

# CNNによる複数品食事画像の同時カロリー推定とそのモバイル実装

會下 拓実<sup>†</sup> 丹野 良介<sup>†</sup> 柳井 啓司<sup>†</sup>

<sup>†</sup> 電気通信大学大学院情報理工学研究所 〒182-8585 東京都調布市調布ヶ丘 1-5-1

E-mail: <sup>†</sup>{ege-t,tanno-r}@mm.inf.uec.ac.jp, <sup>††</sup>yanai@cs.uec.ac.jp

あらまし 近年では、食に関する健康志向の高まりにより個人の食事記録を支援するアプリケーションが数多く登場している。そうしたアプリの中には画像認識技術により料理写真から料理名の候補を自動で提案するものまで存在するが、料理のカロリー量計算に関しては、ユーザーが入力した情報からカロリー量を推定するものや、栄養士の方が料理画像からカロリー量を推定するものなどがある。このようなアプリにより栄養学の知識のない一般ユーザーがカロリー量を記録することが可能になったが、これらのカロリー量推定方法は人手による操作に手間やコストがかかるという問題がある。そこで本研究では、Convolutional Neural Network(以下 CNN)を用いた画像認識手法を用い、複数料理画像の同時カロリー量推定を行い、さらにそのモバイル実装を行う。我々はまず Faster R-CNN [10] による物体検出を行い、複数料理画像から個々の料理を検出し、さらに検出された料理に対してカロリー量推定を行う。このカロリー量推定では、食事画像からのカロリー量推定 [1] を用い、検出された料理のクロップ画像からカロリー量を推定する。また、本研究では実際の食事で使用することを考慮し、モバイル実装も行う。このモバイル実装では既存のフレームワークで学習したモデルを iOS 用にコンバートすることで推論処理を iOS 上で行う。実験ではカロリー量推定が端末上でリアルタイムに行われることを確認した。

キーワード カロリー量推定, アプリケーション, 深層学習

## 1. はじめに

近年、日々の食事のカロリー量を記録することのできるアプリケーションが数多く公開されているが、これらのカロリー量の推定には、ユーザー入力による情報が必要であったり、栄養士を雇ったりと人手のかかるものとなっている。また、中には画像認識技術により料理の写真からその料理名の候補を自動で提案するものまで存在するが、単品料理の画像にのみ対応している場合が多く、複数の料理が並ぶ場面では、ユーザーは1品ずつ写真を撮るか、画像から手作業で単品料理の画像を切り出す必要があり、手間がかかる。

一方、画像認識分野では CNN の登場により飛躍的に精度が向上し、クラス分類や物体検出を高精度に行うことが可能となっており、既にこのような画像認識技術を用いたアプリケーションは数多く存在する。

そこで本研究では、CNN を用いた画像認識手法を用い、複数料理画像の同時カロリー量推定を行い、さらにそのモバイル実装を行う。我々はまず複数料理画像から個々の料理を検出し、検出された料理のカロリー量を推定することで、複数料理画像中の複数の料理のカロリー量を同時に推定する。また、このカロリー量推定の処理をモバイル OS 上で行う。

本研究では画像認識分野のタスクの1つである物体検出により個々の料理を検出する。物体検出は画像に含まれる対象物体の矩形領域とカテゴリを推定する技術であり、これにより画像中の個々の料理のカテゴリとバウンディングボックスを推定

する。本研究では CNN を用いた物体検出ネットワークである Faster R-CNN [10] により複数料理画像からの単品料理の検出を行う。Faster R-CNN は、従来研究と比較して高精度かつ高速な物体検出を実現し、後に多くの研究に応用されている。本研究ではこの Faster R-CNN より複数料理画像からの料理の検出を行い、さらに検出された単品料理を切り出した画像からカロリー量推定を行うことで複数料理画像の同時カロリー量推定を実現する。

會下ら [1] は CNN を用い料理画像から直接カロリー量を推定するネットワークを構築した。料理の種類ごとに固定されたカロリー量を使用する方法と異なり、この手法では食材や調理手順の違いによる料理の見た目の違いを考慮したカロリー量推定が行われることが期待でき、例えば同一カテゴリの料理だとしても食材の違いによる見た目の違いに応じて異なるカロリー量を出力する。ただし、このネットワークの入力は単品料理の画像にのみ対応しており、複数料理画像から個々の料理のカロリー量を推定することはできない。そこで本研究では、物体検出により事前に単品料理を検出し、単品料理の矩形領域をカロリー量推定ネットワークの入力として使用する。また、このネットワークの出力のカロリー量の値は、1人分のカロリー量の値となっている。したがって画像中の料理の量が何人前であっても、その料理の1人分の量に対応するカロリー量を出力する。

また本研究ではこの複数品食事画像の同時カロリー推定のモバイル実装を行う。一般的に CNN の学習は膨大な計算量を必要とするため GPU の使用が必須であるが、学習済みモデルを

用いた推論処理にはそこまで多くの計算資源は必要ではなく iPhone 上での利用が可能となっている。実際、これまでに我々は高速化の工夫及び省メモリ化を行うことで深層学習を用いたリアルタイム画像認識システムのモバイル上での動作を実現している。本研究では既存のフレームワークで学習したモデルを iOS 上で使えるように変換し、推論部分を独自に実装する。

まとめると、本研究では複数料理画像に対して Faster R-CNN [1] を用い、単品料理の検出を行う。さらにカロリー量推定と組み合わせることで複数料理画像からの同時カロリー量推定を行う。また、そのモバイル実装を行う。

## 2. 関連研究

現在では様々な自動カロリー量推定システムが提案されている。宮崎ら [4] は食事画像からカロリー量を直接推定している。色ヒストグラムや SURF などの低レベル特徴量に基づいて、データベース上の類似画像を検索し、特徴量ごとに類似度の高い上位  $n$  枚のカロリー量の平均値を計算し、それらの値から最終的にカロリー量を推定している。データセットには Web サービスである FoodLog に投稿された食事画像 6512 枚を使用している。このデータセットには複数料理画像も含まれており、各料理画像には 1 人分のカロリー量がアノテーションされている。したがってこの手法では料理の量に応じたカロリー量を推定することはできない。本研究のカロリー量もこれと同様で食事画像からカロリー量を直接推定するものである。

岡元らはモバイル上でのカロリー量推定システム Grillcam [7] を提案している。このシステムはまず、食事シーンを撮影し、口と箸を検出することでその 2 点が近づいた瞬間の画像を切り出し、画像認識を行うことで料理カテゴリを推定し、カロリー量を記録している。食事シーンを動画に撮影しているため、食べた量が不明瞭な焼き焼き肉や鍋といった料理にも適用可能である。

また、岡元ら [6] は食品と一緒に大きさが既知の基準物体を撮影することで食品の体積を推定し、高精度のカロリー量推定を実現した。この手法ではまず、基準物体と食品と一緒に撮影し、基準物体と食品の領域を比較することで食品の大きさを推定しカロリー量を計算する。食品の領域の抽出では、まずエッジにより背景から皿領域を検出し、その皿領域に対して  $k$ -means により色情報に基づく領域分割を行い、最終的に GrabCut により皿領域から食品領域を推定する。

Pouladzadeh ら [8] も食品とユーザーの親指を同時に撮影することで指の大きさと比較を行い食品の大きさを求め、カロリー量を推定するシステムを提案している。しかし指の出し方や角度、映り方などによっては誤差が生じてしまう可能性がある。

Myers ら [5] はモバイル上で動作するカロリー量推定システムを提案しており、食事/非食事の認識、複数品目の認識、深度推定、領域分割などの複数のタスクを CNN により行い、カロリー量を推定している。まず、食事/非食事認識により画像中に食品が存在するかを判定し、その後マルチラベル認識により画像中の複数の食品を認識する。次に深度推定と領域分割を行い、オブジェクトの三次元構造と食品の領域を抽出し、これらの情

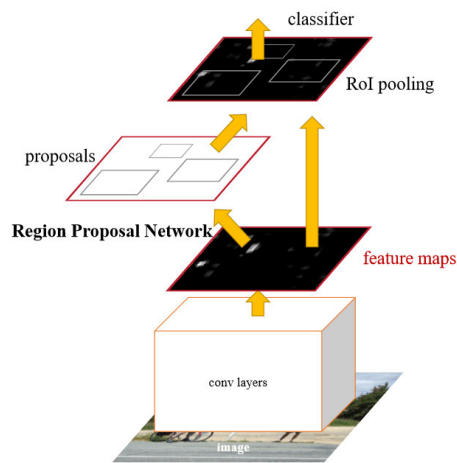


図 1 Faster R-CNN のネットワークの概略図 ([11]).

報を統合して食品の量を推定する。最後に食事カテゴリや量の情報から食品のカロリー量を推定している。ただしこの研究では、カロリー量情報付きのデータセットが不足し、十分に評価が行われていない。

## 3. 手法

ここでは複数料理画像の同時カロリー量推定手法とその実装法について説明する。

### 3.1 Faster R-CNN

Faster R-CNN [10] は物体検出タスクにおいて最新研究の基盤となっている手法であり、二つのモジュールで構成される。

1 つ目のモジュールは、候補領域を生成する Region Proposal Network (RPN) である。この RPN は 1 層の全結合層 (VGG16 [13] では 512 次元) と、それに続いて枝分かれする reg layer と cls layer で構成される。RPN の入力、特徴マップ (VGG16 では最後の畳み込み層の出力) 上の  $3 \times 3$  の spatial window であり、出力は後述する anchor の個数分のバウンディングボックスとそれに対応するスコアである。この出力のスコアは各バウンディングボックスにおいて物体が存在する確率を表している。実際には RPN は、 $3 \times 3$  のカーネルを持つ畳み込み層と、それに続いて枝分かれする  $1 \times 1$  のカーネルを持つ畳み込み層により実装される。また、RPN では物体の形状に柔軟に対応するために 9 個の anchor を導入しており、この anchor は 3 スケール ( $128^2$ ,  $256^2$ ,  $512^2$  ピクセル) と 3 アスペクト比 (2: 1, 1: 1, 1: 2) の形状をもつ。そして RPN では各 spatial window において、anchor を基準とする 9 個の候補領域を生成する。

2 つ目のモジュールは、RPN により生成された候補領域を洗練する Fast R-CNN detector [2] である。この Fast R-CNN detector の入力、RPN により生成された候補領域であり、出力はバウンディングボックスとクラス確率である。このバウンディングボックスはクラスごとに推定される。このように Faster R-CNN は 2 つのモジュールにより構成され、畳み込み層により実装される。重要なことは RPN と Fast R-CNN detector の

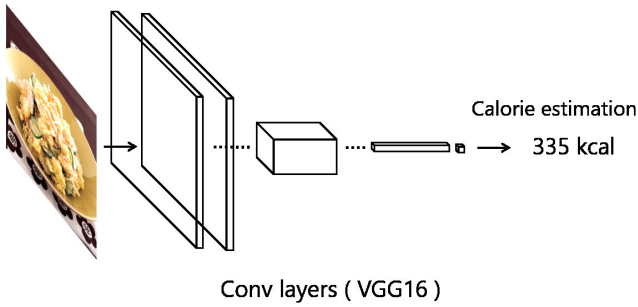


図 2 カロリー量推定のためのネットワークの概略図 ([1]).

両方のネットワークは畳み込み層を共有しており、システム全体は単一のネットワークであるということである (図 1). 入力画像は一度だけ畳み込み層を通過し、RPN が生成する候補領域が Fast R-CNN detector により洗練される. 結果的に Faster R-CNN は高精度かつ高速な物体検出を実現した.

### 3.2 食事画像からのカロリー量推定

本研究では食事画像からのカロリー量推定 [1] により, Faster R-CNN により検出された単品料理のクロープ画像からカロリー量を推定する. このネットワークの入力は単品料理の画像に限られ, また, 出力のカロリー量は料理画像中の料理の量に関係なく 1 人分の量に対応するカロリー量を出力する. ここからは [1] に従い, 本研究で使用するネットワークについて説明する. 我々は [1] のカロリー量付き食事画像データセットを用い, 図 2 のネットワークを学習する. このネットワークは VGG16 [13] に基づいており, 出力層がカロリー量を出力する単一のユニットで構成され, 1 人分の量に対するカロリー量を出力する. このような回帰問題においては, 一般的に損失関数として 2 乗和誤差が用いられるが, [1] では次のような損失関数  $L_{cal}$  を使用している. 絶対誤差を  $L_{ab}$ , 相対誤差を  $L_{re}$  とすると, カロリー量推定タスクの損失関数  $L_{cal}$  は下のように定義される.

$$L_{cal} = \lambda_{re} L_{re} + \lambda_{ab} L_{ab} \quad (1)$$

ただし  $\lambda$  は各損失にかかる重みである. 絶対誤差は推定値と正解値の差の絶対値であり, 相対誤差は絶対誤差と正解値の比である. ある画像  $x$  を入力したときの推定値を  $y$ ,  $y$  に対する正解値を  $g$  とすると, 絶対誤差  $L_{ab}$  と相対誤差  $L_{re}$  は下のように定義される.

$$L_{ab} = |y - g| \quad (2)$$

$$L_{re} = \frac{|y - g|}{g} \quad (3)$$

### 3.3 複数食事画像からの同時カロリー量推定

本研究ではまず, Faster R-CNN [10] により複数料理画像中の単品料理を検出し, そのバウンディングボックスに基づき画像を切り出すことで単品料理画像を生成し, これをカロリー量推定ネットワーク [1] に入力として与えることで, 各単品料理のカロリー量が得られる. これにより各単品料理のカロリー量が 1 枚の複数料理画像から推定される.

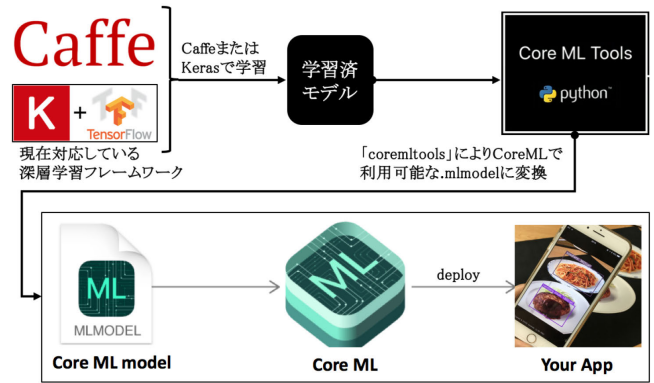


図 3 CoreML を用いた iOS アプリ deploy フロー.

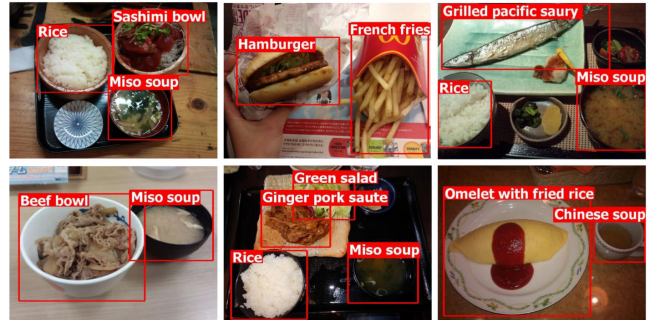


図 4 UEC FOOD-100 [3] に含まれる複数料理画像の例.

### 3.4 実装法

本研究では「CoreML」を用いて複数料理画像のカロリー量推定のモバイル実装を行う. CoreML (ML: Machine Learning) とは iOS11 から追加された機械学習 API 群のことであり, 学習済みモデルを iOS 上で推論できるようにモデルを変換して利用することができる. また, 学習済みモデルを CoreML 用に変換するコンバータ「coremltools」も同時に公開されている. CoreML を用いたアプリ実装フローを図に示す. CoreML が現在サポートしている深層学習フレームワークは Caffe 及び Keras (TensorFlow backend) の 2 つであり, 学習したモデルを coremltools により CoreML 用に変換して利用する. 学習済みモデルを変換するコンバータの存在がする点や, CNN の各レイヤー実装などは CoreML により隠蔽して利用できる点で, iOS10 と比べて容易に深層学習をアプリに実装できるようになっている. 本研究ではフレームワークは Keras を利用し, 「coremltools」で.mlmodel に変換後のモデルを CoreML で利用している.

## 4. データセット

本研究では複数料理画像からの単品料理の検出の学習と評価のためにバウンディングボックス付きの複数料理画像を含む UEC Food-100 [3] を使用する. また, 複数料理画像の同時カロリー量推定の評価のために合計カロリー量がアノテーションされた学校給食画像を使用する.

### 4.1 UEC Food-100

UEC FOOD-100 [1] は料理 100 カテゴリに関する食事画像

データセットであり、複数料理画像も含まれている。単品料理画像は各料理カテゴリ 100 枚以上あり、合計 11566 枚ある。それに加え、複数料理画像は 1174 枚ある。合わせて 12740 枚のすべての画像にバウンディングボックスが付与されている。このデータセットは Faster R-CNN の学習と評価に使用される。図 5 に UEC FOOD-100 に含まれる複数料理画像の例を示す。

#### 4.2 合計カロリー量付き学校給食画像

学校給食画像は Web 上の給食ブログから収集された。この給食ブログでは一日ごとに給食画像と共に合計カロリー量の値が掲載される。合計カロリー量とは、個々の料理のカロリー量の合計値であり、個々の料理のカロリー量は不明である。このデータセットは合計 690 枚の合計カロリー量付き学校給食画像を有し、複数料理画像からの同時カロリー量推定の評価に使用される。

### 5. 実 験

まず UEC Food-100 [3] を用い、複数料理画像からの単品料理の検出の実験を行う。次に合計カロリー量付き学校給食画像を用い、複数食事画像の同時カロリー量推定の評価を行う。このとき Faster R-CNN [10] の学習にはバウンディングボックスがアノテーションされているがカロリー量がアノテーションされていない学校給食画像を使用し、カロリー量推定ネットワーク [1] の学習にはカロリー量付き食事画像データセット [1] を使用する。最後に複数料理画像のカロリー量推定の実装について述べる。

#### 5.1 複数料理画像から料理の検出

本実験では UECFood-100 を用いて Faster R-CNN を学習し評価を行う。ネットワークの学習ではパラメータの初期値として VGG16 [13] の事前学習済みモデルを使用する。最適化手法として SGD を使用し、momentum 値を 0.9、weight decay 値を 0.0005 とする。また、バッチサイズを 1 とし学習率  $10^{-3}$  において 50k 回反復し、その後、学習率  $10^{-4}$  において 20k 回反復する。RPN と Fast R-CNN detector の 2 つのモジュールを学習するためにネットワーク全体を Approximate joint training [11] により学習する。この Approximate joint training では、RPN により生成される候補領域は事前計算されたものとして扱われ、RPN の損失と Fast R-CNN の損失が独立に誤差逆伝搬され、共有された畳み込み層において 2 つの損失が結合されて誤差逆伝搬される。評価指標として PASCAL VOC detection task<sup>(注1)</sup> の標準的な評価指標である mean Average Precision(mAP) を使用する。

学習には単品料理画像を使用し、テストには複数料理画像を使用する。また、下田ら [12] の手法との比較を行う。下田らは誤差逆伝搬により得られるサリエンスマップに基づき領域分割を行い料理の検出を行った。[12] に従い、100 カテゴリ、53 カテゴリ (評価用サンプル数  $\geq 10$  のカテゴリ)、11 カテゴリ (評価用サンプル数  $\geq 50$  のカテゴリ) それぞれについての mAP を

表 1 UEC Food-100 [3] に含まれる複数料理画像からの料理検出結果 (料理カテゴリごとに AP を計算)。100 カテゴリ (全体)、53 カテゴリ (評価用データ数  $\geq 10$  のカテゴリ)、11 カテゴリ (評価用データ数  $\geq 50$  のカテゴリ)。

UEC Food-100 mAP(%)	100 カテゴリ (all)	53 カテゴリ (#item $\geq 10$ )	11 カテゴリ (#item $\geq 50$ )
R-CNN (in [12])	26.0	21.8	25.7
[12]'s method (BP)	49.9	55.3	55.4
Faster R-CNN	42.0	46.3	57.9

表 2 UEC Food-100 [3] に含まれる複数料理画像からの料理検出結果 (料理カテゴリを無視し、評価用データ全体で AP を計算)。100 カテゴリ (全体)、53 カテゴリ (評価用データ数  $\geq 10$  のカテゴリ)、11 カテゴリ (評価用データ数  $\geq 50$  のカテゴリ)。

UEC Food-100 AP(%)	100 カテゴリ (all)	53 カテゴリ (#item $\geq 10$ )	11 カテゴリ (#item $\geq 50$ )
[12]'s method (BP)	57.3	58.0	58.8
Faster R-CNN	57.7	59.2	67.0



図 5 UEC Food-100 [3] に含まれる複数料理画像からの料理検出例。

表 1 に示す。また、図 5 に検出結果の例を示す。我々の手法は 11 カテゴリの mAP において [12] の精度を超え、特に日本食に頻繁に現れるようなサンプル数の多い“ごはん”や“味噌汁”において高い精度での検出が見られた。[12] と比較すると、“ごはん”と“味噌汁”に関してそれぞれ、60.0%から 90.2%、68.3%から 80.2%の向上が得られた。さらに [12] に従い、料理カテゴリを無視し、全体で AP(%) を計算した結果を表 2 に示す。サンプル数の多い“ごはん”や“味噌汁”が強調されるため、精度の向上が見られた。

#### 5.2 複数料理画像の同時カロリー量推定

本実験では Faster R-CNN とカロリー量推定ネットワークを組み合わせ、複数料理画像の同時カロリー量推定を行う。

複数料理画像からの単品料理の検出にはバウンディングボックスがアノテーションされた学校給食画像を学習した Faster R-CNN を用いる。これは評価に用いる画像が学校給食画像であるためである。このバウンディングボックス付き学校給食画像は日本の給食の料理 21 カテゴリに関する食事画像データセットである。合計 4877 枚の学校給食画像はすべて複数料理画像であり、バウンディングボックスが付与されている。これらの画像は、Web 上の給食センターが更新している給食ブログから

(注1) : <http://host.robots.ox.ac.uk/pascal/VOC/voc2012/index.html>

収集された。

また、単品料理画像からのカロリー量推定には、[1]において収集されたカロリー量付き食事画像で学習したカロリー量推定ネットワークを使用する。[1]で構築されたカロリー量付き食事画像データセットは、Web上のレシピ情報サイトから大量に収集されたものであったが、[1]では料理カテゴリ分類を行うために料理15カテゴリに含まれる画像のみを使用していた。この料理15カテゴリは本実験の評価に使用する学校給食画像中の料理にはほとんど対応しておらず、この料理15カテゴリのデータセットで学習したカロリー量推定ネットワークを学校給食画像に用いることは難しいと考えられる。そこで本研究では、料理カテゴリを限定せず、[1]で収集された大量のカロリー量付き食事画像を使用し、カロリー量推定ネットワークの学習を行う。ノイズ除去として我々はまず $256 \times 256$ 以下の画像を取り除いた。また、[1]のカロリー量推定ネットワークの入力は単品料理の画像にのみ対応しているため、UEC Food-100により学習済みのFaster R-CNNにより複数の料理が検出された画像の除去も行った。最終的に我々は55,020枚のカロリー量付き食事画像を使用して図2に示すカロリー量推定ネットワークの学習を行った。パラメータなどの値は主に[1]に従う。最適化手法としてSGDを使用し、Momentum値は0.9とし、バッチサイズは8とする。学習率 $10^{-3}$ において150k回反復し、その後 $10^{-4}$ において50k回反復する。

こうしてそれぞれ学習されたFaster R-CNNとカロリー量推定ネットワークを組み合わせることで複数料理画像の同時カロリー量推定を行う。まずFaster R-CNNにより複数料理画像中の単品料理のバウンディングボックスと料理カテゴリを推定する。次に各単品料理のバウンディングボックスに基づき単品料理画像を切り出し、これをカロリー量推定ネットワークに入力として与えることで、各単品料理のカロリー量が得られる。最終的に推定された各料理のカロリー量から合計カロリー量を計算する。評価には合計カロリー量付き学校給食画像データセットを使用する。評価では、単品料理ごとに推定されたカロリー量の正確さではなく、それらの合計カロリー量の正確さの評価を行う。ただしこの実験ではFaster R-CNNにより“牛乳”と分類された料理に関しては134kcalと固定する。カロリー量推定ネットワークの学習データに学校給食の“牛乳”が含まれていることは考えづらく、また、サイズが定まっており、カロリー値に大きな変動はないと考えられるためである。

表3に複数料理画像の同時カロリー量推定の結果を示す。比較のために[1]の料理15カテゴリを対象とする単品料理画像に対するカロリー量推定の結果を示すが、使用されるデータセットが異なるため参考比較である。[1]に基づき、カロリー量推定の評価指標として相対誤差、絶対誤差、相対誤差20%以内の推定値の割合、相対誤差40%以内の推定値の割合を用いた。絶対誤差は推定値と正解値の差の絶対値であり、相対誤差は正解値に対する絶対誤差の割合である。表3からわかるように、相対誤差や相対誤差20%/40%以内の推定値の割合に関して、[1]の料理15カテゴリの単品料理画像に対するカロリー量推定の結

表3 複数料理画像からの同時カロリー量推定結果。単品料理カロリー量推定[1]は料理15カテゴリのみを対象としており、使用したデータセットが異なるため参考値である。

	相対誤差 (%)	絶対誤差 (kcal)	誤差 20% (%)	誤差 40% (%)
単品料理カロリー量推定 [1]	30.2	105.7	43	76
複数料理合計カロリー量推定	21.4	136.8	53.0	85.1



図6 複数料理画像からの同時カロリー量推定結果の例。バウンディングボックス中の数字は推定された各料理のカロリー量 (kcal) である。ESは推定されたカロリー量の合計値 (kcal) であり、GTは合計カロリー値の正解値 (kcal) である。

果を上回っている。本手法では合計カロリー量を計算しているため、各単品料理の推定カロリー量の誤差が緩和されているのではないかと考えられる。

図6に複数料理画像の同時カロリー量推定結果の例を示す。図6のように各単品料理のカロリー量が推定され、それらの合計値が計算される。同一の料理を複数回検出することや、料理が存在しない領域を検出してしまふような過剰な検出により、計算された合計カロリー量が正解値よりも高い値になってしまう例が多く見られた。密集する同一カテゴリのバウンディングボックスの統合や、非料理画像 (0kcal) の学習によりこれに対処することができるのではないかと考えられる。

### 5.3 複数料理画像の同時カロリー量推定のモバイル実装

本実験では料理検出とカロリー推定を組合せた複数料理画像の同時カロリー量推定システムのモバイル実装を行う。本アプリの処理イメージを図7に示し、あり、動作例を図8に示す。カロリー値を推定するワークフローは以下になる。

- (1) YOLOv2 [9] を用いて食事検出をする。
- (2) 検出された各単品領域のバウンディングボックス情報

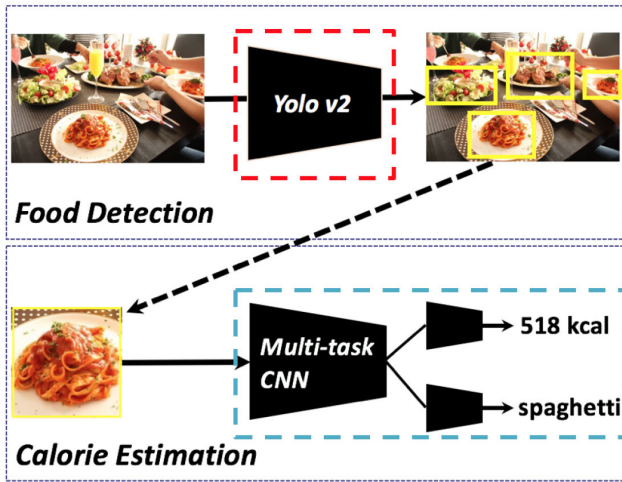


図 7 アプリの処理イメージ

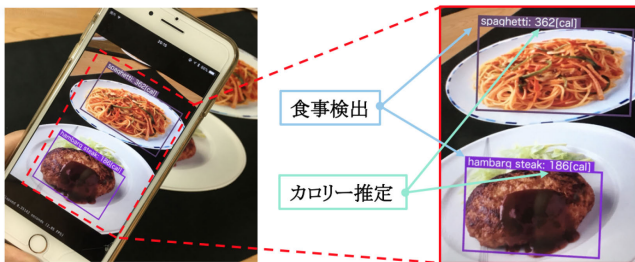


図 8 アプリ使用例

から画像をクロップする。

(3) クロップした各単品画像をカロリー値推定ネットワークに入力とする。

料理検出器としては高速な検出が可能な最新の物体検出ネットワークである YOLOv2 を使用する。料理検出器の学習では、UEC Food-100 [3] を用いて、YOLOv2 を fine-tune する。

カロリー値推定ネットワークの学習では、カロリー量付き食事画像データセット [1] を用いて、カロリー量と料理カテゴリを同時に推定する Multi-task CNN [1] を fine-tune する。ただしここでは Multi-task CNN の元となるネットワークとして Inception-v3 を使用する。フレームワークは Keras を利用し、「coremltools」で mlmodel に変換後のモデルを CoreML で利用している。モバイル OS 上において複数の料理のカロリー量を同時にリアルタイムに推定することが可能であることを示した。

## 6. おわりに

本研究ではまず Faster R-CNN [10] による複数料理画像からの単品料理の検出を行い、これとカロリー量推定 [1] を組み合わせることで、複数料理画像の同時カロリー量推定を行った。また、モバイル実装を行い、複数料理画像の同時カロリー量推定がモバイル OS 上においてリアルタイムに実行可能であることを確認した。今後の課題として、物体検出とカロリー量推定を単一のネットワークで行い同時学習することや、料理の量に応じたカロリー量推定を行うことなどを考えている。

- [1] T. Ege and K. Yanai. Simultaneous estimation of food categories and calories with multi-task cnn. In *Proc. of IAPR International Conference on Machine Vision Applications(MVA)*, 2017.
- [2] R. Girshick. Fast R-CNN. In *Proc. of IEEE International Conference on Computer Vision*, 2015.
- [3] Y. Matsuda, H. Hajime, and K. Yanai. Recognition of multiple-food images by detecting candidate regions. In *Proc. of IEEE International Conference on Multimedia and Expo*, 2012.
- [4] T. Miyazaki, G. Chaminda, D. Silva, and K. Aizawa. Image - based calorie content estimation for dietary assessment. In *Proc. of IEEE ISM Workshop on Multimedia for Cooking and Eating Activities*, 2011.
- [5] A. Myers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and P. K. Murphy. Im2calories: towards an automated mobile vision food diary. In *Proc. of IEEE International Conference on Computer Vision*, 2015.
- [6] K. Okamoto and K. Yanai. An automatic calorie estimation system of food images on a smartphone. In *Proc. of ACM MM Workshop on Multimedia Assisted Dietary Management*, 2016.
- [7] K. Okamoto and K. Yanai. Grillcam: A real-time eating action recognition system. In *International Conference on Multimedia Modeling*, 2016.
- [8] P. Pouladzadeh, S. Shirmohammadi, and R. Almaghrabi. Measuring calorie and nutrition from food image. In *IEEE Transactions on Instrumentation and Measurement*, pages 1947–1956, 2014.
- [9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You Only Look Once: Unified, real-time object detection. In *Proc. of IEEE Computer Vision and Pattern Recognition*, 2016.
- [10] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015.
- [11] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.
- [12] W. Shimoda and K. Yanai. CNN-based food image segmentation without pixel-wise annotation. In *Proc. of IAPR International Conference on Image Analysis and Processing*, 2015.
- [13] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *arXiv preprint arXiv:1409.1556*, 2014.