社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

陰関数表現とRGB-D画像を用いた実寸通り食事と食器の三次元再構成

成冨　志優†　　柳井　啓司†

† 電気通信大学 大学院情報理工学研究科 情報学専攻
E-mail: †naritomi-s@mm.inf.uec.ac.jp, ††yanai@cs.uec.ac.jp

あらまし　食事のカロリー量管理は近年ではマルチメディア分野の研究においては重要なトピックであり, 多くの研究者や企業が食品のカロリー量管理に関する研究やアプリケーションの研究開発を行っている. 食事のカロリー量を推定するための多くの手法は画像認識を用いている. ただしこれらの手法は実際の食事は3次元オブジェクトであるにも関わらず, 食事を平面的にしか捉えていないため, 三次元的な盛り上がりや奥行きなどを考慮出来ていないという問題がある. これを解決するため, 近年発展を遂げている深層学習を用いた三次元再構成技術を活用したいが, そのほとんどの手法が再構成されるオブジェクトは正規化が行われているため実寸が分からず, カロリー量推定などには活用しづらい. そこで本論文ではRGB-D画像とカメラパラメータを用いた, 実寸通りの食事と食器の三次元形状を再構成する陰関数表現を用いた手法を提案する.

キーワード　三次元再構成, 陰関数表現, 食事画像

1. はじめに

IT技術を用いた食事のカロリー量管理は近年重要なトピックであり, カロリー量を推定するさまざまな手法やアプリケーションが研究開発されている. 食事のカロリー量を推定する既存の方法の多くは 画像認識を用いている. そのため実際の食事は3次元オブジェクトであるにも関わらず, 二次元的な認識しかしていない手法[1], [2]が多い. 三次元的に認識する手法[3], [4]も存在するが, これらの研究では, 食品は平皿の上になければならないという制約が存在するため, 丼などに対応ができない. そこで我々は以前, 食事を三次元的に認識するために, 単一のRGB画像から, 食事 (食品 + 食器) と食器の三次元再構成を行う “Hungry Networks” [5] という陰関数表現を用いた手法を提案し, 高精度な再構成, 体積推定を実現した.

しかし Hungry Networks には, 1つの課題が存在していた. それは再構成された三次元形状は正規化されているという点である. 正規化された三次元形状では, 実体積が分からないため, 再構成とは別に実寸を計測する必要があった. この課題を解決するために, RGB-D画像と古典的なカメラモデルである透視投影モデルを活用し, 正規化されていない, 実寸通りの三次元形状を再構成する陰関数表現を用いた手法を提案する. 実験の結果から, 本手法は実寸通りの三次元形状の再構築を精度よく実現し, 実体積推定に活用出来る事を示した.

2. 関連研究

2.1 単一RGB画像からの三次元再構成

深層学習を用いて三次元再構成を行う場合, どのような表現で再構成を行うかは重要な設定である. 表現方法を大きく分けると, ボクセル, 点群, Mesh, 陰関数のいずれかに分類する事が出来る. ボクセル表現を用いる手法[6], [7]や点群表現を用いる手法[8]では計算コストが高く高解像度にできなかったり, 複雑な後処理が必要になるなどの問題があった. そして Mesh 表現は, Mesh テンプレートを使用する手法[9], [10]や, 動的に Mesh テンプレートを生成してから最適化する手法[11]などが存在する. Mesh 表現はボクセル表現を用いる手法に比べメモリ効率よく高解像度にでき, 点群と違い点同士の接続情報もあるので形状も取れるなど, 利点が多い. しかし, mesh テンプレートをベースにした手法では, あくまでテンプレート mesh を変形させて目標の三次元形状に近付けるため, 表現力が低く, 自己交差が起こってしまうなどの問題があり, あまり良い結果が得られなかった. そこで近年では陰関数表現による三次元形状表現が注目されている. 陰関数表現を用いた手法[5], [12], [13]は三次元形状をスカラー場で表現する関数を学習する. 最終的に, 推論したスカラー場に Marching Cube[14]を適用する事で Mesh として三次元形状を取り出す. こうした陰関数表現は, これまでの手法よりも圧倒的に表現力が高く, 高精度な再構成を実現し, メモリ効率もネットワークサイズも優れていたため画期的であった.

2.2 深度画像を用いた三次元再構成

RGB-D画像を用いた三次元再構成手法には深度画像のみを用いる手法[15]と, RGB画像と深度画像の双方を用いるもの手法[16]がある. これらの深度画像を用いる手法では, RGB画像の有無にかかわらず深度画像は volumetric grid 表現に変換され, ネットワークの入力として用いられる. volumetric grid 表現に変換する手法は 3D Convolution やボクセル表現と相性がいいが, 3D Convolution は計算コストが高く, ボクセル表現は陰関数表現と比べて表現力が低く高解像度にできない問題を抱えている. そこで本論文で提案する手法では, 深度画像と RGB 画像の特徴量を volumetric grid 表現に変換する事なく統合する, 陰関数表現を用いた再構築手法を提案する.

2.3 三次元形状を考慮した食事認識

Chen らの手法[17]では深度センサを用いて深度画像を撮影し, 食事のカロリー量を推定を行っている. また古典的な複数視点によるカメラ行列を推定することで三次元形状を復元する手法として, Puri ら[18]の手法や, DietCam[19]などが存在する. 近年では CNN を用いた研究が発展している. Lu ら[20]は深度画像を深層学習を用いて生成し, 生成した深度画像から食事の量を推論しようとした. Im2calories[3]では RGB 画像か

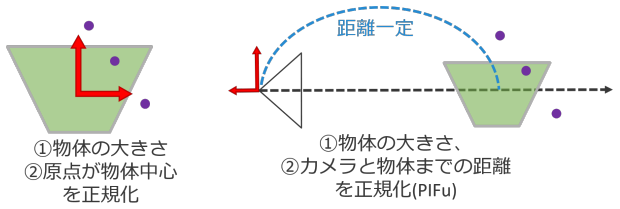

図 1　左: 殆どの陰関数表現で行われる正規化, 右: PIFu [13] で行われる正規化

らボクセル形式で三次元形状を推定し, カロリー量推定に活用している. また近年では, Nutrition5k [21] という栄養価がアノテーションされた 5000 枚の RGB-D 画像データセットなどが公開されている.

3. 手　　法

本手法では, RGB 画像と深度画像, そしてカメラパラメータを用いて, 正規化されていない, 実寸通りの食事と食器の 2 つの三次元形状を水密な Mesh として取り出す. 本手法でもっとも特徴的な点は, 占有率場の推論を正規化された空間ではなく, 実寸に対応する空間で行う事である. これは過去の研究である Hungry Networks [5] の問題点であった, 再構築される三次元形状は正規化されているため, カロリー量推定などにに利用するためには再構成とは別の手段で実寸を計算する必要があった点を解決する事が出来る. これを実現するため, 正規化された空間内での三次元再構成という深層学習にとって学習しやすい設定を捨て, 古典的なカメラモデルの一つである透視投影モデルと深度画像を活用する手法を提案する.

3.1　カメラモデルと深層学習

一般的に, 透視投影モデルと深度画像を用いれば, 画像中の物体の実寸を計算することができる. しかし深度画像を用いるだけでは, カメラから見えているオブジェクトの表面の形状は分かるものの, 背面などの形状は分からないため体積は計算できない. そのため深層学習による三次元再構成を行い, 完全な三次元形状を得る必要がある. そこで透視投影モデルと深度画像を活用し, 正規化されていない実寸通りの三次元形状の再構成を行いたい. しかし, 図 1 の左に示すように, 陰関数表現を用いた三次元再構成手法では, 物体の中心に原点を設定し, 物体の大きさが一定になるよう正規化を行う. そのためカメラモデルと統合する事は難しい. そこでカメラモデルを活用した陰関数表現を用いて単一の RGB 画像から三次元再構成を行う PIFu [13] という手法に注目した. この手法ではカメラモデルに弱透視投影法を利用する事で, 図 1 の右のように, 物体の大きさとカメラから再構成対象オブジェクトまでの距離を正規化している. 弱透視投影を用いて正規化を行う事で学習可能になっていた一方で, 実際の深度や物体の大きさなどの情報は破壊されてしまうため, 実寸通りの再構成は出来ない. また特定の視点からしか正しく再構成できないという問題も存在する.

本研究で解決したい課題は正規化された三次元形状ではなく, 実寸通りの三次元形状の再構成を実現する事である. また, 同時に任意の角度からの画像を入力として扱えなければならない. なぜならアプリケーションとして利用する際, 特定の角度の入力にしか対応できないのでは使い物にならないからである. そのため, 本論文で提案する手法は弱透視投影投影ではなく, 透視投影を用いて実寸通りの三次元形状を再構成する. しかし透

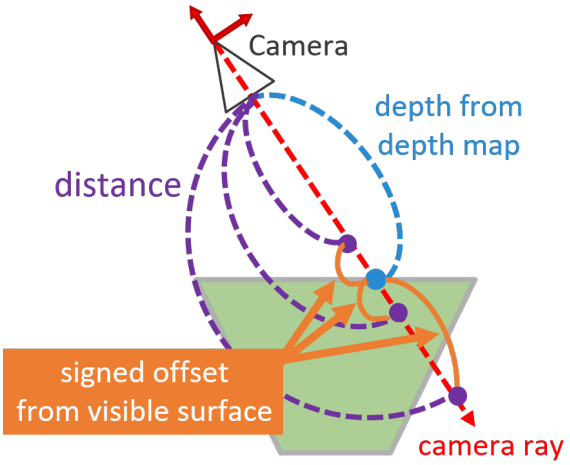

図 2　深度画像から分かる深度値 (depth) と占有率を求めたい点 (図中の紫の点) までの距離の符号付きのオフセットは, オブジェクトの表面を基準に一定の範囲に収まる. そのため深度を正規化しなくてもオフセットからオブジェクトの形状を学習可能であると考えられる.

視投影を用いる場合, 再構成対象空間の深度を弱透視投影のように正規化できないため学習が難しくなる. そこで本手法では RGB 画像に加えて, 深度画像も入力に用いる. 本手法では深度画像を (1) 占有率場を求めるべき空間の設定, (2) 可視表面までの距離のサンプリング, (3) 形状に関する特徴量の抽出, に用いる事で, 実寸通りの三次元再構成を実現するネットワークの学習を可能にし, 再構成精度も向上させた.

3.2　深度画像の活用

本項では実寸通りの三次元再構成を高精度に行うための深度画像の活用法を提案する.

3.2.1　占有率場を求めるべき空間の設定

三次元形状を得るためには, まず占有率場を求める空間を正しく設定しなければならない. 従来の正規化を行う手法では正規化された空間, 例えば x, y, z 各軸 [-0.5, 0.5] の間の空間から占有率を求めるべき座標をサンプリングすればよかった. しかし透視投影をそのまま活用する場合はサンプリングするべき空間が分からない. なぜなら, カメラから伸びる視推台は深度方向に対して制限がなく, 物体がどこに存在するか分からないためである. これは深度画像を用いる事で解決できる. なぜなら深度画像のピクセル値から, 食事がどの程度の深度にあるか判明するため, それに基づいて占有率場を求めるべき空間を設定する事が出来るからである.

3.2.2　可視表面までの距離のサンプリング

Hungry Networks のような, 正規化された Mesh を再構成する手法の場合は占有率場を x, y, z 軸それぞれ [-0.5,0.5] の空間で推論すればよかった. そして PIFu も, 弱透視投影を用いることで深度の正規化を行い, オブジェクトが存在する位置をある一定の深度内に収める事で 学習が上手くいっていた. しかし, 本手法のように正規化せず実寸のまま扱う場合, カメラとオブジェクトまでの距離はまったく異なるため, ネットワークを学習する事は難しい. そこで深度画像から得られる深度値を活用する. 図 2 から分かる通り, カメラから占有率を求めたい点 $p \in \mathbb{R}^3$ までの距離 (distance) と共に, 深度画像から取り出した深度値を用いれば, その深度値と点までの距離の差分から, カメラから見えているオブジェクト表面からどれだけ離れているか

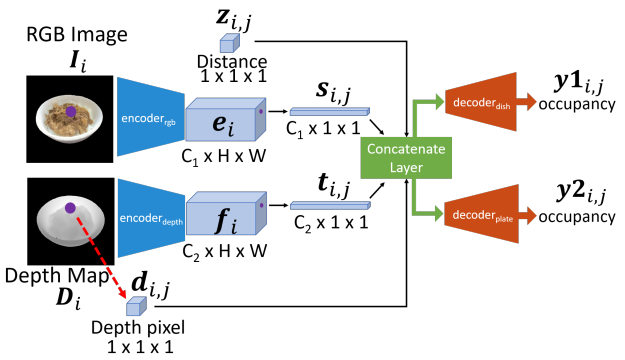

図 3　ネットワークは 2 つのエンコーダと 2 つのデコーダから成る．エンコーダはそれぞれ RGB 画像と深度画像の特徴量を抽出する．デコーダはそれぞれ食事と食器の占有率を推論する．

を示す符号付きのオフセットが分かる．このオフセットはオブジェクトの表面を基準に一定の範囲に収まるため，深度を正規化せずとも，これに基づいてネットワークは三次元再構成を正しく学習可能なのではと考えた．そこで深度画像から深度値をサンプリングし，デコーダの入力に用いる事とした．これを深度値サンプリングと呼称する．なお，後の実験でこの考えが精度に寄与するかを示す．

3.2.3　形状に関する特徴量の抽出

深度値サンプリングでは，深度画像からピクセルレベルで情報を取得していた．しかしそれだけでは，そのピクセル周辺の表面形状及び，物体全体の三次元形状に関する極めて有益な情報を取り逃している事になる．そこで深度画像も RGB 画像と同様に CNN を適用することで特徴量を抽出し，これを活用する事とした．これを深度特徴量と呼称する．

3.2.4　ネットワーク

本手法のネットワークの概要図を図 3 に示した．ネットワークは 2 つのエンコーダと 2 つのデコーダから成る．エンコーダはそれぞれ RGB 画像の特徴量抽出用と，深度画像の特徴量抽出用の 2 つである．デコーダは食事と食器のそれぞれの占有率を推論するために 2 つ存在する．

3.2.5　推　　論

推論時の動作について説明する．まず，サンプリングした点 $p \in \mathbb{R}^3$ とその点が投影された画像上の座標 $(u, v) \in \mathbb{R}^2$ を標準的な透視投影モデルを用いて計算する．そして次にエンコーダを用いて，RGB 画像，深度画像のそれぞれの特徴量を抽出する．ここではそれぞれの特徴量の形状は $C_1 \times H \times W$, $C_2 \times H \times W$ とする．推論時これらの特徴量は初めに一度だけ計算すればよい．そして次に，占有率の推論を行う．分かりやすさのため，例えば図 2 に紫の点で示した三次元座標 $p \in \mathbb{R}^3$ の占有率を推論すると考えてほしい．推論には，RGB 画像/深度画像のそれぞれの特徴量から，予め計算しておいた (u, v) に対応する座標を用いてバイリニアサンプリングした $C_1 \times 1 \times 1$, $C_2 \times 1 \times 1$ の特徴量と，同様に深度画像から (u, v) に対応する座標を用いてバイリニアサンプリングした深度値 $1 \times 1 \times 1$, 加えて p からカメラまでの距離である Distance を用いる．これらの 4 つの特徴量を concatenate layer で結合し，$(C_1 + C_2 + 2) \times 1 \times 1$ の特徴量を作成する．この作成した特徴量を 2 つあるデコーダにそれぞれ渡す．デコーダの中では 1 次元畳み込みし，占有率を推論する．

3.2.6　学　　習

ネットワークの学習のためのミニバッチロスを以下のように定める．

$$w \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{K} \left[\begin{array}{c|c} \mathbf{R} & \mathbf{T} \end{array} \right] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{1}$$

$$\boldsymbol{e}_i = \text{encoder}_{rgb}(I_i) \tag{2}$$

$$\boldsymbol{f}_i = \text{encoder}_{depth}(D_i) \tag{3}$$

$$(u, v)_{i,j} = \text{projection}(p_{i,j}, K_i, R_i, T_i) \tag{4}$$

$$\boldsymbol{s}_{i,j} = \text{sample}(\boldsymbol{e}_i, (u, v)_{i,j}) \tag{5}$$

$$\boldsymbol{t}_{i,j} = \text{sample}(\boldsymbol{f}_i, (u, v)_{i,j}) \tag{6}$$

$$\boldsymbol{d}_{i,j} = \text{sample}(D_i, (u, v)_{i,j}) \tag{7}$$

$$\boldsymbol{z}_{i,j} = \text{distance}(p_{i,j}, K_i, R_i, T_i) \tag{8}$$

$$\boldsymbol{c}_{i,j} = \text{concatenate}(\boldsymbol{s}_{i,j}, \boldsymbol{t}_{i,j}, \boldsymbol{d}_{i,j}, \boldsymbol{z}_{i,j}) \tag{9}$$

$$y1_{i,j} = \text{decoder}_{dish}(\boldsymbol{c}_{i,j}) \tag{10}$$

$$y2_{i,j} = \text{decoder}_{plate}(\boldsymbol{c}_{i,j}) \tag{11}$$

$$\mathcal{L}_{\mathcal{O}}(\hat{o}, o) = \mathcal{L}_{bce}(\hat{o}, o) \tag{12}$$

$$\mathcal{L}_{\mathcal{C}}(o1, o2) = \max(o2 - o1, 0) \tag{13}$$

$$\mathcal{L}_{\mathcal{B}} = \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \sum_{j=1}^{K} \Bigg(\lambda_1 \mathcal{L}_{\mathcal{O}}(y1_{i,j}, o1_i(p_{i,j})) + \lambda_2 \mathcal{L}_{\mathcal{O}}(y2_{i,j}, o2_i(p_{i,j})) + \lambda_3 \mathcal{L}_{\mathcal{C}}(y1_{i,j}, y2_{i,j}) \Bigg) \tag{14}$$

ここで，式 1 の $\mathbf{K}$ は 3×3 のカメラの内部パラメータ行列であり，$\mathbf{R}$, $\mathbf{T}$ はそれぞれ 3×3, 3×1 の回転，平行移動を表すカメラの外部パラメータ行列である．左辺の u, v は点 $p = (x, y, z)$ が画像上に投影された時の点の座標である．また式 2,3 の I_i および D_i はミニバッチの i 番目の RGB/深度画像であり，$\text{encoder}_{rgb}, \text{encoder}_{depth}$ は RGB, Depth それぞれの特徴量を抽出するエンコーダ，$\boldsymbol{e}_i, \boldsymbol{f}_i$ は抽出された特徴量である．式 4 の projection は式 1 の右辺を用いて計算した左辺の u, v を取り出す関数であり，式 5,6,7 の sample は与えられた特徴量 $\boldsymbol{e}_i, \boldsymbol{f}_i$ 及び深度画像 D_i から $(u, v)_{i,j}$ を用いてバイリニアサンプリングを行う関数であり，$\boldsymbol{s}_{i,j}, \boldsymbol{t}_{i,j}, \boldsymbol{d}_{i,j}$ はバイリニアサンプリングで取り出された特徴量である．なお，j はバッチ i 番目の三次元形状の周辺から K 個の点をサンプリングしたうちの j 番目の点を意味する．式 8 の distance はカメラ座標系上の点 $p_{i,j}$ の z 軸の値の絶対値を取ったもの，つまり距離を計算する．$\boldsymbol{z}_{i,j}$ はカメラ座標系の原点から点 $p_{i,j}$ までの距離ではなく，カメラの原点を通る，投影面に平行な面からの距離である事に注意し

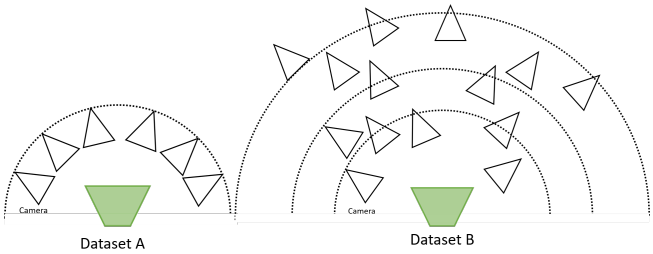

図 4　レンダリング方法によるデータセットの違い. Image dataset A より B の方が深度の分布が多様で学習が難しい.

てほしい. そして式 10,11 の $\text{decoder}_{dish}, \text{decoder}_{plate}$ はそれぞれ食事と食器の占有率を推論するためのデコーダである. 推論された占有率をそれぞれ $y1_{i,j}, y2_{i,j} \in \mathbb{R}$ とする. 式 12 の $\mathcal{L}_{\mathcal{O}}$ は占有率を学習するための binary cross entropy loss, 式 14 の $\mathcal{L}_{\mathcal{C}}$ は食器の一貫性を維持するための plate consistency loss [5] である. 最終的に得られた $y1_{i,j}, y2_{i,j}$ と, 式 14 を用いて学習を行う. なお, 式 2~11 の $\boldsymbol{e}_i, \boldsymbol{f}_i, \boldsymbol{s}_{i,j}, \boldsymbol{t}_{i,j}, \boldsymbol{d}_{i,j}, \boldsymbol{z}_{i,j}, \boldsymbol{y1}_{i,j}, \boldsymbol{y2}_{i,j}$ は図 3 と対応する.

3.3　データセット

本手法の学習/評価に用いるデータセットには Hungry networks で用いられている水密な Mesh データセットを用いて生成した. Hungry networks でのデータセットには RGB 画像が含まれていたが, RGB-D 画像は含まれていない. そこで本手法を学習するために Hungry networks と同一の mesh データから, RGB-D 画像をレンダリングした. なお, Hungry networks の学習に利用するために作成した水密な Mesh は正規化されているため, 本手法においては実寸大にもどしてから利用した.

3.3.1　RGB-D 画像のレンダリング

本手法を学習するために, 食事の Mesh を 2 通りの方法でレンダリングする事で 2 種類のデータセットを作成した. この撮影方法の違いを図 4 に示した. 1 つは, 食品を中心とした半径 20cm の半球状から約 30 点サンプリングして, そこから食事にカメラを向けてレンダリングした RGB-D 画像である. これは Hungry Networks でレンダリングした画像の条件に近いものとなっている. もう一つは, 食事を中心とした半径 20,30,40cm の半球状からそれぞれ 25 点ずつサンプリングし, それらに対して平均 0, 分散 2.5cm の正規分布からサンプリングしたノイズを加算した点から, 食事を視野に入れてレンダリングした RGB-D 画像である. 1 つめの RGB-D 画像に対して, 2 つ目の RGB-D 画像の方が画像中に占める食事の大きさが違うのに加えて, 図 4 が示している通り, 様々な距離から撮影されているため, 深度画像の多様性が 1 つめに比べて高くなっており, 学習が難しい. この 2 つの画像データセットをそれぞれ Image dataset A, B と呼称する. また本手法ではカメラの内部/外部パラメータが重要になってくるため, RGB-D 画像を撮影するとともに, その際のカメラパラメータも同時に保存した. なお, レンダリングには Open3D と OSMesa を用いた.

4. 実　　験

本手法は RGB-D 画像, カメラパラメータを入力として, 食事と食器の実寸三次元再構成を行う. 本手法では, 特徴量からバイリニアサンプリングを行うため, エンコーダが出力する特徴量の大きさは重要であると考えた. そこでまず, 様々なエンコーダを用いて学習を行い, 定量, 定性的な評価を行った. この実験には学習が比較的簡単な Image dataset A を用いた. その後, 深度画像をどのように取り扱うのがもっともよい精度が得られるか, つまり深度値サンプリングや深度特徴量がどのような影響を与えるかというのを, Image Dataset B と 1 つ目の実験で精度が良かったエンコーダを採用したネットワークで実験した.

表 1　各エンコーダの入力サイズに対する出力サイズ.

backbone	input	output
Custom UNet	$3 \times 224 \times 224$	$128 \times 112 \times 112$
ResNet50 (Layer 1)	$3 \times 224 \times 224$	$255 \times 56 \times 56$
ResNet50 (Layer 2)	$3 \times 224 \times 224$	$512 \times 28 \times 28$
ResNet50 (Layer 3)	$3 \times 224 \times 224$	$1024 \times 14 \times 14$
ResNet50 (Layer 4)	$3 \times 224 \times 224$	$2048 \times 7 \times 7$

4.1　エンコーダ

実際の実験に入る前に, どのようなエンコーダを利用したかを紹介する. 画像の特徴量を抽出するための Encoder には Custom UNet, ResNet50 Layer4, ResNet50 Layer1-4 の 3 つを用意した. Custom UNet は独自に実装した Unet like なアーキテクチャのネットワークである. ResNet50 Layer4 は ResNet50 の最終層の出力を用いるもの. ResNet50 Layer1-4 は ResNet50 の中間層が出力する特徴量全てを用いるものである. それぞれのネットワークの大きな違いは出力される特徴量のサイズである. 表 1 に示した通り, Custom UNet が出力する特徴量は width と height が大きく, ResNet50 Layer4 の出力は width と height が小さくチャンネル数が大きい.

4.2　評 価 指 標

評価指標は 4 つ用意した. 1 つ目は IoU, これは再構成された Mesh と真値の Mesh の間の union と intersection の商である. 2 つ目は Chamfer L1 distance, これは再構成された Mesh 上の点から, 真値の Mesh 上の点までの最近傍点までの距離と, 真値の Mesh 上から, 再構成された Mesh 上の点までの最近傍点までの距離との平均で計算される. この点はそれぞれの Mesh からサンプリングした 10 万点をもちいた. 3 つ目は Food volume error, これは食品の絶対体積誤差を表す. 食品の体積は食事 (dish) と食器 (plate) の差分から計算したものである. 4 つめは Relative food volume error, これは食品の相対体積誤差である. これは真値の食品体積に対して, 推論した食品体積がどの程度の割合でずれているのかを表す.

4.3　食事と食器の三次元再構成

まず, 本手法で正しく再構成ができるか, 先ほど述べた 3 種類のエンコーダを用いて, 実験を行った. 学習には比較的学習が簡単である Image dataset A を用いた. その定量評価の結果を表 2 に示した.

結果的に Custom Unet を用いる手法が再構成精度, 体積推定ともに良かった. 次に定性的な評価であるが, 再構成結果を図 5 に示した. Custom UNet を用いた場合が定量的にもっとも精度が良かったが, 定性的にも良好であることが見て取れる. ResNet50 の最終出力を用いた場合は食品の形状の詳細が完全に失われ, ResNet50 の中間特徴を用いた場合でも, Custom Unet を用いた手法に比べると食事の詳細が正しく捉えられていない事が分かる. この実験から, エンコーダの特徴量のサイズが極めて重要であると分かった.

4.4　深度画像の活用手法での違い

今までの実験では, 深度の分布がある程度定まっている Image dataset A で学習していた. しかし, 実際のアプリケーションで使う場合は深度が一定に定まっている事はない. そこで, より現

表 2　RGB-D 画像を用いた食事と食器の三次元再構成の定量的結果. 表中の C-L1 は Chamfer L1 distance, FVE は Food Volume Error, r-FVE は relative Food Volume Error を意味する.

encoder	C-L1 (dish)	IoU (dish)	C-L1 (plate)	IoU (plate)	Mean FVE (cm^3)	Median FVE (cm^3)	Mean r-FVE	Median r-FVE
Custom UNet	**0.00341**	**0.702**	0.00581	**0.537**	**73.253**	**46.046**	0.595	**0.13**
ResNet50 (Layer 4)	0.00445	0.636	0.00607	0.437	167.271	99.129	0.658	0.377
ResNet50 (Layer 1-4)	0.00516	0.558	**0.00566**	0.470	80.4859	54.293	0.633	0.166

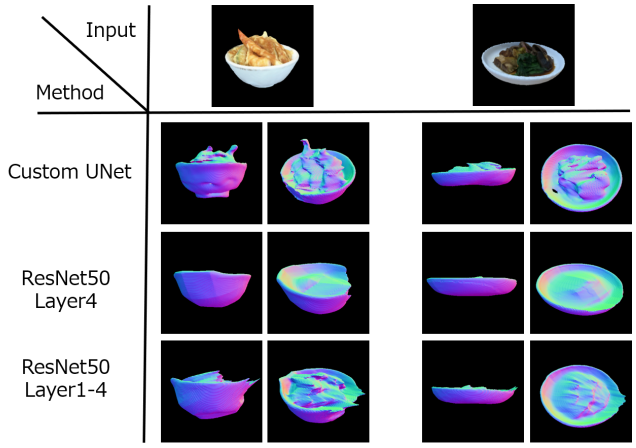

図 5　エンコーダの違いによる再構成結果の比較.

実に近い設定の Image dataset B を学習に用いて, 深度画像をどのように扱うのが高い精度を実現出来るかを実験した. なお, 全ての手法のエンコーダに先ほど最も精度が良かった Custom UNet を採用した. その結果を表 3 に示した. 表 3 から, 深度値サンプリングと, 深度特徴量の双方を利用する手法がもっとも精度が良い事が分かる. その最も良かった手法と, 深度特徴量のみを用いる 2 つの手法で, どのような再構成を行うかを定性的に示したのが図 6 である. 深度画像の特徴量だけでなく, 深度値のサンプリングも行いデコーダの入力として用いることで, 食事の再構成結果の輪郭がくっきりとし, また食器についても正確に再構成できるようになっている事が見て取れる. また, もっとも精度が良かった, エンコーダに Custom UNet を採用し, 深度値サンプリングと深度特徴量を利用した手法での再構成結果を図 7 に示した.

4.4.1　体積誤差の分析

表 3 で最も精度が良かった, エンコーダに Custom UNet を採用し, 深度値サンプリングと深度特徴量の双方に用いるモデルの体積誤差について分析を行った. 体積誤差の分布をグラフにすると図 8 のようになった. 多くのデータで絶対誤差が $50cm^3$ 以下, 相対誤差が 0.2 以下となっており, 精度が良い事が分かる. 一方で, 体積誤差が極端に大きいモデルもある. 相対誤差が極端に悪い, 2 つの入力とその再構成結果を図 9 に示した. 双方ともに, 再構築される食器の形状が食事に比べて薄かったり, 欠けてしまっている事が分かる. plate consistency loss は食器が占有している空間を食事が占有していない問題を修正する損失関数であるが, その逆は修正が効かないためこのような問題が起きていると考えられる.

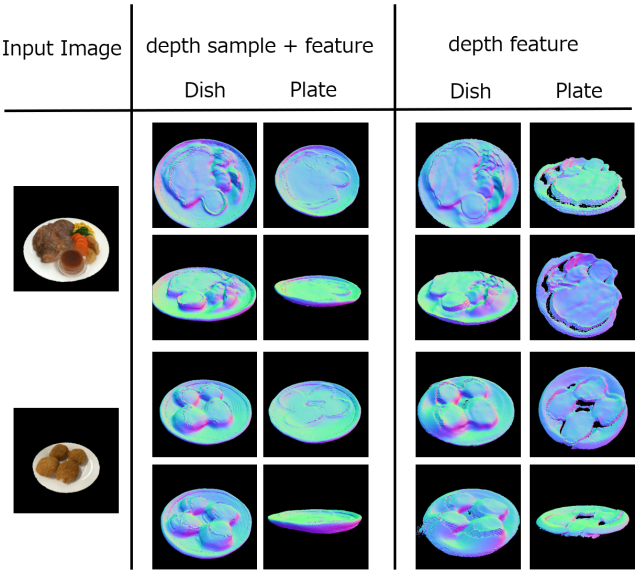

図 6　エンコーダに Custom UNet, 深度画像を深度値のサンプリングと特徴量抽出に用いるモデルと深度画像の特徴量のみを用いるモデルの比較. 深度値のサンプリングは食事の再構成結果の輪郭をくっきりとし, 食器についても正確に再構成できるよう貢献をしている事がわかる.

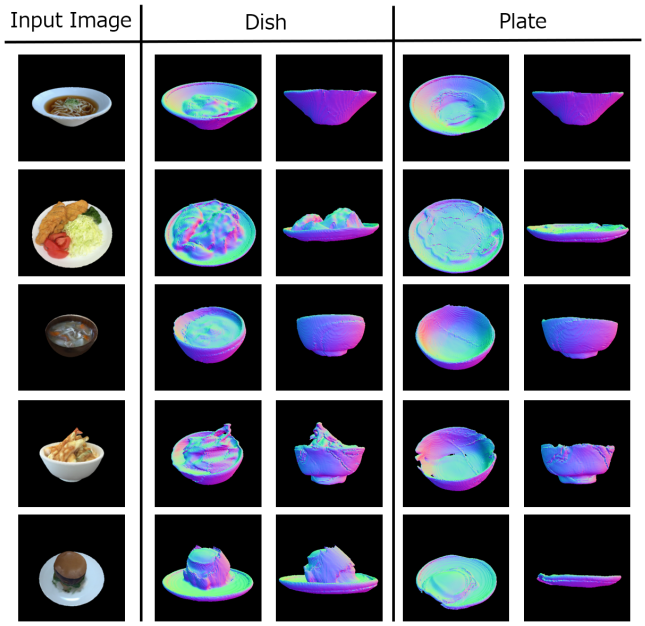

図 7　エンコーダに Custom UNet, 深度画像を深度値のサンプリングと特徴量抽出に用いるモデルで Image dataset B で学習した結果.

5.　おわりに

本研究では, 我々の過去の研究である “Hungry Networks” [5] の課題であった, 再構成されるオブジェクトは実寸ではないという課題を解決するため, RGB-D 画像とカメラモデルを活用した実寸三次元再構成手法を提案した. この手法では深度画像を活用する事で三次元形状の正規化を行う事なく, 実寸通りに高精度に再構成し, その再構成結果を用いる事で, 精度の高い実体積推定を実現した. 今後の課題としては, 推定した体積を元にカロリー量推定などに活用していきたいと考えている.

表 3　どのような深度画像の取り扱いが最も精度が出るかの比較．Depth の S は深度値サンプリング，C は深度特徴量を意味する．それ以外は表 2 の表記と同一．

Depth	valid	C-L1 (dish)	IoU (dish)	C-L1 (plate)	IoU (plate)	Mean FVE (cm^3)	Median FVE (cm^3)	Mean r-FVE	Median r-FVE
C, S	24/24	**0.00307**	**0.567**	**0.00498**	**0.407**	**79.524**	**51.24**	0.688	**0.150**
C	24/24	0.00333	0.534	0.00592	0.337	112.314	90.291	**0.616**	0.259
S	23/24	0.00847	0.356	0.0104	0.124	125.344	104.122	0.791	0.32
None	1/24	invalid	invalid	invalid	invalid	invalid	invalid	invalid	invalid

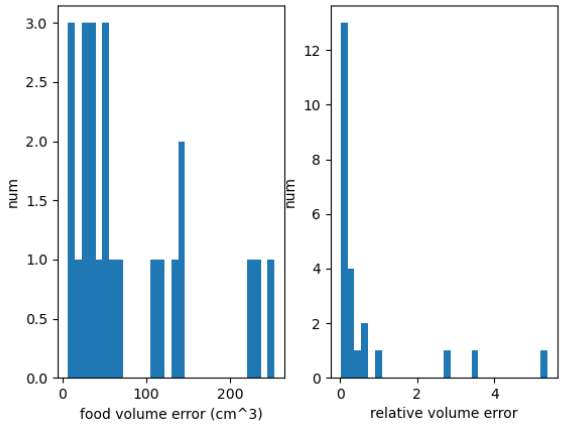

図 8　エンコーダに Custom UNet, 深度値サンプリングと特徴量抽出の双方に用いるモデルを使って評価データセットで体積誤差をとった時の分布．左が絶対誤差で，右が相対誤差の分布となっている．絶対誤差の平均が 79.524(cm^3), 中央値が 51.241(cm^3) であった．また相対誤差の平均値が 0.688, 中央値が 0.15 であった．

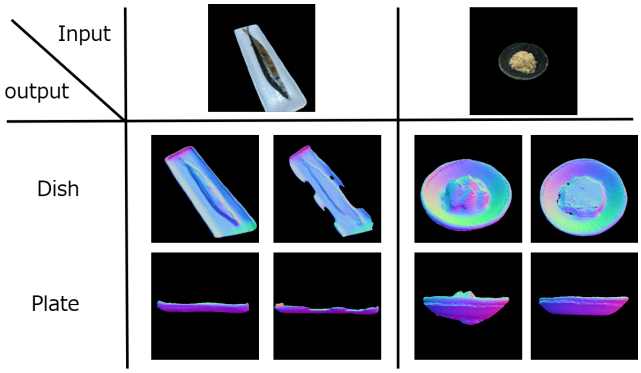

図 9　相対誤差が最も大きかった 2 つの入力画像と再構成結果．

文　献

[1] T. Ege and K Yanai. Image-based food calorie estimation using recipe information. *IEICE Transactions on Information and Systems*, Vol. E101-D, No. 5, pp. 1333–1341, 2018.

[2] T. Ege and K. Yanai. Imag-based food calorie estimation using knowledge on food categories, ingredients and cooking directions. In *Proc. of ACM Multimedia Thematic Workshop*, 2017.

[3] A. Meyers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K. P. Murphy. Im2Calories: towards an automated mobile vision food diary. In *ICCV*, pp. 1233–1241, 2015.

[4] Y. Ando, T. Ege, J. Cho, and K. Yanai. DepthCalorieCam: A mobile application for volume-based foodcalorie estimation using depth cameras. In *Proc. of the 5th International Workshop on Multimedia Assisted Dietary Management*, pp. 76–81, 2019.

[5] S. Naritomi and K. Yanai. Hungry Networks: 3D mesh reconstruction of a dish and a plate from a single dish image for estimating food volume. In *Proc. of ACM Multimedia Asia*, 2020.

[6] S. Tulsiani, T. Zhou, A. A. Efros, and J. Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *CVPR*, pp. 2626–2634, 2017.

[7] C. B. Choy, Danfei. Xu, J. Gwak, K. Chen, and S. Savarese. 3D-R2N2: A unified approach for single and multi-view 3d object reconstruction. In *ECCV*, 2016.

[8] H. Fan, H. Su, and L. J. Guibas. A point set generation network for 3d object reconstruction from a single image. In *CVPR*, pp. 605–613, 2017.

[9] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y. G. Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*, pp. 52–67, 2018.

[10] Y. Nie, X. Han, S. Guo, Y. Zheng, J. Chang, and J. J. Zhang. Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In *CVPR*, June 2020.

[11] G. Gkioxari, J. Malik, and J. Johnson. Mesh R-CNN. In *ICCV*, pp. 9785–9795, 2019.

[12] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy Networks: Learning 3d reconstruction in function space. In *CVPR*, pp. 4460–4470, 2019.

[13] S. Saito, Z. Huang, R. Natsume, S. Morishima, A. Kanazawa, and H. Li. PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *ICCV*, 2019.

[14] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, Vol. 21, No. 4, pp. 163–169, 1987.

[15] S. Song, F. Yu, A. Zeng, Angel X Chang, M. Savva, and T. Funkhouser. Semantic scene completion from a single depth image. In *CVPR*, 2017.

[16] j. Li, Y. Liu, Gong D., Q. Shi, X. Yuan, C. Zhao, and I. Reid. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In *CVPR*, 2019.

[17] M. Y. Chen, Y. H. Yang, C. J. Ho, S. H. Wang, S. M. Liu, E. Chang, C. H. Yeh, and M. Ouhyoung. Automatic chinese food identification and quantity estimation. In *Proc. of SIGGRAPH Asia 2012 Technical Briefs*, pp. 1–4, 2012.

[18] M. Puri, Zhiwei Zhu, Q. Yu, A. Divakaran, and H. Sawhney. Recognition and volume estimation of food intake using a mobile device. In *WACV*, pp. 1–8, 2009.

[19] F. Kong and J. Tan. Dietcam: Regular shape food recognition with a camera phone. In *2011 International Conference on Body Sensor Networks*, pp. 127–132, 2011.

[20] Y. Lu, D. Allegra, M. Anthimopoulos, F. Stanco, G. M. Farinella, and S. Mougiakakou. A multi-task learning approach for meal assessment. In *Proc. of the Joint Workshop on Multimedia for Cooking and Eating Activities and Multimedia Assisted Dietary Management*, pp. 46–52, 2018.

[21] Q. Thames, A. Karpur, W. Norris, F. Xia, L. Panait, T. Weyand, and J. Sim. Nutrition5k: Towards automatic nutritional understanding of generic food. In *CVPR*, 2021.