他視点のオブジェクトを検索可能な画像検索システムの構築

雛形　奎祐† 　欅　 惇志†,†† 　宮崎　 純†

† 東京工業大学情報理工学院情報工学系　〒 152–8550 東京都目黒区大岡山 2–12–1
†† 国立研究開発法人科学技術振興機構，ACT-I　〒 332–0012 埼玉県川口市本町 4-1-8
E-mail: †{hinagata,keyaki}@lsc.cs.titech.ac.jp, ††miyazaki@cs.titech.ac.jp

あらまし　本研究では，画像を入力として，その画像中のオブジェクトをユーザーによって指定された方向から見た画像を提示する検索システムの構築方法を提案する．従来の画像検索システムでは，画像からさまざまな特徴量を抽出し，その特徴量を用いて画像同士の類似度を計算して，より類似度が高いものから順に検索結果を提示している．しかしこの性質上，従来の画像検索システムは，あるオブジェクトを一方から写した画像を入力として，そのオブジェクトを他の特定の方向から見た画像だけを検索することが難しい．そこで本研究では，オブジェクトの 3D 情報を利用し，画像中のオブジェクトに対応する 3D モデルの射影を考えることで，入力画像中のオブジェクトをある特定の視点から見た画像を提示するシステムを構築する方法を提案する．また，その提案手法を評価実験により検証する．

キーワード　情報検索，画像検索，3D モデル，モデルベーストラッキング

1. はじめに

近年，ウェブ上には膨大な量の情報があふれ，ユーザーがそれらの情報を利用するためには適切な検索システムを使って情報を取捨選択する必要がある．加えてユーザーのニーズも多様化しており，検索対象はテキストだけではなく画像や音声，動画などのマルチメディア情報にまで広がっている．これらのマルチメディア情報を検索対象とする場合，検索要求をキーワードでは表現しにくいということも多々ある．

そのような場合に用いられる検索システムの中に，画像をクエリとして類似した画像を提示する画像検索システムがある．画像をクエリとして類似した画像を提示する画像検索システムは，そのアルゴリズムが盛んに研究されている [11]．既存の画像検索のシステムは，クエリ画像と検索対象画像それぞれから，色味や勾配などさまざまな特徴量を抽出し，それらを比較することで画像の類似度を計算・結果の出力を行う．このようなシステムは，図 1 のようにクエリと同じもの，あるいは似たものを被写体とした画像を検索することを目的に据えて研究が進められている．だがこのようなシステムは，図 2 のように与えられたクエリ画像内のオブジェクトを他の特定の視点から見た画像をピンポイントで探してくるという用途には使うことは難しい．解決方法の一つとして，視点情報をテキストとしてクエリ条件に含めることも考えられるものの，必ずしも画像に視点情報が付与されているとは限らず，また，視点情報を言語にすることも必ずしも容易とは限らない．そこで本研究では，この問題の解決を目指す．

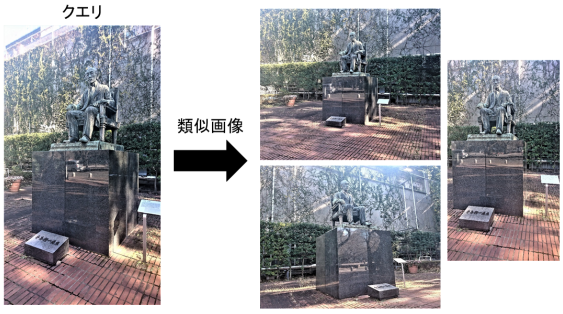

図 1　既存手法で可能な検索

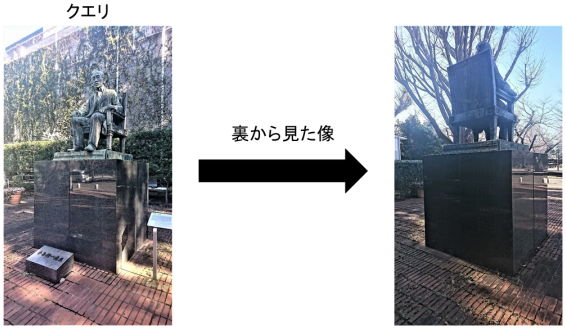

図 2　既存手法では想定されていない検索

画像中の物体を他の角度から見た画像を探すためには，その物体自体の形状の情報，すなわちその物体の 3D モデルを利用する．本研究では，クエリに用いる画像を物体までの距離情報を持った画像 (デプス画像，以下では D 画像と呼称) とし，そこから取り出した 3D 情報を基に画像中の被写体に対応する 3D モデルを探し，その 3D モデルを利用してユーザーが物体をどの方向から見た画像を検索するのかを指定したのち，物体を指定された方向から見た画像を検索するという方法で，先述した問題の解決を目指す．D 画像は，二つのレンズの視差によって対象物までの距離を測るデプスカメラや，Microsoft Kinect(注1) に代表される対象物とレンズとの距離を測るデプスセンサーなどから生成できる．近年ではこれらの技術が発展しており，一般社会にも普及し始めているため，本研究でもこれを用いることとする．画像から深さ情報を取り出す手法は，たとえば画像

(注1)：https://developer.microsoft.com/ja-jp/windows/kinect/hardware

に対して CNN(Convolutional Neural Network) による深さ推定を行う手法 [13] など他の手法も研究されている．しかし事前の学習が必要などというディスアドバンテージも大きいため，本研究では使用しない．

2. 関 連 研 究

画像の情報を基に似た画像の検索を行う手法はすでに数多く提案されている．Liu らの調査 [11] では，画像中のオブジェクトが持つ特徴量として色，テクスチャ，形，画像中における位置が挙げられている．このような具体的な特徴量のほかにも，画像の回転や明度、拡大縮小に対して頑強性を持たせた SIFT(Scale-Invariant Feature Transform) [9] や，画像中にどのような特徴をどれだけ持っているかをヒストグラムで表す Bag of Visual Words(BoVW) [10] など抽象的な特徴量も存在する．これらの特徴量を使って画像同士の類似度を計測することが，先述した通り画像検索の基本的な発想である．加えて，Liu らの調査 [11] では，検索の精度を向上させることを目的とした，オントロジーや教師データを利用した学習，教師なし学習を利用した手法について触れられている．そのほかにも，フィードバックを利用して学習を進める手法 [8] などが研究されている．

学習を利用した手法で組み立てられたシステムは，特にシステムが学習済みのオブジェクトの画像がクエリである場合に，同一オブジェクトを写した画像を返すことができる場合が多い [8]．また，Fujiwara らの研究 [14] では，マニフォールドランキングと呼ばれるグラフベースのアルゴリズムを用いて，クエリ画像中のオブジェクトと同一のオブジェクトを写した画像の検索に成功している．しかし，このような手法をもってしても，物体を特定の方向から見た画像を検索するという目的は達成することが難しい．その目的を達成することが，本研究の目標である．

3. 提 案 手 法

本研究では「画像中のオブジェクトをユーザーが指定する方向から見た画像を検索する検索システム」の構築手法を提案する．ここでは，クエリ画像中の 3D 情報を基に，画像中の被写体に対応する 3D モデルを探し，その 3D モデルを使ってオブジェクトを他の視点から見た画像を検索することを考える．本研究ではクエリから容易に 3D 情報を抽出できるようにするため，クエリに用いる画像を物体までの距離情報を持った画像，すなわち D 画像とする．

以下，本章では提案するシステムのアーキテクチャについて概説し，その後各構成部分について詳細を述べていく．

3.1 システムのアーキテクチャ

ここでは，本研究で提案するシステムのアーキテクチャについて包括的に述べる．

本研究で構築を目指すシステムの全体図が図 3 である．このシステムのフローは，大きく以下の四つに分けられる．

（1） 画像から 3D 情報を抽出．
（2） クエリに対応する 3D モデルの検索．

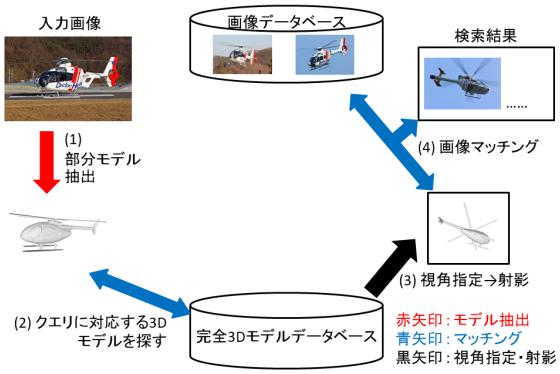

図 3 システムの全体図

（3） 視角を指定，モデルの射影の取得．
（4） 画像マッチング．

まずは入力された D 画像を基に，その画像に写っている範囲のオブジェクトのモデル (以下では部分モデルと呼称) を書き起こす (1)．たとえば，図 3 のようにヘリコプターを左側から写した D 画像からは，ヘリコプターの左側だけの 3D 情報を持った部分モデルが生成できる．

次に生成した部分モデルから，その部分モデルを一部として含むオブジェクトのモデル (以下では完全モデルと呼称) を検索する (2)．検索は，部分モデルの形状の特徴とデータベース中の 3D モデルの形状の特徴を比較することで行う．部分モデルに対応する完全モデルが発見された場合 (3) に進む．そのためには部分モデルに対応する完全モデルを確実に検索できる手法が必要となるが，そのような検索手法は知られていない．したがって，システムが部分モデルと一番類似度が高いモデルとして部分モデルに対応しない完全モデルを最上位に提示してしまった場合，以降のフローではその間違ったモデルを使用して検索を続けることとなってしまい，ユーザーの検索要求に正しく応えることができない．それを防ぐため，部分モデルをクエリとして完全モデルを検索した際の検索結果はユーザーに提示し，ユーザーが正解の完全モデルを選択する必要がある．

ユーザーによって正解の完全モデルが選択された後は，その完全モデル (オブジェクト) をどこから見た画像を検索するのかをユーザーが選択する．つまり，ユーザーが選択した完全モデルをウィンドウ上に描画し，ユーザーの操作によってモデルを回転させる．ユーザーはオブジェクトをどの角度から見た画像を探したいかを指定する (3)．

最後に，得た完全モデルとその視点を基に画像マッチングを行う (4)．ここでは，完全モデルを指定された視点から見た場合に得られる射影と，画像中のオブジェクトが描く輪郭線を比較して，その差異が小さいものを適合度が高い画像として判断する．

本研究で構築するシステムの動作イメージを図 4 に示す．入力画像は最も左側にある 2 枚の画像のうち下の画像である (わかりやすさのため，明度を加工している)．上の画像は，D 画像に対応する RGB 画像である．この D 画像を検索システムに入力すると，システムは部分モデルの抽出と 3D モデルマッチ

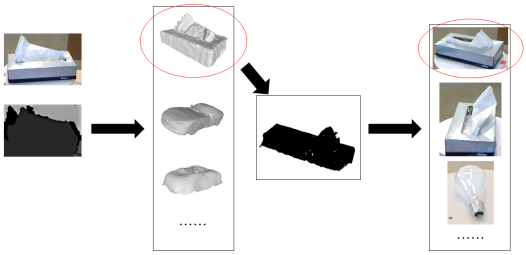

図 4　動作イメージ

ングを行い結果を提示する (中央左側の長方形内)．該当するモデルをユーザーが選択すると，システムはそのモデルを別ウィンドウに描画し，ユーザーが検索する視点を決定する．視点が決定 (中央右側の長方形内) したら，システムはモデルとその姿勢の情報を基に画像の検索を行う．その結果が最も右側の長方形内に示されている．

各プロセスについての詳細については，以下で述べる．

3.2　部分モデルの抽出

画像中のオブジェクトに対応する 3D モデルを探すために，まずはその前段階として入力された D 画像から画像中に格納されている 3D 情報 (部分モデル) を抽出する (図 3，(1))．

デプスカメラやデプスセンサーによって書き出される D 画像は，そのピクセルの値一つ一つがセンサーによって取得された距離に対応している．したがって，D 画像中のそれぞれピクセルについて値と座標を統合すれば，画像に写った範囲のオブジェクトの部分モデルを点の集まりである点群形式 [1] で再構築できる．こうして生成した点群形式の部分モデルについて，まずは点群中の点の並びを調整する必要がある．D 画像から生成したモデルは，ノイズを含んでいる影響で凹凸が実物以上に表現されていることが多い．そのため点群中の点の並びを調整し，ノイズの影響をできるだけ除去しておく必要がある．点群中の点の並びを調整した後に近い点同士を結び，メッシュを再構成する (メッシュ化) ことで部分モデルの抽出は完了とする．

3.3　3D モデルマッチング

部分モデルをクエリとして完全な 3D モデルを検索する機構は，Liu らの手法 [2] を参考にして構成した．この手法はモデル上のいくつかの点から，それぞれ周辺に存在する頂点の分布をとり，その分布をクラスタリングすることによって特徴量を生成する．すなわち，この手法におけるモデルの特徴量とは，そのモデル上に「どのような形状がどの程度存在するか」ということに相当する．計算はデータベース内の 3D モデルの集合に対して特徴量を事前計算しておくオフライン処理と，クエリが来たときに事前計算した特徴量を用いてデータベース内の 3D モデルとの類似度を計算するオンライン処理に分かれる．

まず，オフライン処理は以下のように行う．

（1）　3D モデルを構成する頂点の数の調整．

3D モデルの頂点の数を N 個，モデルの表面積を S，モデル重心からすべての頂点までの距離の二乗平均平方根 (RMS) を R としたときに，

$$N = k \times \frac{S}{R^2}$$

となるように頂点の数を調整する．ここで k は定数であり，本研究では先行研究 [2] を基に $k = 3019$ で固定とした．頂点数が求めた N よりも少ない場合はメッシュ分割により頂点を増やし，多い場合はメッシュを統合することによって減らす．

（2）　モデル上の点を選び，周辺の頂点の分布をとる．

モデルを構成するメッシュ上のある点を中心とし，メッシュの水平方向を r 軸，メッシュの垂直方向を z 軸とする円柱座標を定義する．次に，円柱座標で表されたモデル上の各頂点について，領域 $r < 0.4R, |z| < 0.4R$ [2] に含まれるものを $w = r, h = |z|$ となるように二次元平面上の点 (w, h) に射影する．こうして生成した wh 平面上の $0.4R \times 0.4R$ の正方形領域を，それぞれの軸に平行に 15 分割 [3] する．格子に含まれる点の最大数で各格子の点の数を割って，$225 (= 15 \times 15)$ 次元のベクトルを生成する．このベクトルはスピンイメージと呼ばれる．

モデル上から 500 個 [2] の点をランダムに選んでこの計算を行い，スピンイメージを 500 個生成する．

（3）　検索対象となるすべてのモデルについて (1)(2) を行う．

各モデルについて，500 個ずつスピンイメージを生成する．

（4）　ベクトルのクラスタリング．

抽出したすべてのスピンイメージを，k 平均法を用いて 1500 個 [3] のクラスタへクラスタリングする．このとき，ベクトル同士の距離計算には L2 ノルムを用いる．

（5）　ヒストグラムの生成．

各モデルについて，どのクラスタに属するスピンイメージが何個生成されたかをヒストグラムとして保存する．このヒストグラムが完全モデルから抽出された特徴量となる．また，これとは別に各クラスタの中心を表すベクトルも保存しておく．

オフライン処理で事前計算した特徴量とクラスタを用いて，クエリが入力されたときの処理 (オンライン処理) を行う．オフライン処理と同様にクエリのモデルに対して 500 個のスピンイメージを生成した後，それらの分布を事前計算したクラスタを用いてヒストグラムで表す．

クエリの部分モデル Q からモデル P までの距離は，それぞれから抽出されたヒストグラムと一様分布 I を用いて以下の式で定義される．

$$D = KL((1-\varepsilon)Q + \varepsilon I || (1-\varepsilon)P + \varepsilon I)$$

ただし，ε は定数であり，$KL(A||B)$ はカルバック・ライブラー情報量である．距離 D を検索対象であるすべてのモデルに対して計算し，距離が昇順となるようにソートして検索終了とする．

3.4　視点の指定

オブジェクトをユーザーが要求する視点から写した画像を検索するには，3D モデルを利用してユーザーが要求する視点をユーザー自身が指定する必要がある．そのため，クエリに対応する完全モデルが検索できたら，そのモデルを使ってユーザーが画像を検索する際の視点を指定する (図 3，(3))．ユーザーによる視点の指定は，視点を固定してウィンドウ上に描画されたモデルを回転させることにより擬似的に行う．

ユーザーが検索に使用する視点を指定したら，システムはその時点のモデルの姿勢 (各軸を中心とした回転角) を保存し，描

画していたモデルとともに次の機構へと渡す．

3.5 画像の検索

検索に用いるモデルとその姿勢が得られたら，それらを使ってオブジェクトをユーザーが指定する視点から見た画像を検索する (図 3，(4))．この検索の要件は，3D モデルを与えられた姿勢で射影した際の輪郭が画像中の物体の輪郭と似ていれば，ユーザーが指定した視点からオブジェクトを見た画像としてその画像を上位に表示することである．すなわち，画像中の物体の輪郭とその上に射影した 3D モデルの輪郭線との誤差をとり，誤差が小さければその画像は 3D モデルに対応する物体を与えられた視点から見た画像とみなす．しかしこの誤差を計算するときには，単純な直線同士の比較とは違い，モデルを射影した際のモデルの辺同士がなす角といったモデルの構造も考慮する必要があるため，計算が煩雑なものとなってしまう．

そこで本研究では，視覚サーボのプラットフォームである ViSP (Visual Servoing Platform) [4] のモデルベーストラッキング機構を用いて，先述した計算を代用することを考える．トラッカーにより画像上に検出されたエッジと射影された 3D モデルの輪郭線との誤差を用いれば，先述した誤差と近い値が計算できる．加えてトラッカーを用いることにより比較的平易にモデルの構造を崩すことなく二つの輪郭を比較することができる．しかし，本来モデルベーストラッキングは決まったカメラで撮影された既知の位置にある画像や動画中の物体を追跡するための機構なので，そのカメラの情報や物体の位置情報が未知である本研究においては，これらの情報を推定する必要がある．また ViSP のトラッカーは 3D モデルを射影した際に見えているすべての 3D モデル上の辺を画像中から検出したエッジとの誤差計算に利用するため，射影した際に輪郭線上に位置しない辺をトラッキングに使用する 3D モデルからあらかじめ除去しておく必要がある．

3.5.1 モデルベーストラッキング

モデルベーストラッキングとは，3D モデルとして与えられた物体が，二次元画像中のどこにあるのかを検出，追跡する手法 [5] のことである．ViSP のトラッカーは，与えられた 3D モデルを平面上に透視投影し，そのモデルが作る輪郭と，追跡対象の画像や動画中から検出されるエッジの差が最小となるようにモデルの射影を動かすことで物体の検出，追跡を行うように実装されている．図 5 は，実際に ViSP のトラッカーに直方体のモデルを与え，画像中の直方体の物体を検出している例である．図中の赤い直線が射影されたモデルの輪郭，画像上の緑色や赤色の点が画像から検出されたエッジである．

ViSP のトラッカーは，画像中の物体を検知するときに画像中のエッジを検出した後，3D モデルを画像上に射影したときに，各エッジが与えられたオブジェクトの輪郭上の点であることが妥当であるかを計算する．検出されたエッジのうち，妥当であると判断されたエッジの割合が与えられた閾値以上ならば物体の検知を成功とみなす．閾値が高いほど，モデルの輪郭と画像中のエッジの差異が小さくないと検知が成功しない．図 5 においては，緑色の点が妥当とみなされた点，赤色の点が妥当とはみなされなかった点である．

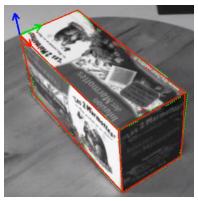

図 5　ViSP を用いたトラッキングの例 (文献 [4] から引用)

3.5.2 実装上の課題と解決法

トラッカーが正確に画像中の物体を検出するためには，3D モデルとその姿勢，先述した閾値のほかに，画像を撮影したカメラの情報 (カメラパラメータ) や物体の位置情報が必要となる．ここで，カメラパラメータとは 3D モデルを射影した像の大きさを決定する値，および画像の中心座標から成る．3D モデルを射影した際の像の大きさはこのカメラパラメータと，カメラレンズと 3D モデルとの距離 (モデル上の各 z 座標の値) に依存する．したがって，画像が与えられた 3D モデルに対応する物体を与えられた視点から撮影したものであった場合，3D モデルとカメラレンズとの距離を固定し，適当なカメラパラメータを与えて 3D モデルを射影した像は，画像中の物体と相似なものになる．そこで，本研究では物体の中心から z 軸方向負の向きに一番離れた点の z 座標を Z_{min}，モデル重心からすべての頂点までの距離の二乗平均平方根 (RMS) を R としたとき，この点とカメラレンズとの距離が

$$|Z_{min}| \times R \times n$$

となる場合の射影を考え，それぞれについてカメラの情報を推定することとした．画像中の物体の検出は，画像中の物体の輪郭が画像中から検出できる最長輪郭であることを仮定して行った．n の値についてはいくつかの 3D モデルについて実際に ViSP による投影を確認し，本研究では $n = 3, 4, 5$ のときの射影をとることとした．なお，物体が十分遠くに存在する場合には，得られる射影は平行投影に近いものとなる．そのため上記の 3 通りの射影に加えて，3D モデルを z 軸方向の十分遠くに配置した場合の射影を考えれば，カメラパラメータが未知である画像に対してトラッカーを用いた距離計測が行える．さらにカメラパラメータの推定と同様，物体の位置は 3D モデルを射影した際の像と画像中の物体の位置を比べることで推定できる．

また，3.5 節でも述べたが，ViSP のトラッカーは 3D モデルを射影した際に見えているすべての 3D モデル上の辺を画像中から検出したエッジとの誤差計算に利用する．しかし，その状態では画像中の物体の輪郭上から検出されたエッジに対して，モデル上の輪郭ではない辺がたまたま近くに射影されたときに，それらの辺と画像中のエッジがマッチすると判断されてしまうことが起こりえる．したがって，モデルが作る射影の輪郭部分のみを本研究では用いたい．そのため，推定したカメラパラメータや物体の位置情報を基に 3D モデルを平面上に射影した際の輪郭の情報を利用して，輪郭部分だけの情報を持つ 3D モデルを作り ViSP に渡し直すことを考えた．こうして生成した輪郭部分だけの情報を持つモデルを用いることで，3D モデ

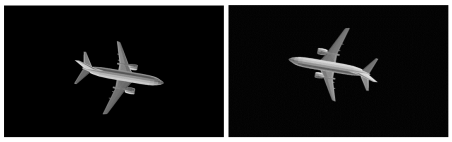

図 6　同一視点にもかかわらずエッジ差が変わってしまう画像の例

ルの輪郭と画像から検出されるエッジとの差を計算することができる．

3.5.3　画像との距離計算

前節までで 3D モデルの輪郭と画像から検出されるエッジとの差を計算する方法を述べたが，こうして得た誤差をそのまま画像への距離とすることによる問題も存在する．というのも，例えば図 6 のように同一視点 (画像自体が 180° 回転しているだけ) から物体を見た場合でもそれぞれの画像への距離が異なるものとなってしまう．また，ちょうど同じ姿勢の画像が得られない場合でも，近い姿勢の画像があればそれを上位に表示したい．これら問題を解消するために，本研究では 3D モデルの姿勢を x 軸，y 軸，z 軸を軸に少しずつ回転し，物体がすべての方向に一周するまでここまでの計算を繰り返すことで解決を図った．本研究では，各軸まわり 30° 刻みに動かした．

最後に，得られた輪郭との差異の中で最小のものを画像への距離とする．なお，物体が一周する間に一度も妥当と見なせる射影が得られなかった場合，その画像への距離は無限遠として検索結果には含めないものとする．

4. 評価実験

4.1 実験方法

本研究では，評価実験として，3.3 節で述べた 3D モデルマッチング機構，および 3.5 節で述べた画像の検索機構について適合率に基づいた評価を行った．なお，ここで述べるすべての実験は，16.0 GB のメインメモリと Intel(R) Core(TM) i7-6700 3.4 GHz の CPU を搭載したマシン上で動作する 64 bit の Windows10 上で行った．

4.1.1　データセット

実験で用いたデータセットは，Microsoft Kinect で撮影された RGB 画像と D 画像の組のデータセットである RGB-D Object Dataset [6] と，3D モデルのデータセットである PSB(Princeton Shape Benchmark) [7] から生成している．以下では, [6] を RGB-D データセット, [7] を PSB データセットと呼称する．RGB-D データセットは 51 種類にカテゴライズされた 300 個のオブジェクトを，Kinect センサーを用いてさまざまな方向から撮影した際の RGB 画像と D 画像から成るデータセットである．また PSB データセットは 161 種類にクラスタリングされた 1814 個の 3D モデルから成るデータセットである．

4.1.2　3D モデルマッチングの評価実験

部分モデルをクエリとし，その部分モデルを一部とする完全モデルを含んだ 3D モデルファイルの集合に対して検索を実行した．検索対象の 3D モデルファイルの集合は，RGB-D データセットの D 画像を結合して生成した 6 種類のモデル (図 7) と，PSB データセットの内のモデルのうち各クラスタから一つずつから取得した 161 種類 (図 8 のものを含む) のモデルの計 167 種類のモデルから成る． 検索クエリに用いたのは，図 7 に含まれるオブジェクトを写した D 画像から生成した部分モデル，および PSB データセットから取得した 144 種類 (いずれも先述した検索対象に含まれる) のモデルの一部を切り取って生成した部分モデルである．各モデルに対応する部分モデルは 4 種類ずつ存在する．各クエリ (部分モデル) に対して，適合するものはその部分モデルに対応する完全モデルのみとし，距離計測に使う ε の値を変えながら，各クエリを入力としたときの出力の上位 10 件に対する AP を計測し，MAP を計算した．また，出力上位 k($= 1, 2, \cdots, 5$) 件に対する P@k も計測した．

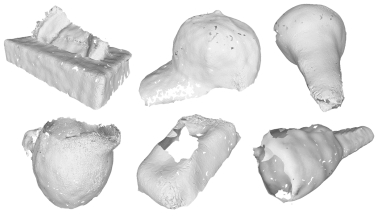

図 7　RGB-D データセットから生成した 3D モデル

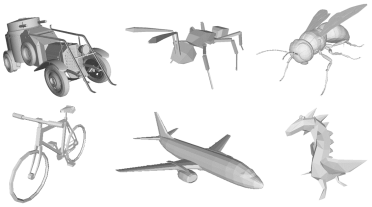

図 8　PSB データセット内のモデルの例

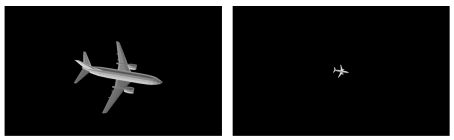

図 9　近くの被写体を写した画像と遠くの被写体を写した画像

なお手法に乱数を使用するため，計測する AP および P@k は，オフライン処理である完全モデルからの特徴量抽出を 5 回行い，それぞれについて 600(= $(6 + 144) \times 4$) 種類のクエリを 15 回ずつ渡した際の平均である．

4.1.3　画像の検索の評価実験

完全な 3D モデルと姿勢をクエリとし，与えられたモデルを与えられた姿勢から見た画像を含む画像ファイルの集合に対して検索を実行した．検索対象は PSB データベースから取得した，図 8 に示す 6 種類のモデルを，それぞれ 4 種類の方向から，モデルを 2 通りの距離 (図 9) から写した計 48 枚の画像群とした．クエリとして与えるのは上述した 6 種類のモデルと，そのモデルを撮影した視点 (モデルの回転角) である．モデルの種類と視点の組をを 1 組と数えれば，クエリの各組に対応する画像が 2 枚画像群に含まれることになる．この 2 枚のみを適合する画像としたとき，トラッカーに渡す閾値の値を変えなが

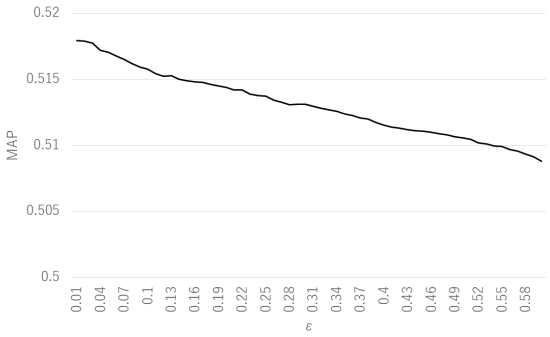

図 10　3D モデルマッチングの実験結果 (MAP)

表 1　3D モデルマッチングの実験結果 (P@k)

ε	0.01	0.06	0.11	0.16	0.21	0.26	0.31	0.36
P@1	0.444	0.441	0.439	0.438	0.437	0.436	0.436	0.435
P@2	0.259	0.260	0.259	0.259	0.259	0.259	0.259	0.259
P@3	0.187	0.187	0.186	0.186	0.186	0.186	0.186	0.186
P@4	0.149	0.149	0.149	0.149	0.149	0.149	0.148	0.148
P@5	0.124	0.124	0.124	0.124	0.124	0.124	0.124	0.123

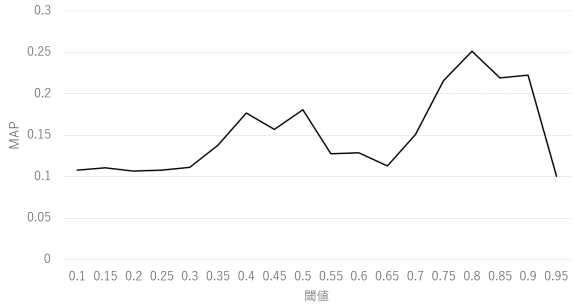

図 11　画像の検索機構の実験結果 (MAP)

ら各クエリの組を入力したときの出力に対する AP を計測し，MAP を計算した．また，出力上位 k($= 1, 2, \cdots, 5$) 件に対する P@k も計測した．なお時間短縮のため，輪郭モデルを生成するときに用いる輪郭線上の点は最大でも 250 個までとした．

4.2　実験結果

3D モデルマッチングの機構について 4.1.2 節で述べた条件の下で ε の値を変え MAP を計測した結果，図 10 のような結果を示した．先行研究 [2] では $\varepsilon = 0.13$ としていたが，本研究では $\varepsilon = 0.01$ のときに MAP が最大の 0.518 となった．また，ε の値を変え P@k を計測した結果は表 1 の通りである．

一方，画像の検索機構について 4.1.3 節で述べた条件の下で閾値を変え MAP を計測した結果，図 11 のような結果を示した．本研究ではトラッカーに渡す閾値が 0.8 のときに MAP が最大の 0.252 となった．また，閾値を変え P@k を計測した結果は表 2 の通りである．

4.3　考察と課題

3D モデルマッチング機構に関しては，用いた手法 [2] の得手不得手がはっきりと出た結果となった．$\varepsilon = 0.01$ のときに MAP が最大で 0.518 となったのは先述の通りだが，図 7 の左上にある箱ティッシュのオブジェクトの部分モデルをクエリとしたときの AP の平均は 0.871，同じく図 7 の右下にある懐中

表 2　画像の検索機構の実験結果 (P@k)

閾値	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
P@1	0.042	0.000	0.167	0.208	0.083	0.083	0.250	0.292
P@2	0.042	0.063	0.083	0.104	0.083	0.104	0.250	0.271
P@3	0.042	0.083	0.083	0.083	0.069	0.139	0.236	0.194
P@4	0.042	0.063	0.073	0.094	0.073	0.104	0.208	0.135
P@5	0.033	0.058	0.083	0.083	0.058	0.117	0.175	0.092

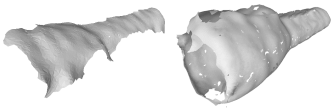

図 12　検索精度の低いモデルの例

電灯のオブジェクトの部分モデルをクエリとしたときの AP の平均は 0.015 であった．加えて 3D モデルが一つだけであることから，表 1 中の P@1∼P@5 はどの ε についても k にほぼ反比例する形になったと考えられる．

原因としては，部分モデルや対応する完全モデルに形状の特徴が少ないものがあったことが挙げられる．図 12 に AP が低かった懐中電灯の部分モデルと完全モデルの例を示す．見てみると，特に部分モデルは形状に凹凸も少なく，平坦な面であることが分かる．さらに，本手法では距離計測にカルバック・ライブラー情報量を用いている．これによってクエリに含まれる形状が検索対象に含まれていない場合に，クエリから検索対象までの距離を大きくすることに成功しているが，裏を返せばクエリがありふれた形状しか持たない場合，さまざまなオブジェクトまでの距離が短くなることにもつながっている．

この問題の解決策としては，関連研究で述べたマニフォールドランキングを用いた手法 [14] の利用が考えられる．前述した通り，Fujiwara らの研究 [14] ではクエリ画像中のオブジェクトと同一のオブジェクトを写した画像を検索することに成功している．これを利用すれば，たとえばクエリの D 画像と対応する RGB 画像を基にクエリと同一オブジェクトを写した RGB 画像とそれらに対応する D 画像を探し，それら全てから部分モデルをとってそれぞれ完全モデルを検索し結果を統合すれば，精度が上がることが期待できる．

画像の検索機構に関しては，MAP が最大でも閾値が 0.8 のときの 0.252 と 3D モデルマッチングの機構と比べて低い値である．閾値が 0.8 のときの P@k の結果を見ても，適合する画像が二つあることを考えると値が低い．こうなった原因を探るため，オブジェクトを遠くに写した画像のみを検索対象とした場合の AP を計測し，MAP を計算した結果，閾値が 0.8 のとき，MAP は最大の 0.455 となった．

この結果から，精度が出ない原因は主に物体がレンズの近くにある場合における輪郭の誤差計算の機構の問題だと推察できる．そしてそれは，モデルを投影する際のモデルとカメラの距離が適切でないために発生していると考えられる．本研究では，モデルを投影をする際のモデルとレンズの距離は，どのような場合でも 3.5.2 節で述べたように一律に定めた．しかし，距離をこのように定めたことにより問題も発生した．たとえば図 13

図 13　カメラとモデルが近すぎる場合の投影

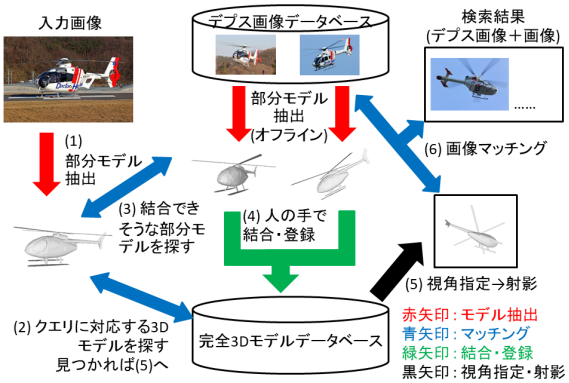

図 14　システムの改良

に示した，飛行機の 3D モデルを回転させ，$|Z_{min}| \times R \times 3$ の距離から投影し生成した画像の例がそれにあたる．3D モデルを平面に投影する場合，レンズの近くにあるものほど大きく射影される．この例では，レンズとモデルの距離が近すぎたことで射影がゆがみ，一部は画像の外へとはみ出してしまったと考えられる．解決策としては，一律に定めた距離の定義を調整するか，モデルの形状や視点ごとに距離の定義を変えるという方法が挙げられる．

最後に，入力画像に対応する 3D モデルが存在しない場合には，現段階のシステムでは検索要求に対応できなくなってしまうという問題がある．これに対応するために考えられるのが，図 14 のような拡張である．さまざまな角度から撮影した D 画像 (部分モデル) を結合して一つの 3D モデルにする手法が研究されている [12]．この手法を応用し，入力画像に対応する 3D モデルが存在しない場合には，D 画像のデータベース中から結合できそうな部分モデルを検索し，それらをつなぎ合わせて完全モデルを作り，そのまま検索を続行するという方法が考えられる．自動で結合する手法 [12] を適用することも考えられるが，正確性と時間の観点から，システムが結合できる可能性が高いモデルを示し，それらを人の手で結合するという解決策もある．単純にクラウドソーシングを利用して結合を進めることも考えられるが，システムを利用するユーザーが新しくモデルを結合していくことで，ユーザー同士が協力することによってシステムが充実していくような仕組みも考えられる．

5.　実画像への適用

前節までの評価実験でクエリや検索対象として用いた 3D モデルや画像は，そのほとんどが 3D モデルのデータセットである PSB データセット [7] から生成したノイズをほとんど含まない 3D モデル，およびそのモデルを使用して生成した画像である．特に，画像の検索機構に関しては，評価実験の際には PSB データセット由来のデータのみを用いている．

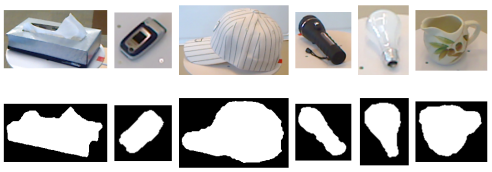

図 15　動作検証に使用する画像

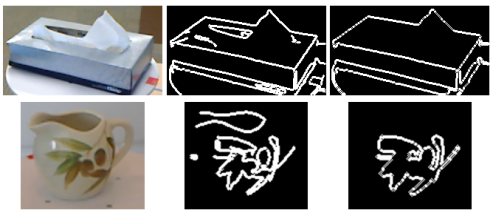

図 16　実画像からの最長輪郭検出

そこで本研究で構築したシステムの検索対象を，RGB-D データセット [6] の画像のみに絞った場合の動作を検証する．この節で行う検証について，ViSP トラッカーの閾値の値は 0.8 で固定とする．また，ここでの検索対象の画像は図 15 の上側の 6 枚とする．6 枚の画像がそれぞれ正解となるような 3D モデルと姿勢をクエリとし，それぞれの場合について画像の検索を行った際の適合率を計測し平均すると，P@1 = 0.167, P@2 = 0.167 となった．4.2 節で述べた結果と比べて検索対象の画像数が減っているのにも関わらず P@k の値が下がっているため，実画像を検索対象とした場合，理想的な画像を検索対象とした場合と同じ条件下でも検索精度は低下すると言える．

原因は，画像中のノイズの影響で物体の位置検出を誤ること，および画像中に写されている物体の輪郭ではない部分から検出されるエッジが輪郭誤差の計算に使われることだと考えられる．図 16 は左の画像から輪郭線のみを抽出した画像 (中央)，および最長輪郭のみを抽出した画像 (右) を表している．本研究における画像中からの物体検出は，その画像から検出できる最長輪郭に依存している．そのため図 16 のように物体の輪郭が正しく取れない場合には，物体が誤った位置に存在するものとしてしまう．その結果，トラッカーに与えるカメラパラメータの値や物体の位置の推定を誤る．加えて画像中の物体に柄などがついている場合，そこから検出されたエッジと誤って投影された 3D モデルの輪郭線とを比較してしまい，クエリと画像との距離が妥当ではない値になっていると考えられる．

このことを確かめるため，検索対象を先ほどの各画像に対応した，画像中の被写体が占める領域を表した画像に変更し，再度検証を行った．用いた画像は，RGB-D データセット中に含まれる図 15 の下側の 6 枚の画像である．6 枚の画像がそれぞれ正解となるような 3D モデルと姿勢をクエリとし，それぞれの場合について画像の検索を行った際の適合率を計測し平均すると，P@1 = 0.333, P@2 = 0.333 となった．加えて，各オブジェクトの画像を検索した結果にも変化が見られた．たとえば図 16 の左側に示した水差しの画像が正解である場合の試行では，検索対象の画像を図 15 の上側の 6 枚としたときの検証では正解の画像が 4 位にランキングされた．これに対して，画像

を被写体の領域を表す画像に変えたときの検証では，正解の画像は 1 位にランキングされた．一方，図 16 の左側に示した箱ティッシュの画像が正解である場合の試行では，前者の検証のときは正解の画像が 2 位にランキングされた．しかし，後者の検証のときは正解画像との距離が計算されず，1 位にランキングされたのは電球の領域を表す画像という結果となった．

これはデータセット内の箱ティッシュの領域を表すとされている画像が，対応する画像中の箱ティッシュの領域を正しく表していないことが原因と考えられる．図 15 は，下側の画像が上側の画像中の被写体が占める領域を表した画像となるように示している．用いた箱ティッシュの画像は図 15 の最も左側にある．上下の画像を比べてみると，上側の画像における箱の右側の側面や左上の頂点付近の部分が，下側の画像では物体の領域としてみなされていないことがわかる．これは，画像を撮影した時の深度情報などを基に上側の画像の物体位置を推定して生成されたものが下側の画像である [6] ことが原因である．その結果，3D モデルを射影した際の輪郭と正解画像から検出できるエッジとの差が大きくなったと考えられる．

加えて，トラッカーによるエッジ検出の特性も影響していると考えられる．トラッカーは，射影された 3D モデルの輪郭とあまりにも離れた場所から検出されたエッジを，輪郭線の妥当性の判断や輪郭の誤差計算には用いない．そのため，画像中の被写体とモデルの射影が描く輪郭が違う場合においても，二つの輪郭線の一部のみが重なり，ほかの部分は大きく離れているような場合では，トラッカーが二つの輪郭がよく重なっている部分からのみエッジを検出することにより，数少ないエッジを基に誤差が小さく計算されることがある．エッジ検出数の例を挙げると，水差しの領域を表す画像を正解としたときの検証では，1 位の水差しの領域を表す画像に対するトラッカーによるエッジの検出点数は 146 であった．しかし，箱ティッシュの領域を表す画像を正解としたときの検証では，1 位の電球の領域を表す画像に対するエッジの検出点数は 23 であった．

以上のことから，画像から被写体の領域を切り取る際の正確性の向上，およびエッジの検出点数を基に射影された輪郭線の妥当性の考慮が本研究の課題として挙げられる．

6. ま と め

本研究では，D 画像を入力とし，その画像中のオブジェクトをユーザーによって指定された方向から見た画像を提示する検索システムの構築方法を提案した．加えて本研究では構築したシステムのうち，検索を行う二つの機構について評価実験により性能を検証した．その結果，特に背景などのノイズを含まない理想的な画像を検索対象とした場合，それぞれの検索精度に関しては本研究で提案した手法の有効性が確認できた．

今後の課題は，画像の検索機構の改良よる処理時間の短縮や検索精度の向上，3D モデルマッチング機構や画像の検索機構をさらに正確に評価するためのデータセットの再考，およびクエリに対する完全モデルが無い場合に検索を続行可能にする機構の設計と構築である．

謝　　辞

本研究の一部は，JSPS 科研費 (JP15H02701，JP16H02908，JP15K20990，JP17K12684)，JST ACT-I の助成を受けたものである．ここに記して謝意を表す．

文　　献

[1] Radu Bogdan Rusu and Steve Cousins, “3D is here: Point Cloud Library (PCL),” IEEE International Conference on Robotics and Automation (ICRA), pp. 1–4, 2011.

[2] Yi Liu, Hongbin Zha and Hong Qin, “Shape Topics: A Compact Representation and New Algorithms for 3D Partial Shape Retrieval,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06), pp. 2025–2032, 2006.

[3] Andrew Edie Johnson and Martial Hebert, “Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes,” IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 21 No. 5, pp. 433–449, 1999.

[4] Éric Marchand, Fabien Spindler and François Chaumette, “ViSP for Visual Servoing : A Generic Software Platform with a Wide Class of Robot Control Skills,” IEEE Robotics & Automation Vol. 12 Issue 4, pp. 40–52, 2005.

[5] Andrew I. Comport, Eric Marchand, Muriel Pressigout and François Chaumette, “Real-Time Markerless Tracking for Augmented Reality: The Virtual Visual Servoing Framework,” IEEE Transactions on Visualization and Computer Graphics Vol. 12 Issue 4, pp. 615–628, 2006.

[6] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox, “A Large-Scale Hierarchical Multi-View RGB-D Object Dataset,” IEEE International Conference on Robotics and Automation (ICRA), pp. 1817–1824, 2011.

[7] Philip Shilane, Patrick Min, Michael Kazhdan and Thomas Funkhouser, “The Princeton Shape Benchmark,” Shape Modeling International, pp. 167–178, 2004.

[8] Jingrui He, Mingjing Li, Hong-Jiang Zhang, Hanghang Tong and Changshui Zhang, “Manifold-Ranking Based Image Retrieval,” Proceedings of the 12th annual ACM international conference on Multimedia, pp. 9–16, 2004.

[9] Yushi Jing and Shumeet Baluja, “PageRank for Product Image Search,” Proc. 17th international Conference on World Wide Web (WWW2008), pp. 307–316, 2008.

[10] E.G. Karakasis, A. Amanatiadis, A. Gasteratos and S.A. Chatzichristofis, “Image moment invariants as local features for content based image retrieval using the Bag-of-Visual-Words model,” Pattern Recognition Letters Vol. 55, pp. 22–27, 2015.

[11] Ying Liu, Dengsheng Zhang, Guojun Lu and Wei-Ying Ma, “A survey of content-based image retrieval with high-level semantics,” Pattern Recognition Vol. 40, Issue 1, pp. 262–282, 2007.

[12] Sotiris Malassiotis and Michael G. Strintzis, “Snapshots: A Novel Local Surface Descriptor and Matching Algorithm for Robust 3D Surface Alignment,” IEEE Transactions on Pattern Analysis and Machine Intelligence Vol. 29, No. 7, pp. 1285–1290, 2007.

[13] David Eigen, Christian Puhrsch and Rob Fergus, “Depth Map Prediction from a Single Image using a Multi-Scale Deep Network,” Advances in Neural Information Processing Systems 27, pp. 2366–2374, 2014.

[14] Yasuhiro Fujiwara, Go Irie, Shari Kuroyama and Makoto Onizuka, “Scaling Manifold Ranking Based Image Retrieval,” Proceedings of the VLDB Endowment Vol. 8, No. 4, pp. 341–352, 2014.