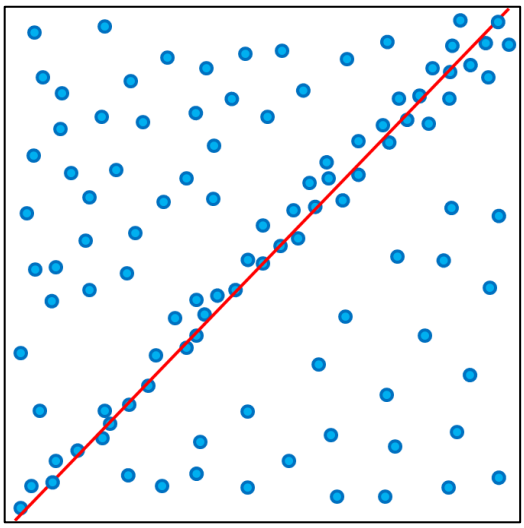


図 4.2：仮パラメータ推定結果

最後に仮パラメータを適用し，誤差が許容範囲内のデータを抽出し，図 4.3 のように抽出したデータからパラメータを計算することで，誤差やノイズなどの影響を除去した結果を得ることができる．

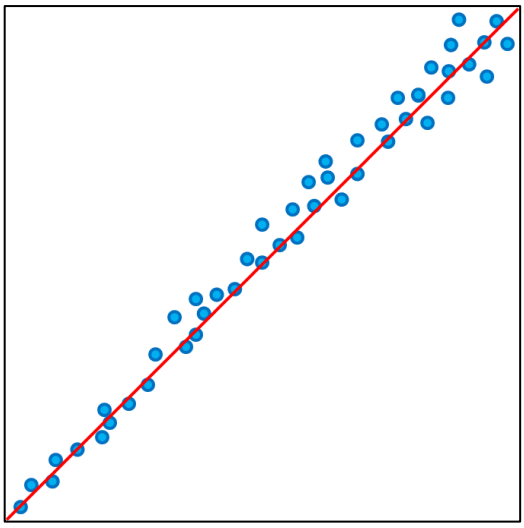
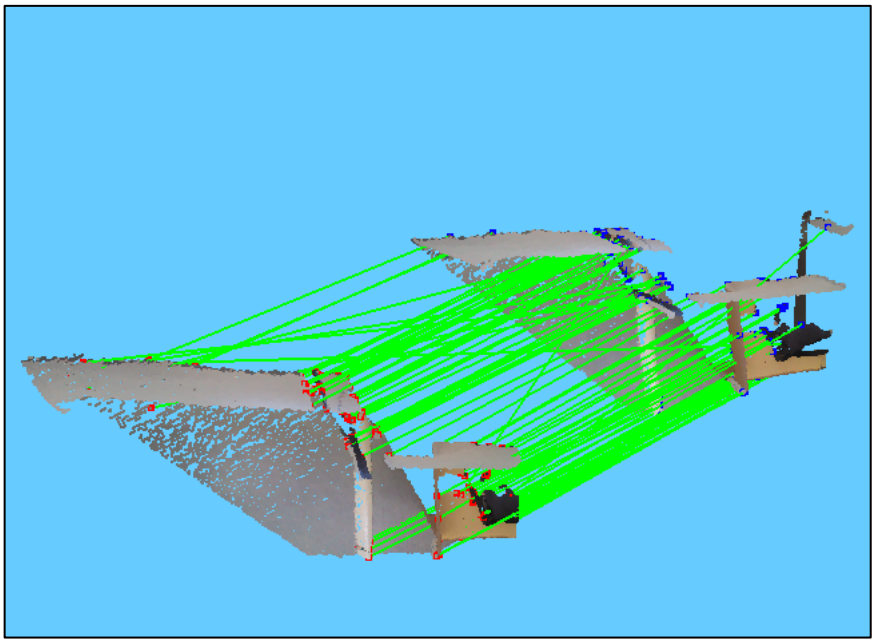


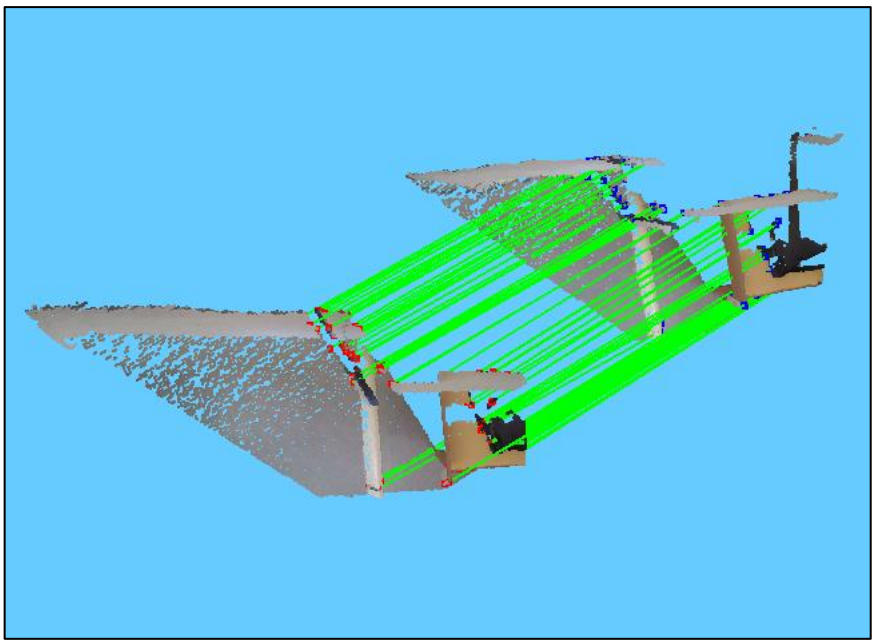
図 4.3：RANSAC による推定結果

本研究では，ランダムに 12 点抽出を行い，500 回変換パラメータの推定，変換後の対応点間の距離を計算し，その値が 0.03 以下の点の数が最大となる変換パラメータを仮パラメータとして用いて，誤対応情報の除去を行う．実際に

Kinect から取得したデータに対して, RANSAC を行う前後での結果の比較を図 4.4 に示す.



(a): RANSAC による対応点除去前



(b): RANSAC による対応点除去後

図 4.4：RANSAC による誤対応点除去結果の比較

4.2　データの重ね合わせ

RANSAC による誤対応点の除去により，精度の高い対応点の取得した．次に得られた対応情報からデータの重ね合わせを行う．本研究では Kinect センサを手に持って移動しながら撮影を行うため，取得したデータごとに点群の密度や向きが変化している．しかし，室内環境を撮影したものであるため，短い撮影時間内で大きく形状変化することはないと考えられる．そこでデータの重ね合わせには，データの形状を変化せずに平行移動と回転変化の補正を行うために，対応情報から微分補正法を用いて剛体変換パラメータの計算を行い，剛体変換を用いて重ね合わせていく．

しかし，Kinect センサでの計測誤差や対応点間の位置の僅かなズレ，変換パラメータの計算誤差など，様々な要因で重ね合わせ時に正しく重ならず，わずかな誤差が発生する．少ないデータ数ならば大きな問題にはならないが，多数のデータを重ね合わせる際には誤差が累積されていくため，問題となる．そこで，室内環境の 3 次元モデルの作成という目的から，データの最初と最後に重なる領域が存在すると考えられる．そのため，この特徴を用いて誤差の補正を行う．

まず，図 4.5 に示すように，変換パラメータを求める際に，Kinect センサで撮影した 3 次元データを前半部分と後半部分で分割し，それぞれを剛体変換により重ね合わせることで，前半部分と後半部分の位置合わせ済み 3 次元モデルを作成する．そして，2 つの 3 次元モデルを 3 次元アフィン変換で重ね合わせることで単一の 3 次元モデルデータを作成する．通常アフィン変換で重ね合わせると，データの形状を変換して重ねるため，少ない対応点情報では，データの形状が無理やり変えられてしまい，正しくモデルが作成されないことがある．

しかし，データの前半と後半で 3 次元モデルを作成することで，データの両端の対応点情報を用いることができるため，大きくデータの形状が変えられる恐れがない．また，アフィン変換で重ね合わせることで，撮影時や重ね合わせ時に発生したデータの誤差を補正することができる．

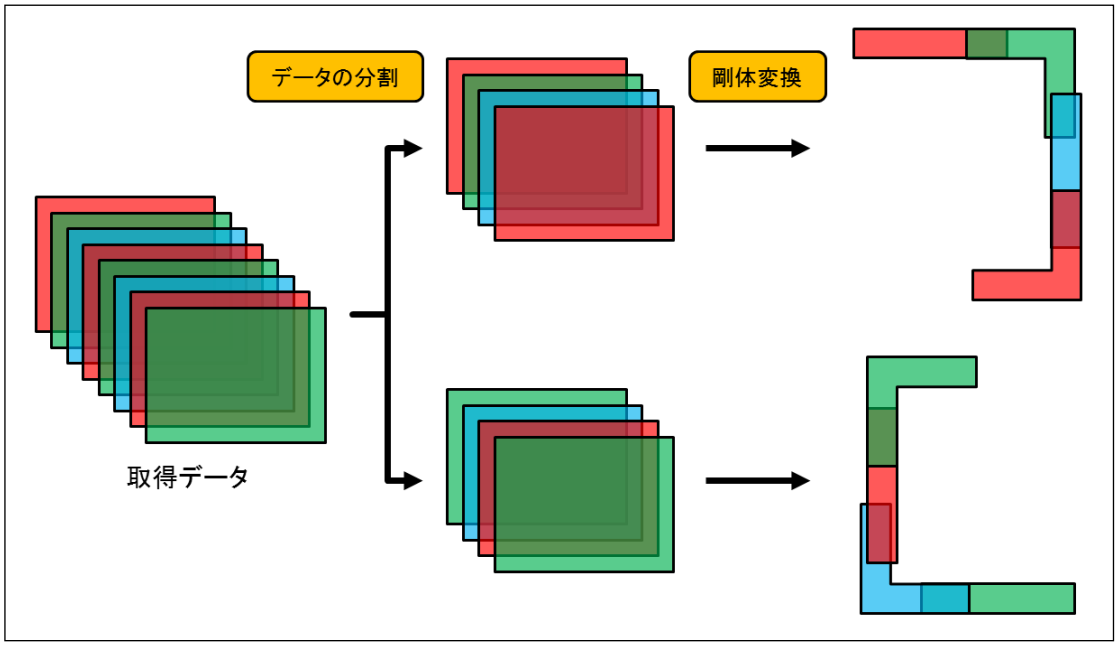


図 4.5：データの重ね合わせ処理の流れ

最後に，撮影距離の変化によりデータ毎に点群の密度が異なることや多くのデータを重ね合わせて単一のモデルを作るため，データの重なる領域で多くの点群が存在しておりデータの無駄がある．作成した 3 次元モデルの点群が非常に多く描画に時間がかかるといった問題があるため，前章で説明したボクセルフィルタを本来のダウンサンプリング処理として用いることで，データの密度を均一にしながら，3 次元モデルの点群の削減を行う．

第 5 章

実験

本章では，本研究の有効性を検証するため，Kinect センサより取得した室内の 2 次元画像と 3 次元情報を用いて，3 次元モデリングを行った例を示す．

まず，本研究で使用したセンサの仕様を表 1 に示す．そして，実験シーンごとに，撮影したデータの 2 次元画像と 3 次元データの一部を示し，次に本アルゴリズムで作成した 3 次元モデルを上から見た結果とモデルの一部を拡大した結果を示す．シーン 3 から 5 に関しては，作成したモデルの精度検証に実測値とのモデルの大きさ比較を行った結果を示す．また，シーン 1 と 2，5 に関してはデータの重ね合わせ精度の確認のためにデータの重ね合わせ時のアフィン変換による補正を行わないシーンであり，シーン 3 と 4 は重ね合わせのアフィン変換による補正を行ったシーンである．最後に各シーンの重ね合わせ結果についての考察を述べる．

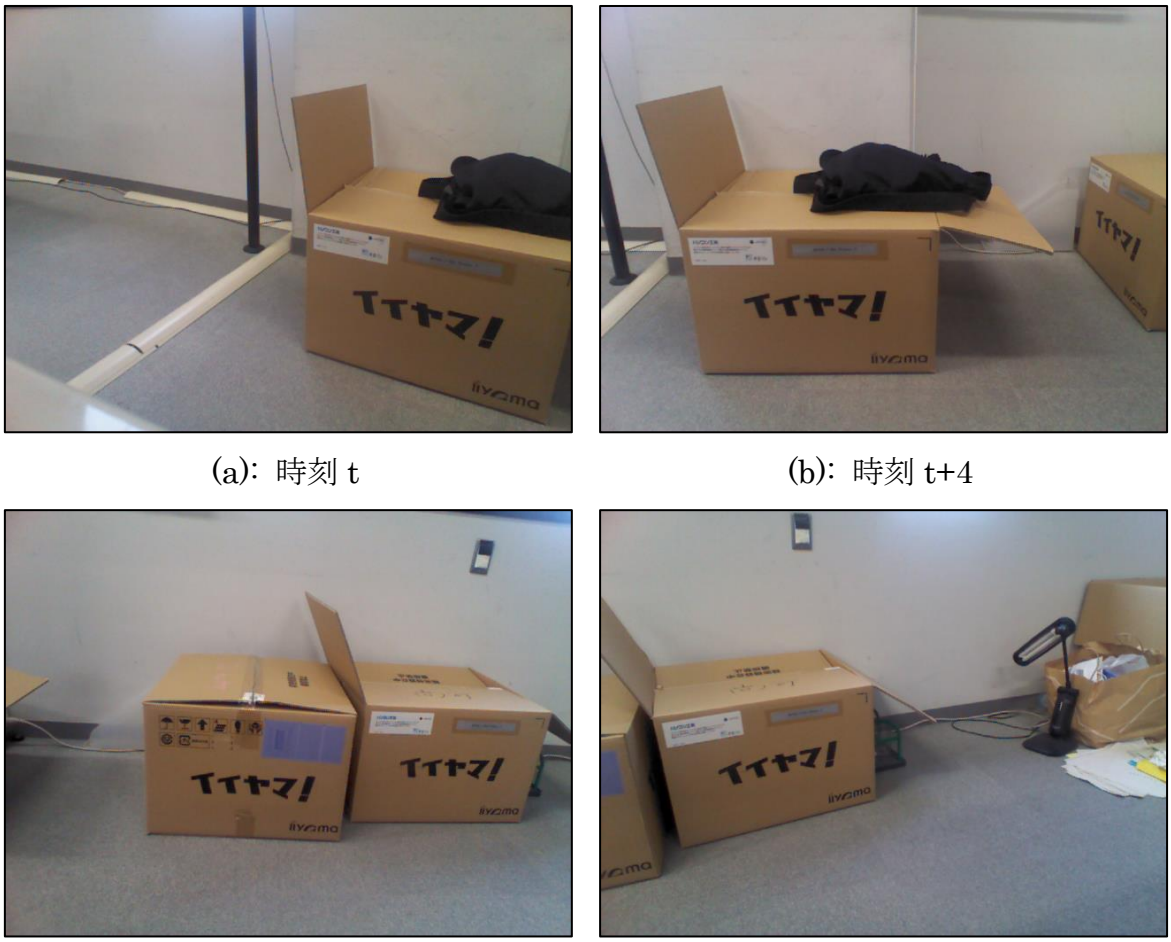
表 1: Kinect センサの仕様

RGB カメラ	解像度	640×480
	fps	30fps
深度センサ	解像度	640×480
	fps	30fps
深度の取得範囲		0.8～4.0m
画角	水平	57°
	垂直	43°

5.1　実験

5.1.1　シーン 1

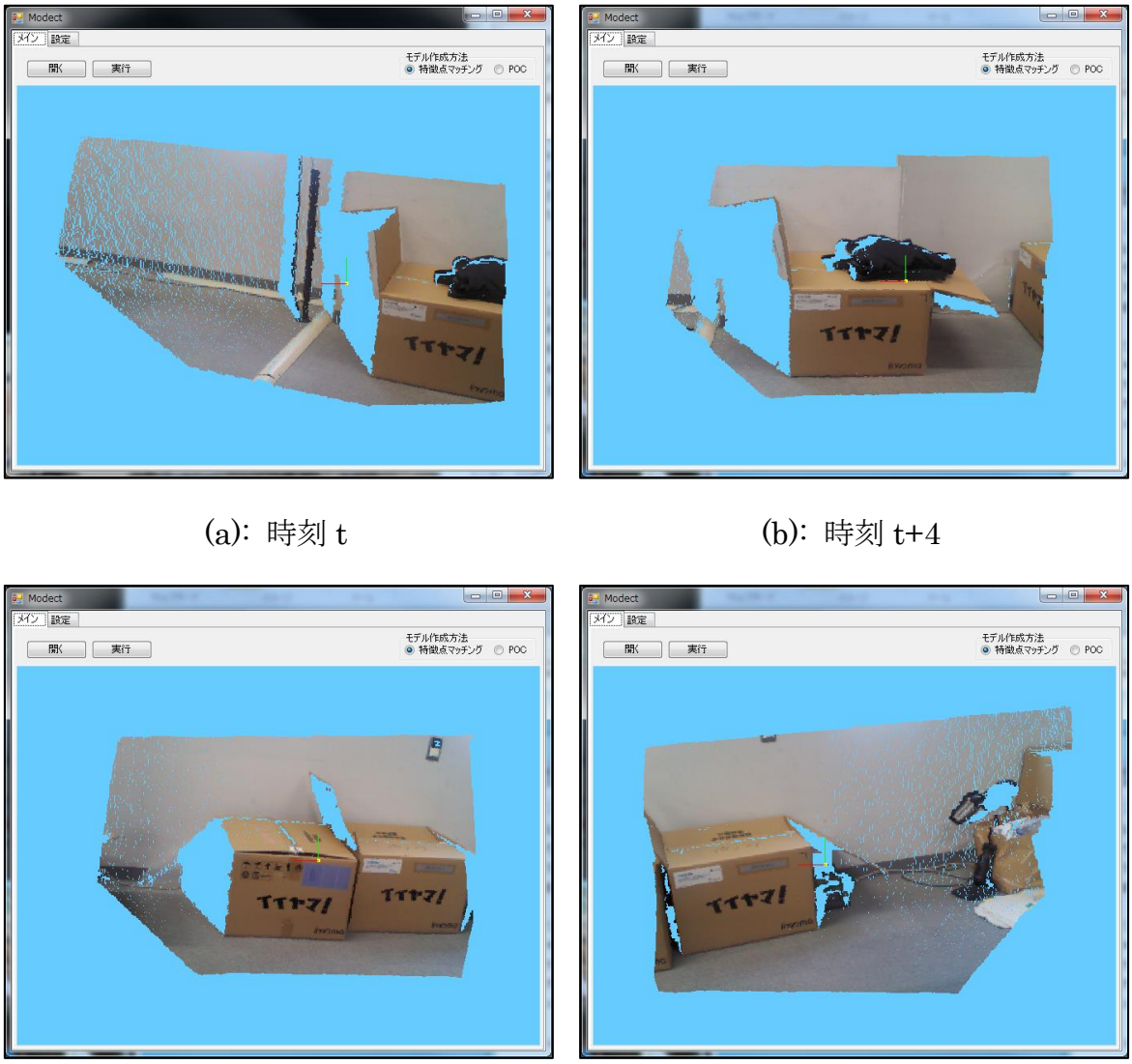
シーン 1 は Kinect センサを手に持ち，横方向に Kinect センサと移動しながら撮影したシーンである．



(a): 時刻 t　　(b): 時刻 t+4

(c): 時刻 t+8　　(d): 時刻 t+12

図 5.1: Kinect センサから取得した 2 次元画像



(a): 時刻 t　　(b): 時刻 t+4

(c): 時刻 t+8　　(d): 時刻 t+12

図 5.2: Kinect センサから取得した 3 次元情報

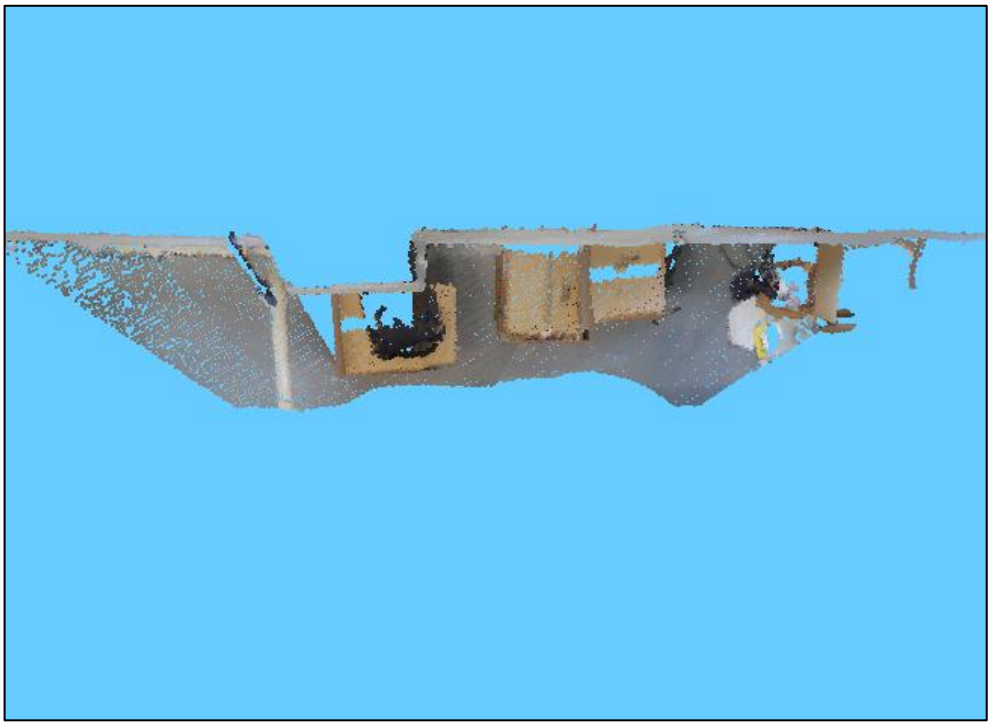


図 5.3: 3 次元モデルの作成結果

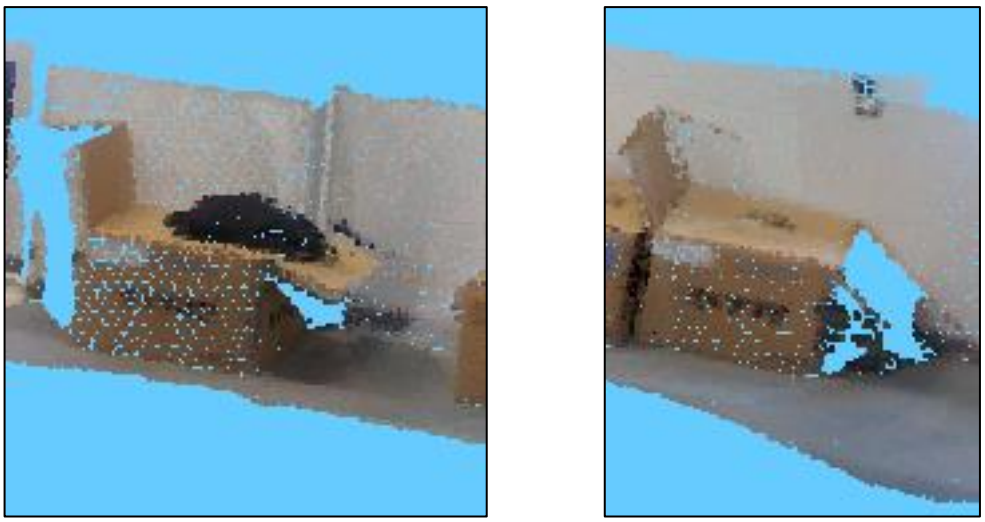
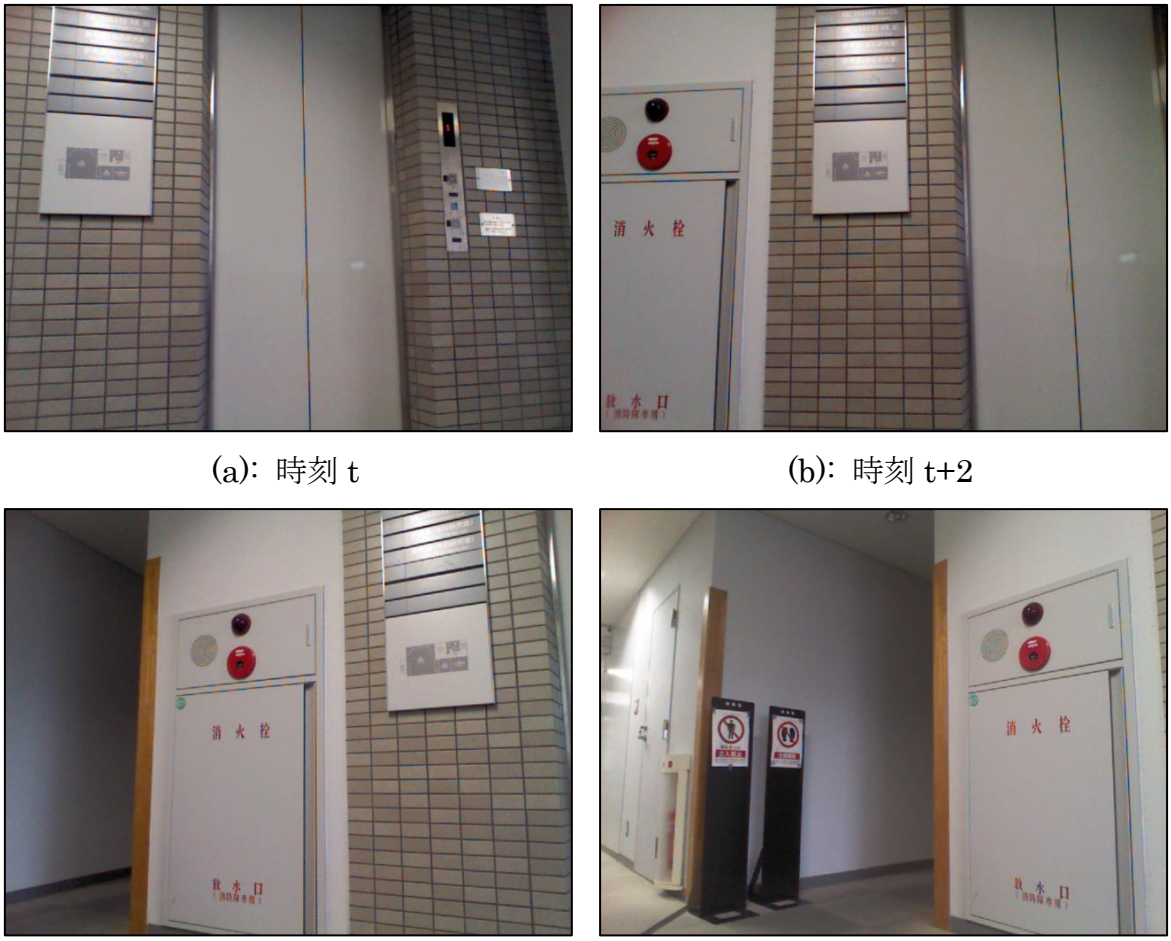


図 5.4: 3 次元モデルの拡大結果

5.1.2　シーン 2

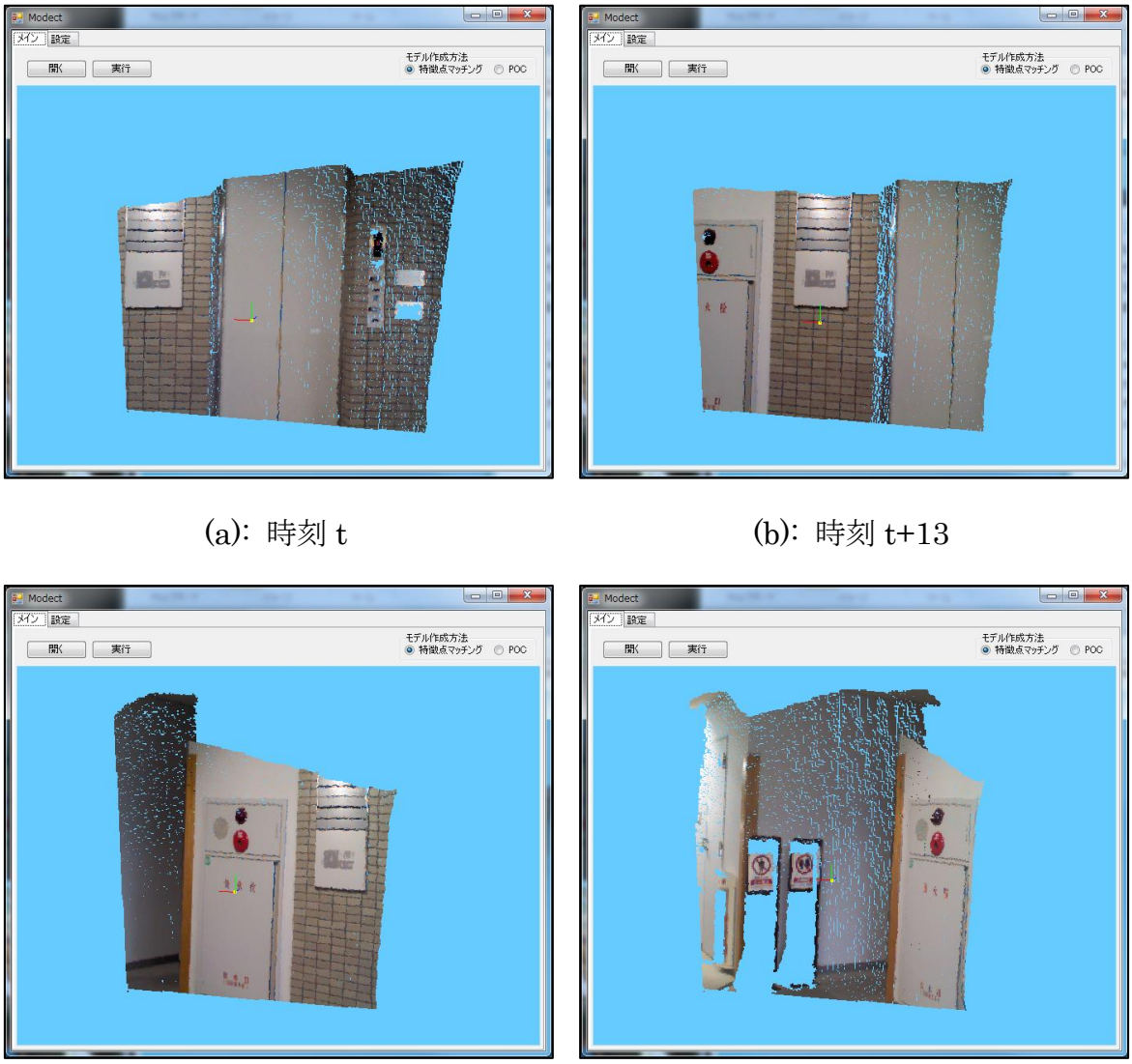
シーン 2 は Kinect センサを手に持ち横方向に回転させながら撮影したシーンである.



(a): 時刻 t　　(b): 時刻 t+2

(c): 時刻 t+4　　(d): 時刻 t+6

図 5.5: Kinect センサから取得した 2 次元画像



(a): 時刻 t　　(b): 時刻 t+13

(c): 時刻 t+26　　(d): 時刻 t+39

図 5.6: Kinect センサから取得した 3 次元情報

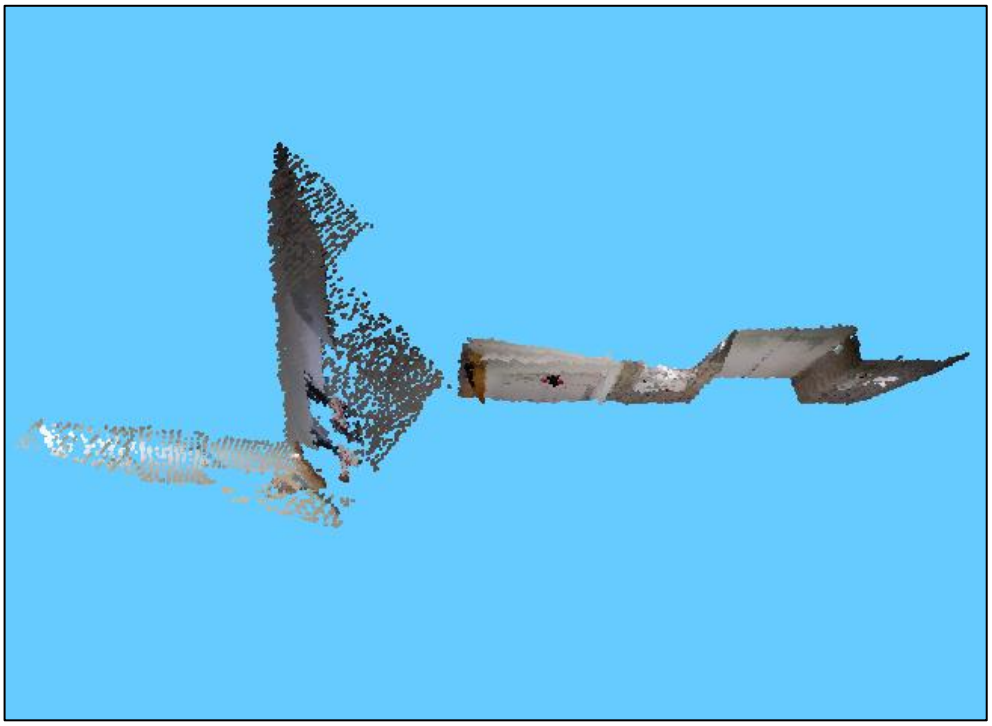


図 5.7: 3 次元モデルの作成結果

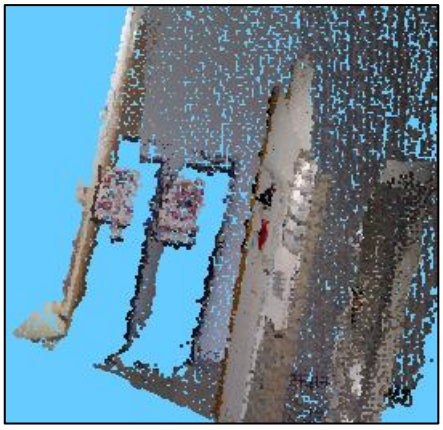


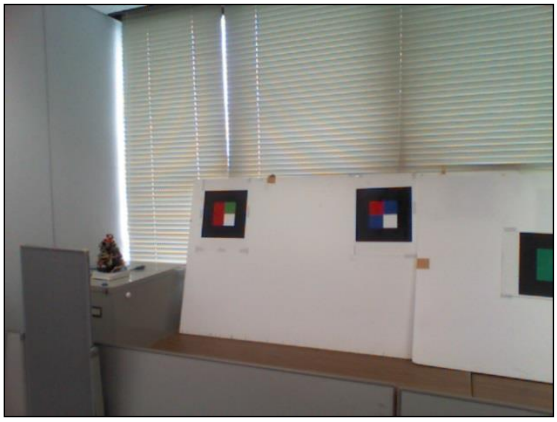
図 5.8: 3 次元モデルの拡大結果

5.1.3　シーン 3

シーン 3 は横 4.9m，奥行き 5.7m の部屋で，形状変化もテクスチャ変化も多いシーンである．

(a)：時刻 t

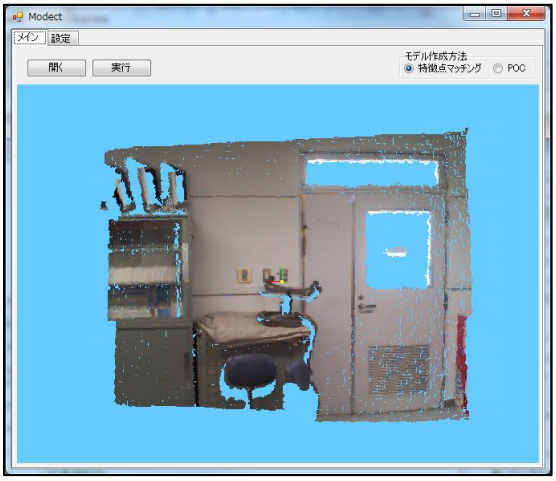
(b)：時刻 t+13



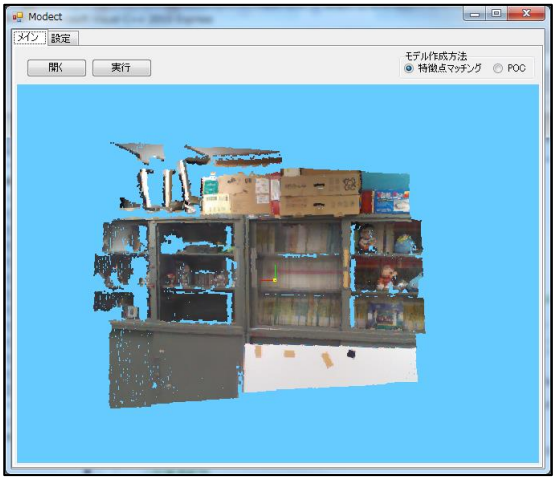
(c)：時刻 t+26

(d)：時刻 t+39

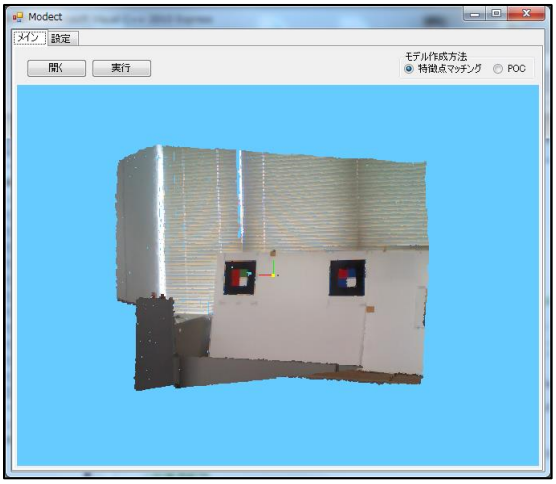
図 5.9: Kinect センサから取得した 2 次元画像



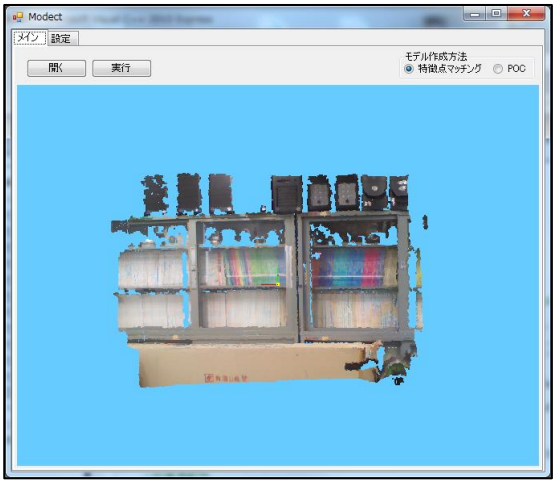
(a): 時刻 t



(b): 時刻 t+13



(c): 時刻 t+26



(d): 時刻 t+39

図 5.10: Kinect センサから取得した 3 次元情報

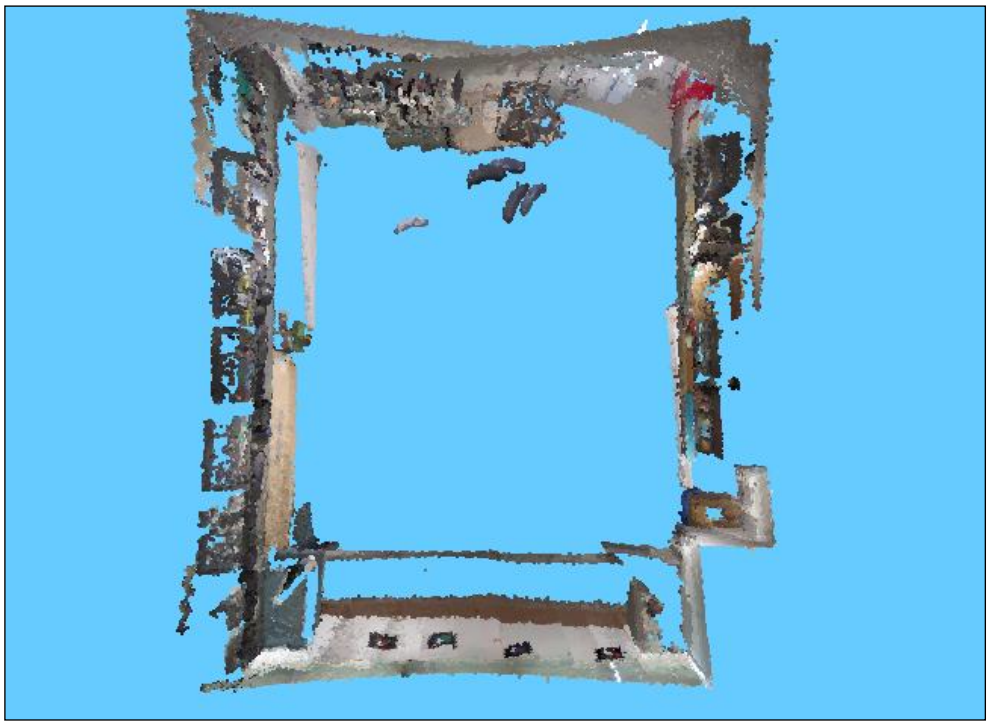


図 5.11: 3 次元モデルの作成結果

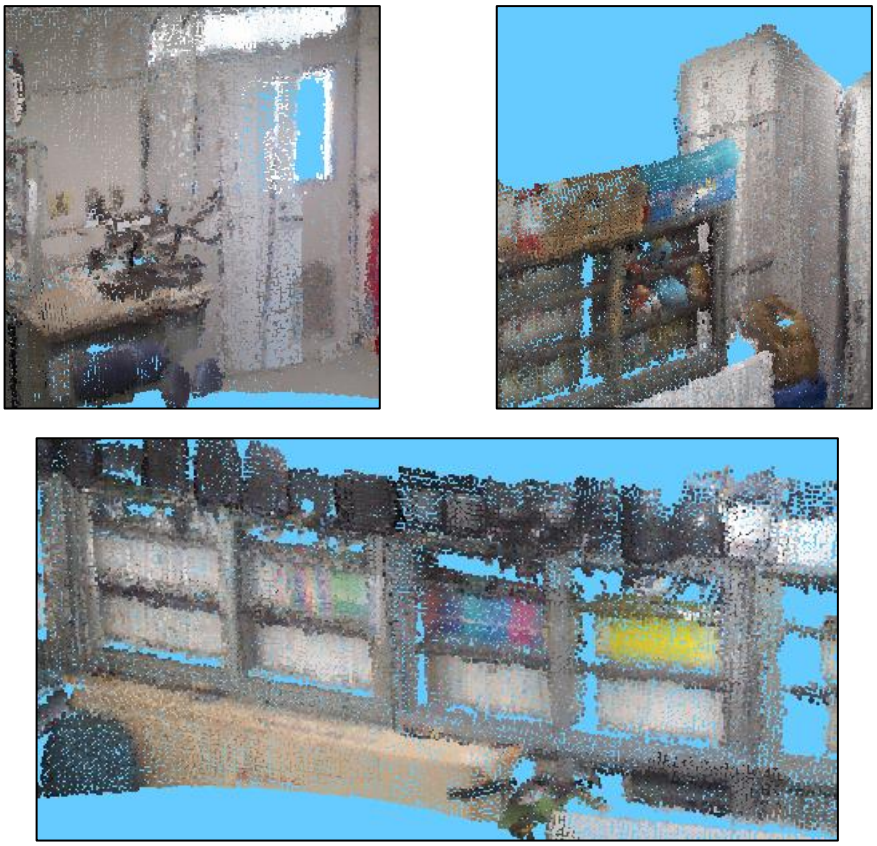


図 5.12: 3 次元モデルの拡大結果

表 2: 作成したモデルの大きさの推定結果

	実測値[m]	推定値[m]	誤差[%]
奥行	5.70	5.60	1.8
幅	4.90	4.80	2.0

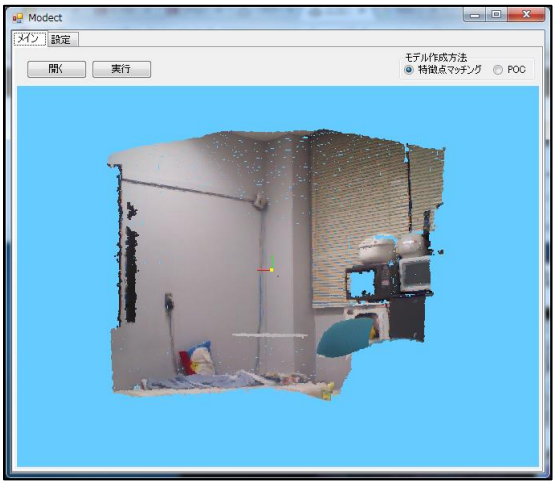
5.1.4　シーン 4

シーン 4 は横 5.55m，奥行き 6.8m の部屋で，データの前半部分では形状変化もテクスチャ変化も多く存在しているが，データの後半部分では，白い壁やテレビの液晶などが存在し，特徴が得られない情報量の少ないシーンである．

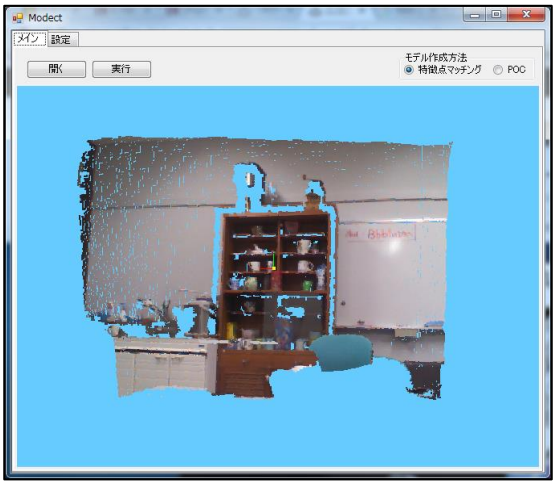
(a): 時刻 t　　(b): 時刻 t+12

(c): 時刻 t+24　　(d): 時刻 t+37

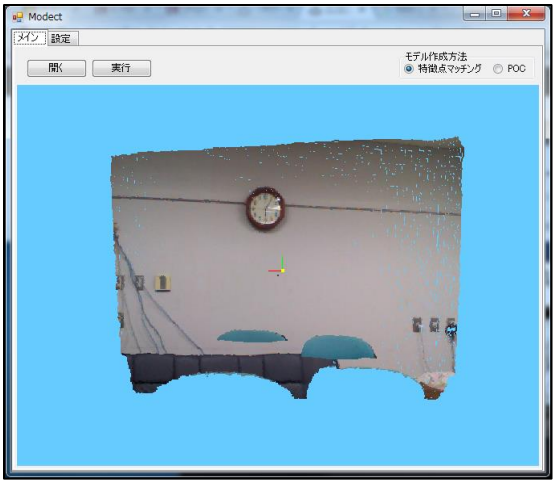
図 5.13: Kinect センサから取得した 2 次元画像



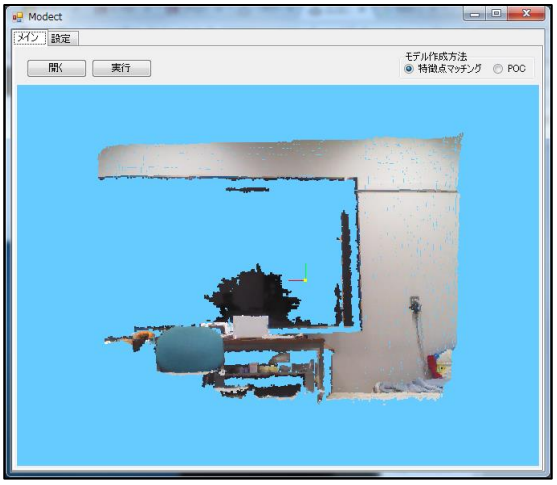
(a): 時刻 t



(b): 時刻 t+12



(c): 時刻 t+24



(d): 時刻 t+37

図 5.14: Kinect センサから取得した 3 次元情報

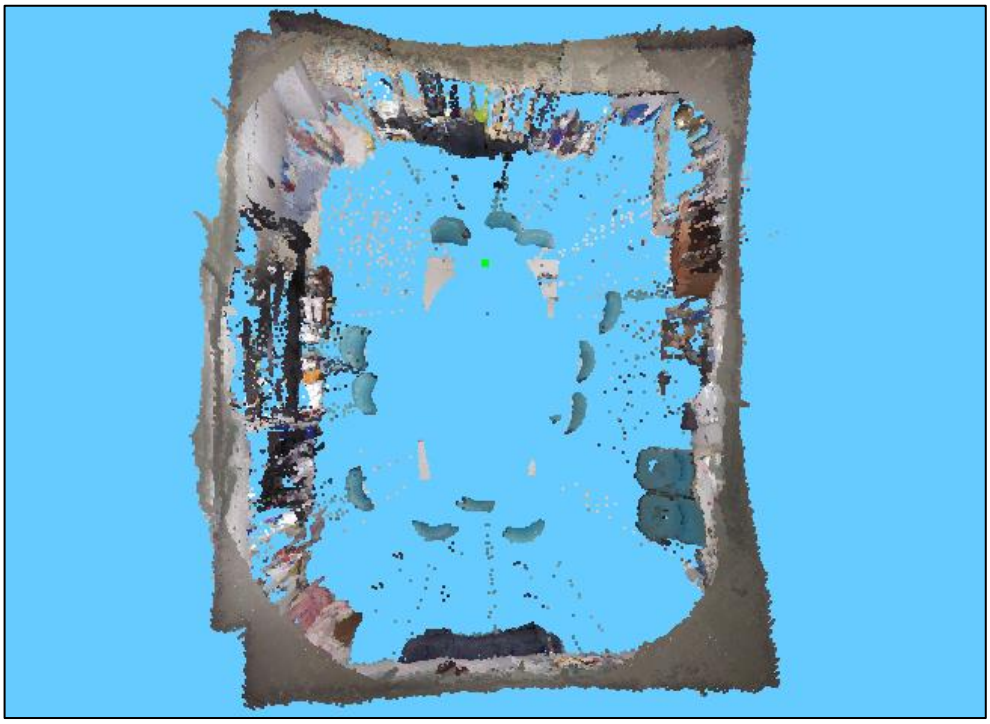


図 5.15: 3 次元モデルの作成結果

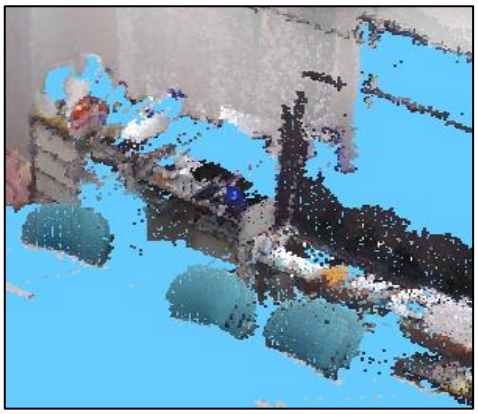


図 5.16: 3 次元モデルの拡大結果

表 3: 作成したモデルの大きさの推定結果

	実測値[m]	推定値[m]	誤差[%]
奥行	6.80	6.65	2.2
幅	5.55	5.26	5.2

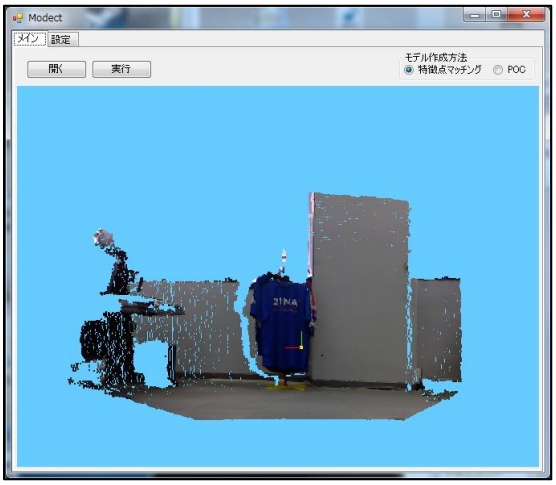
5.1.5　シーン 5

シーン 5 は非常に広い室内の一区画で，横 6.0m，奥行き 3.2m の範囲において実験を行った．シーン 5 は細かな形状変化が多く存在しているシーンである．

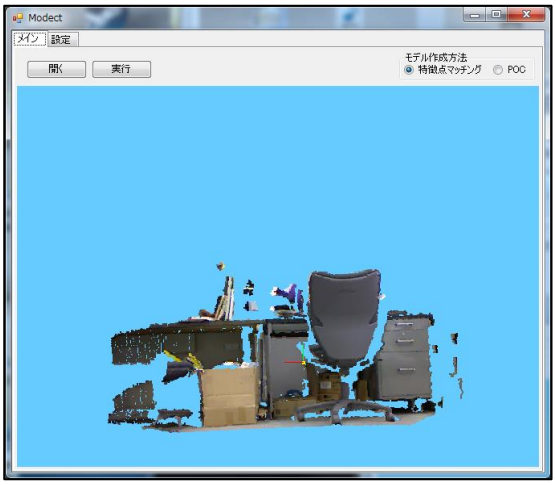
(a)：時刻 t+2　　(b)：時刻 t+12

(c)：時刻 t+22　　(d)：時刻 t+32

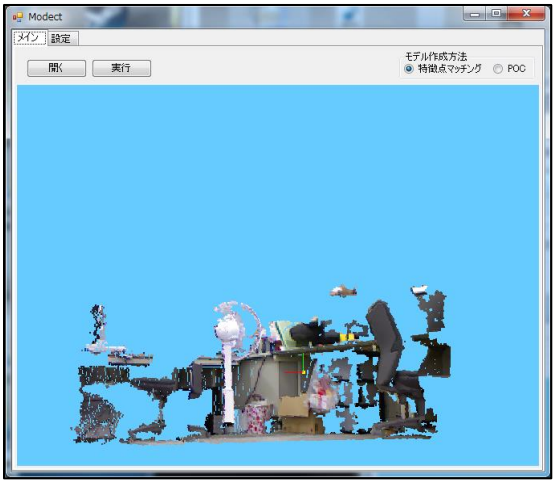
図 5.17: Kinect センサから取得した 2 次元画像



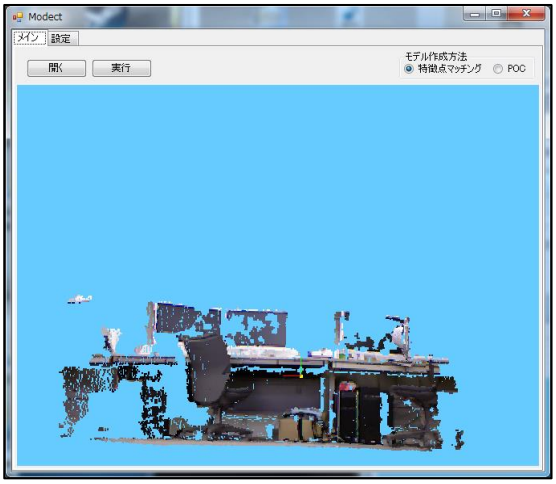
(a): 時刻 t+2



(b): 時刻 t+12



(c): 時刻 t+22



(d): 時刻 t+32

図 5.18: Kinect センサから取得した 3 次元情報

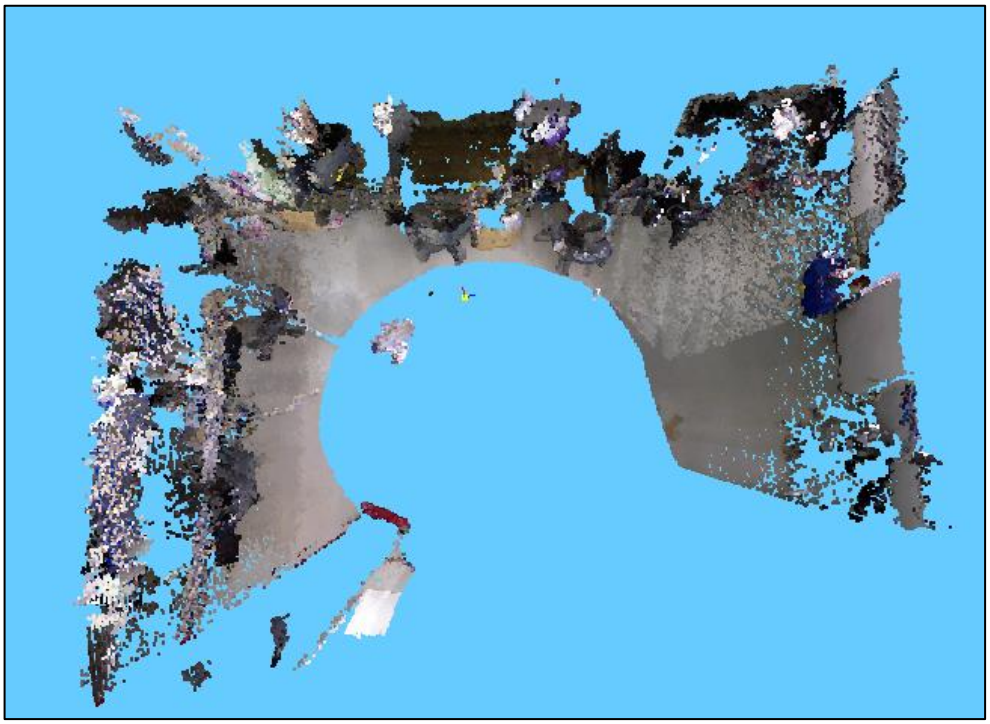
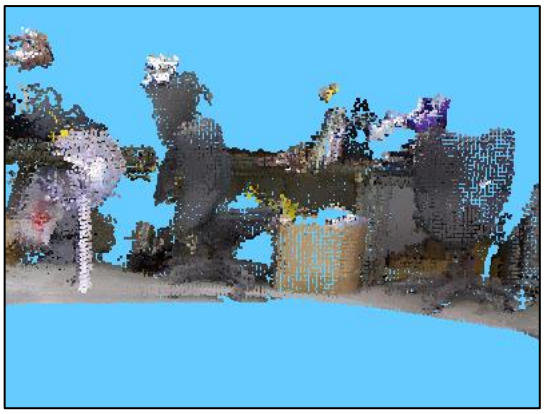
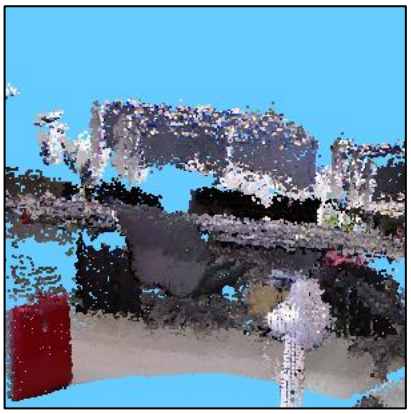


図 5.19: 3 次元モデルの作成結果





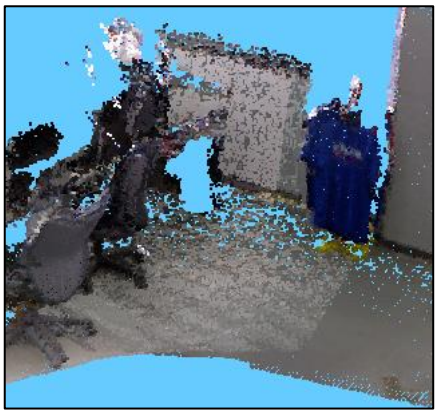


図 5.20: 3 次元モデルの拡大結果

表 4: 作成したモデルの大きさの推定結果

	実測値[m]	推定値[m]	誤差[%]
奥行	3.20	3.34	-4.4
幅	6.00	6.01	-0.2

5.2　考察

[シーン 1]

シーン 1 はカメラを水平方向に平行移動させたシーンであるため，回転変化などもほとんどなく，また段ボールなどから特徴を取得できたため，実際の壁の大きさとモデルの大きさを比較したところ，実際の壁のある特定の領域の長さが 1.7m であるのに対し，モデルの同一領域の大きさが 1.65m となり，誤差 3%の精度の高いモデルを作成することができた.

[シーン 2]

シーン 2 はシーン 1 と違いカメラを水平方向に回転させて撮影したシーンである．しかし，回転変化に強い特徴量を用いたため，正しく対応付けが行えており，回転変化したデータに対しても正しく 3 次元モデルを作成することができた.

[シーン 3]

シーン 3 はシーン全体を通してテクスチャ特徴量が多く存在したため，精度の高いモデルを作成することができた．また，Kinect によるデータ取得時の誤差をアフィン変換により軽減することができ，データを正しく重ね合わせることができた. 作成されたモデルも横幅・奥行きともに実寸と近い値（誤差約 2%）を示している．しかし，データの最後の部分がアフィン変換で補正しきれておらず，データが正しく繋がり切っていない.

[シーン 4]

シーン 4 はデータの前半部分は特徴が多く，安定して重ね合わせることができた，しかし，データの後半部分では壁や TV の液晶画面がデータ全体に広く映

っており，特徴を得ることが困難であったが，複数次元の特徴量を組み合わせることで，3 次元モデルを作成することができた．また，アフィン変換によりデータの重ね合わせの誤差を軽減でき，3 次元モデルを作成することができた．

[シーン 5]

シーン 5 はデータに細かな形状変化が多い，また PC の液晶画面などからデータが得られないなどの問題があったが，精度の高いモデル作成が行えている．しかし，作成されたモデルの横幅と比べて奥行での誤差が大きいが，これはシーン 4 と同様にシーンの後半部分で特徴が少なく正しく対応付けが行われなかったことが原因であると考えられる．また，全てのシーンに対して同様であるが，現在 3 次元特徴点の検出時のボクセルフィルタのスケールを，同一のデータでは全ての点に対して同一のスケールを使用している．しかし，データの手前側と奥側では点の密度が違うため，同一のスケールで行うと細かな形状変化が捉えられない，点間の距離が離れすぎていて適切でない特徴点の検出を行ってしまうなどの問題があるため，データの形状や点群の密度変化に応じたスケールの計算方法を考える必要がある．

第 6 章

結論

本研究では，Kinect から得られる 2 次元画像と 3 次元情報を組み合わせることで，手軽に，多様なシーンに対応した 3 次元モデルを作成する手法の提案を行った．まず，第 1 章では，序論として現在の 3 次元モデルに対する需要や利便性に対して, 3 次元モデル作成が困難であることが問題として存在することを述べ，簡易で高精度な 3 次元モデル作成手法の必要性について述べた．また，現在の 3 次元データ間の重ね合わせ手法の問題点を述べることで，本手法を行う理由を明確にした．第 2 章と第 3 章では，各次元のデータに対して，本研究で用いた特徴量の説明を行い，Kinect センサから取得した画像と 3 次元情報から特徴点や特徴量の取得方法について述べた．第 4 章では，重ね合わせ精度を向上させるために，データ間の対応付け結果から誤マッチングしたものを除去する方法，データを前半部分と後半部分で分割してモデルを作成し，それらを重ね合わせることでデータの補正を行う理由について述べた．第 5 章では，本研究の有効性を検証するために複数のシーンにおいて実験を行い，実験結果の考察を行った. 実験結果より, 2 次元画像を用いることで, 3 次元情報のみでは，困難な状況でも安定して 3 次元モデルを作成することができることを確認した.

現在の問題として, 3 次元センサから取得した情報のノイズや計測誤差が多いため，3 次元特徴点検出や 3 次元特徴量の対応付け精度が低く，3 次元モデル作成精度の大部分を 2 次元特徴量に依存しているため，テクスチャ変化が少ないシーンでは精度が大きく落ちてしまう．また，2 次元情報に対する処理と 3 次元特徴点検出は高速に行えているが, 3 次元特徴量の算出に時間が掛かる問題や考

察でも述べたように点群の密度変化に対応してスケールの変更を行う必要がある．

今後の課題として，室内のシーンでは類似のテクスチャやテクスチャ変化の少ない壁などを含むシーンに対する精度向上，3 次元特徴量の見直し，実験シーン追加による精度検証が挙げられる．

参考文献

1. 総務省：“ドローンの現状について”，http://www.soumu.go.jp/main_content/000401647.pdf, (2016-2-25).
2. 株式会社三菱総合研究所：“VR/AR 技術の将来展望”，http://www.mri.co.jp/opinion/column/uploadfiles/tec_10.pdf, (2016-5).
3. Freelancer.com：“Freelancer.com reveals the Fast 50 small business trends driving online employment for Q1 2014”，http://press.freelancer.com.s3.amazonaws.com/20140414 Freelancer.com reveals the Fast 50 small business trends driving online employment for Q1 2014.pdf, (2014).
4. 中谷友樹:「3 次元でみる京都の景観と災害」, 立命館大学・神奈川大学 21 世紀 COE プログラム ジョイントワークショップ「歴史災害と都市－京都・東京を中心に－」, (2006-8-26).
5. Paul J. Besl, Neil D. McKay : “A Method for Registration of 3-D Shapes”, IEEE Trans. on PAMI, Vol. 14, No. 2, pp.239-256, (1992).
6. 羽田 遼, 寺田 賢治:“3 次元回転不変位相限定相関法を用いた 3 次元モデリング”，電気学会論文誌 C (電子，情報，システム部門誌), Vol.136, No.8, pp.1128-1134, (2016).
7. E.Rosten, T.Drummond：“Machine learning for high-speed corner detection”, European Conference on Computer Vision (ECCV), pp. 430-443, (2006).

8. S.Leutenegger, M.chli, R.Y.Siegwart：“BRISK：Binary Robust Invariant Scalable Keypoints”, Proceedings of International Conference on Computer Vision (ICCV), pp.2548-2555, (2011).
9. 武井翔一, 秋月秀一, 橋本学：“3 次元点群の空間的広がりに着目した高速キーポイント検出”, SSII2015, No. IS1-18-1-5, (2015).
10. R.B.Rusu, N.Blodow, M.Beetz：“Fast Point Feature Histograms (FPFH) for 3D Registration”, ICRA2009, pp.1848-1853, (2009).
11. M.A. Fischler and R.C. Bolles：“Random Sample Consensus: A Paradigm for Model Fitting with Apphcatlons to Image Analysis and Automated Cartography”, Comm. ACM, Vol. 24, No. 6, pp.381-395, (1981).
12. David G. Lowe：“Distinctive image features from scale-invariant keypoints”, International Journal of Computer Vision, 60, 2, pp. 91-110, (2004).
13. R.B.Rusu, N.Blodow, Z.C.Marton, M.Beetz：“Aligning Point Cloud Views using Persistent Feature Histograms”, IEEE Proc. IROS, pp.3384-3391, (2008).

謝辞

本研究を行うにあたり，常時懇切丁寧に御指導，御教授下さった寺田賢治教授に深く感謝の意を表します.

また，多方面にわたり御助言，御指導頂きましたStephen Karungaru講師，辻明典技官に深く感謝申し上げます.

そして，様々な事にご協力して下さった研究室の皆さんに心より感謝いたします.