# Mobile Food Calorie Estimation Using Smartphone LiDAR Sensor

Haruto Fujita[1[0009−0001−0443−4055]] and Keiji Yanai[1[0000−0002−0431−183X]]

The University of Electro-Communications, Tokyo, Japan
fujita-h@mm.inf.uec.ac.jp, yanai@cs.uec.ac.jp

**Abstract.** In recent years, meal calorie management has played an important role in health maintenance and dieting. Particularly, interest in recording meal content and calories has increased, and various apps that support this have been provided. However, many apps that focus too much on convenience suffer from reduced accuracy or have usability issues. For example, some apps require users to manually input the type and amount of food one by one, requiring prior knowledge about food, while others automatically recognize food types from food images but have fixed calorie values regardless of quantity.
Therefore, this research proposes a system called "LiDARCalorieCam" that estimates meal calorie amounts in real-time using the LiDAR sensor installed in iPhones. This system takes images of meals and obtains the three-dimensional shape of the food using depth sensors. Based on the three-dimensional shape, it estimates food volume and calculates calories based on the volume. Building on existing research, we expanded the range of supported meal categories and improved calorie estimation accuracy, achieving more reliable calorie estimation.

**Keywords:** food calorie estimation · LiDAR sensor · mobile application · depth sensing

## 1 Introduction

With the increasing prevalence of obesity and lifestyle-related diseases, daily dietary management has become increasingly important. In particular, food calorie estimation using smartphones is highly practical, as it can be conveniently used by end-users. However, conventional approaches often require reference objects or suffer from insufficient accuracy when relying solely on single RGB images.

Traditional deep learning methods have taken approaches to estimate food types and quantities, and calculate calories. However, accurate calorie estimation requires considering food volume, which has remained a challenge. Particularly, it has been difficult to obtain three-dimensional information such as food height and depth from single RGB images alone, limiting volume estimation accuracy.

This research aims to solve this problem by applying LiDAR (Light Detection and Ranging) technology to capture food's three-dimensional shape with high accuracy. The contributions of this work are as follows:

1. We leverage the built-in LiDAR sensor of the iPhone to achieve accurate volume estimation without requiring reference objects.
2. We enable real-time food calorie estimation directly on a single mobile device.
3. We quantitatively demonstrate the effectiveness of LiDAR by comparing it with DepthCalorieCam, which relies solely on RGB-based depth estimation.

In its current version, the system supports 10 specific food categories for which volume-to-calorie conversion formulas have been developed.

## 2 Related Work

Calorie estimation using meal images is an important research area in health management and nutrition guidance. In this field, methods combining meal category classification and volume estimation are common, with various approaches being proposed.

### 2.1 Basic Research in Calorie Estimation

Akpa *et al.* [1] and Tanno *et al.* [13] have proposed smartphone-based calorie estimation methods that use familiar objects as reference points to make food quantity and calorie estimation easier and more accurate. Specifically, by including commonly used items such as chopsticks or rice grains in meal photos, these methods work for estimation of relative size and volume without special equipment. While this offers high convenience, users must consciously place reference objects each time, and accurately measuring complex food shapes remains difficult.

In contrast, Fang *et al.* [6] suggested a method that estimates volume based on geometric models of food, generated from single-view images, without requiring reference objects. By modeling the shapes of foods in images mathematically, this approach enables more convenient estimation, though accuracy depends on the diversity of food shapes.

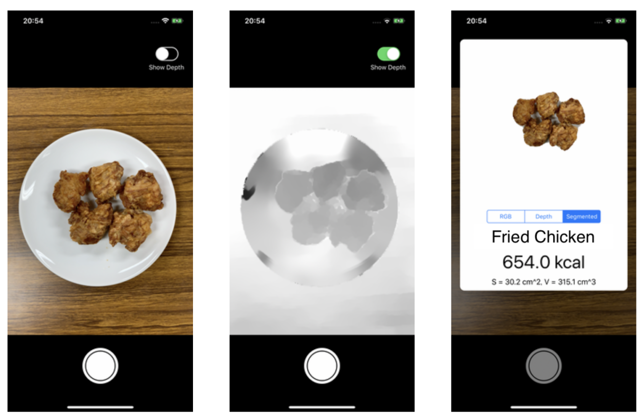
Thames *et al.* [14] developed the "Nutrition5k" dataset, which contains images of over 5,000 foods, labeled with types, calories, and nutrients. While this dataset advances calorie estimation research, it mainly covers Western foods and often contains mixed dishes, which complicates assigning uniform calorie values to specific food categories.

Okamoto *et al.* [8] proposed "CalorieCam," a system that calculates the area of food using reference objects of known size, such as credit cards, photographed together with the meal. This makes area measurement with a smartphone simple and accessible, but always requires reference objects. Additionally, estimating calories based solely on area is limited, especially when food height and depth are important for calorie estimation accuracy.

### 2.2 Application Development Not Requiring Reference Objects

Tanno *et al.* [13] developed “AR DeepCalorieCam V2” that does not require reference objects and combines AR technology. This system estimates food size and calculates calories using smartphone cameras and AR (Augmented Reality) technology. By utilizing AR technology, more real-time size estimation becomes possible, eliminating manual measurement. Users must move the device while placing anchors in AR space.

Ando *et al.* [2] developed “DepthCalorieCam,” a system that estimates meal calories using depth images with dual cameras and deep learning on iPhones. Depth value calculation is performed using disparity between two cameras: telephoto lens and wide-angle lens on iPhones. The calorie estimation flow in DepthCalorieCam involves photographing food from directly above, extracting meal regions using U-Net [11], estimating volume from actual dimensions, and estimating calorie amounts using Xception [4]. Figure 1 shows an exapmle of the DepthCalorieCam interface.




(a) Shooting screen (RGB) (b) Shooting screen (Depth) (c) Result screen

**Fig. 1.** “DepthCalorieCam” UI screenshots.

Like AR DeepCalorieCam V2, this is a method that doesn’t require reference objects, but by obtaining depth images without using AR technology, it becomes possible to estimate meal calories more simply. In this paper, we used DepthCalorieCam as a baseline that does not rely on the LiDAR sensor. DepthCalorieCam leverages depth estimated from RGB images, representing a typical smartphone environment without LiDAR. This allows us to quantitatively evaluate the performance difference between LiDAR-based and non-LiDAR approaches. However, problems include limited meal categories to only three types, lacking practical applications, and restrictions on usage due to photography positions being limited to directly above.

### 2.3 Positioning of This Research and LiDAR Technology Application

Existing methods have achieved certain results, but issues remain—such as the inconvenience of preparing reference objects and the difficulty of handling complex or irregularly shaped foods. Since many of these methods depend on single images or 2D information, accurately estimating the 3D volume of food can be challenging.

To address these limitations, this research proposes the use of LiDAR technology, which measures distances using laser light and can obtain detailed 3D shape data of objects. By leveraging LiDAR, it becomes possible to measure food volume accurately and easily, without the need for reference objects.

A recent example is "NutritionVerse" [12], which utilized the LiDAR sensor of an iPhone 13 Pro Max to capture food shapes, creating an 889-image dataset from 251 dishes and using it to train deep learning models for calorie estimation.

Building on these previous studies, this research aims to develop a calorie estimation system for meals using LiDAR technology on mobile devices, and to evaluate its effectiveness in practical scenarios. Regarding prior work, NutritionVerse focuses on food categories that differ significantly from those in our dataset. Therefore, a direct comparison is not feasible. Instead, we emphasize evaluation on food categories commonly consumed in Japan.

## 3 System Architecture

This system consists of a client application on iOS devices and server-side volume calculation processing as shown in Figure 2. The client application performs depth estimation using ARKit and executes food region segmentation and food category classification using Core ML. Based on this information, three-dimensional point clouds are generated and sent to the server side. The server side calculates volume using multiple methods for received point clouds and returns estimation results to the client.

The system follows a hybrid client-server architecture, performing lightweight processing on the mobile device while offloading computationally intensive volume calculations to the server.

### 3.1 System Processing Flow

The system's processing flow uses RGB images and depth information obtained from smartphones to extract food regions and generate three-dimensional point clouds. Then, the volume is estimated on the server side, and the calorie amounts are calculated by matching with food category information. Figure 2 shows the overall system processing flow.

The processing steps are as follows:

1. **Depth Image Photography:** Use ARKit to photograph food with smartphones and obtain RGB images and depth maps

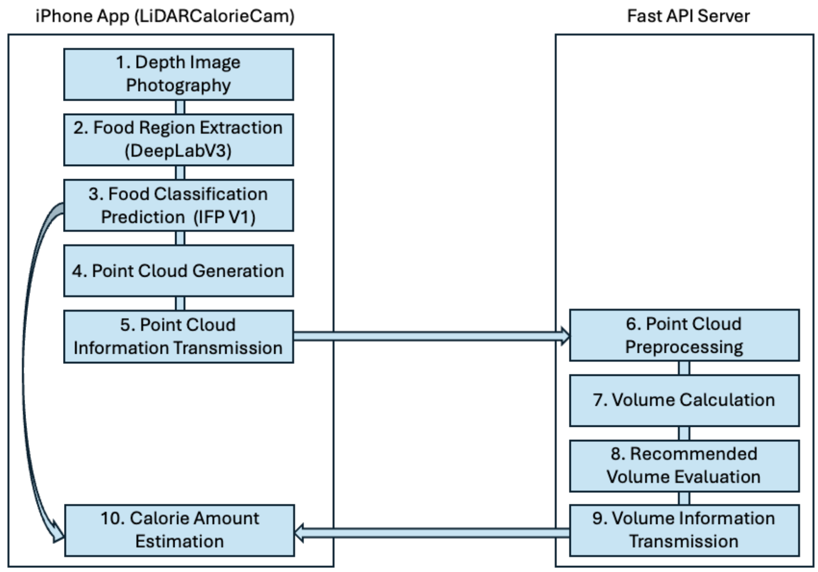



**Fig. 2.** System processing flow.

2. **Food Region Extraction:** Segment the food portions using a segmentation model (DeepLabV3 [3]) and remove background
3. **Food Classification Prediction:** Estimate food categories using classification model (Image Print Feature V1)
4. **Point Cloud Generation:** Use intrinsic camera parameters and depth maps from LiDAR to obtain three-dimensional coordinates
5. **Point Cloud Information Transmission:** Send generated point clouds to server and make analysis requests
6. **Point Cloud Preprocessing:** Perform outlier removal and quality checks
7. **Volume Calculation:** Calculate volume using multiple methods on server side
8. **Recommended Volume Evaluation:** Calculate confidence scores and select optimal volume
9. **Volume Information Transmission:** Send volume calculation results to client
10. **Calorie Amount Estimation:** Calculate and display calorie amounts based on obtained volume and food category information

### 3.2 Food Region Extraction

**Segmentation** To accurately extract food regions, this research adopted the approach of using DeepLabV3 in Core ML format. For food image segmentation, we employed DeepLabV3 with a ResNet101 backbone, fine-tuned for 20 epochs with a batch size of 8 and a learning rate of 1e-4, using an input image size of 513×513.

The model was trained on the training data from UECFOODPix [9] meal image dataset and evaluated on the test data. While the original UECFOODPix dataset contains 102 food categories, we reclassified them into 21 broader categories to minimize architectural changes to the model. Background is masked to generate basic information for obtaining food-specific three-dimensional point clouds. Figure 3 shows actual segmentation result examples.

(a) Input Image (b) Segmentation Region

**Fig. 3.** An example of a segmentation result.

DeepLabV3 was chosen for its balance between inference speed and accuracy, enabling real-time segmentation on iOS devices with efficient Core ML implementation.

**Food Classification** Images extracted through segmentation are input into Apple's iOS machine learning library Create ML Image Classification model to determine food categories. The model uses a convolutional neural network for extracting 2048-dimensional features from $299 \times 299$ pixel images. A model trained using UEC Food-100 [7] dataset is applied.

### 3.3 Depth Estimation and Point Cloud Generation

**Depth Estimation Using ARKit** This research uses ARKit, an augmented reality development framework for iOS devices, to obtain food's three-dimensional information. ARKit integrates multiple sensor information including camera footage, IMU, and LiDAR sensors to perform device pose estimation and environmental mapping.

ARKit performs spatial recognition through sensor data integration, obtaining depth information from LiDAR sensors in real-time.

Camera internal parameters enable conversion from image coordinate system to real-world coordinate system. Using a pinhole camera model as shown in Figure 4, conversion from image coordinates $(u, v)$ to real-world coordinates $(X, Y, Z)$ is performed:

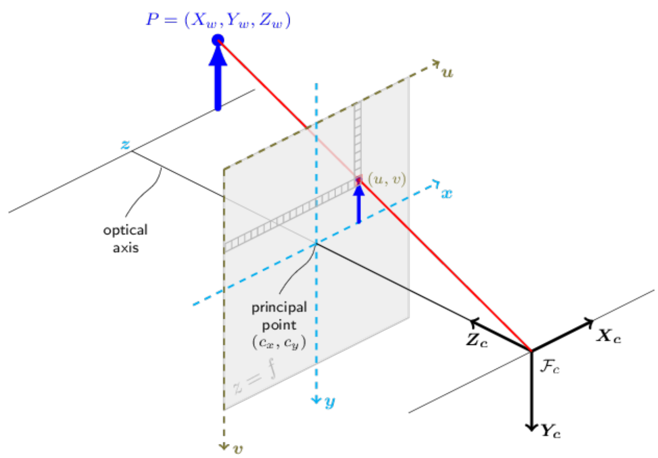



**Fig. 4.** Pinhole camera model. [10]

$$K = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

$$X = \frac{(u - c_x) \cdot d}{f_x}, \quad Y = \frac{(v - c_y) \cdot d}{f_y}, \quad Z = d \tag{2}$$

where $d$ is the depth value from sceneDepth in meter units.

**Point Cloud Generation** Using segmentation masks, only food pixels are extracted and RGB values are combined with coordinate information. Since depth maps (256 × 192) and RGB images (1920 × 1440) have different resolutions, point clouds are created considering scaling. Each point is assigned $(X, Y, Z)$ and $(R, G, B)$ information and sent to the server. Figure 5 shows an example of croquette point cloud data received on the server side.

### 3.4 Volume Calculation

For point clouds received on the server side, outlier detection and removal is performed using DBSCAN [5] with parameters $\epsilon = 5.0$ and minPts $= 5$, chosen to effectively remove noise in point clouds while preserving food shape clustering.

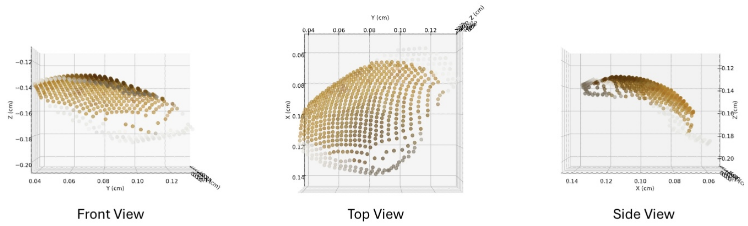



**Fig. 5.** Example of croquette point cloud data.

The system rolls back to original point clouds when more than half the points are lost through outlier removal.

For volume estimation, we employed multiple geometric approaches including Convex Hull, Delaunay triangulation, Alpha Shape, Spline integration, and Open3D mesh-based volume estimation. The outputs of these methods were combined to generate confidence scores and a recommended volume. The mean $\mu$ and standard deviation $\sigma$ of volume values obtained from each method are calculated, and the coefficient of variation $CV = \sigma/\mu$ is obtained. Confidence is defined as confidence $= e^{-CV}$.

For recommended volume selection, mesh-based method results are adopted when confidence is 0.8 or higher, otherwise Convex Hull method results are adopted.

### 3.5 Calorie Estimation

**Volume to Weight Conversion** Regression formulas are created using volume-weight data measured in advance for each food category. Setting the regression formula as $W = aV + b$, where $V$ is volume and $W$ is weight. Parameters $a$ and $b$ are fitted on the training data and evaluated on the test data, with different values for each category. Linear regression was chosen for its simplicity, robustness against overfitting, and its ability to generalize well across different food categories.

**Weight to Calorie Amount Conversion** Calorie density [kcal/g] for each category is obtained from the Standard Tables of Food Composition in Japan provided by the Ministry of Education, Culture, Sports, Science and Technology (MEXT). The calorie amount $\hat{C}$ is calculated from $\hat{C} = D \times \hat{W}$, where $D$ is calorie density and $\hat{W}$ is estimated weight.

## 4 Experiments

In this study, we conducted experiments using the UECFoodPixComplete dataset **??**, covering 21 food categories with approximately 9,000 images for training and

1,000 images for evaluation. The dataset was evenly divided by food categories into training and evaluation subsets. To ensure reproducibility, all results are reported as averages over independent runs.

Figure 6 shows actual application operation examples.

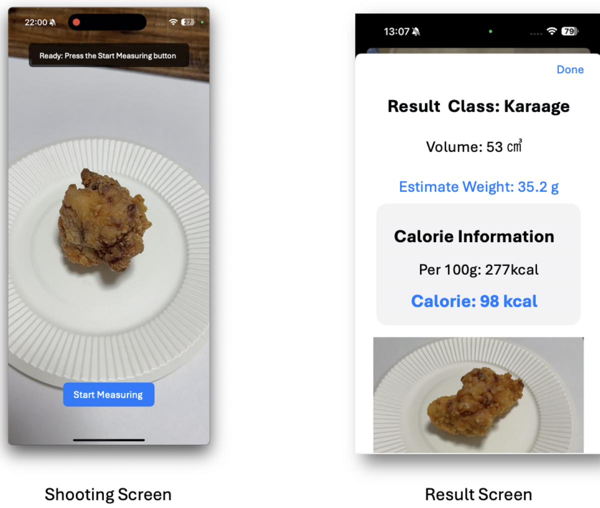



**Fig. 6.** Screenshots of the proposed system: (left) Photography screen, (right) Calorie estimation results.

### 4.1 Volume Estimation Accuracy Evaluation

Multiple food items were photographed obliquely from a distance of about 30cm, and volume was estimated for obtained three-dimensional point clouds. Actual volume measurement used measuring cups and water displacement methods.

Table 1 shows errors between estimated and actual volumes for each food. Since all measurement data exceeded confidence of 0.8, mesh-based volume values were used.

While there were variations due to photography angles and food shapes, few large outliers occurred and results generally fell within acceptable ranges. Figure 7 shows an example volume estimation result using Convex Hull method for karaage (estimated volume: 96cm$^3$).

### 4.2 Weight Estimation Regression Analysis

Relationships between estimated volume ($V$) and actual weight ($W$) are modeled through simple regression analysis. Table 2 shows regression formulas and

**Table 1.** Volume estimation errors by food type.

| **Food** | **MAE** (cm$^3$) | **MAPE** (%) |
|---|---|---|
| Karaage (Japanese fried chicken) | 32.5 | 64.5 |
| Croquette | 58.0 | 31.0 |
| Yakitori (grilled chicken skewers) | 37.4 | 39.5 |
| Hot dog | 28.5 | 8.8 |
| Toast | 61.3 | 13.9 |
| Yakisoba (fried noodles) | 14.4 | 7.8 |
| Potato salad | 14.7 | 12.4 |
| Onigiri (rice balls) | 27.7 | 31.1 |
| Tamagoyaki (rolled omelet) | 3.4 | 14.0 |
| Sautéed vegetables | 17.0 | 13.1 |

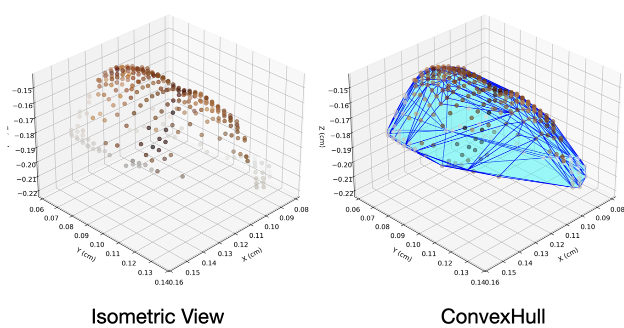



**Fig. 7.** Volume estimation result for “karaage” using Convex Hull method.

determination coefficients $R^2$ for 10 categories. Figure 8 shows the relationship between estimated volume and actual weight for toast, with data from photographing multiple different-sized toasts and regression lines overlaid on plots.

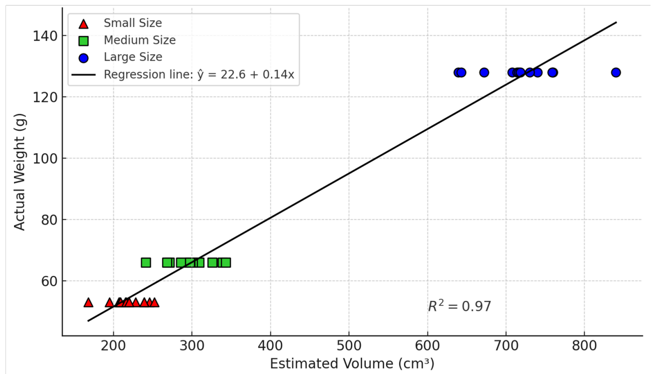



**Fig. 8.** Relationship between estimated volume and actual weight for "toast."

**Table 2.** Regression formulas for volume-to-weight conversion.

| **Food Category** | **Regression Formula** $W = aV + b$ | $R^2$ |
|---|---|---|
| Karaage (Japanese fried chicken) | $W = 0.45V + 11.4$ | 0.95 |
| Croquette | $W = 0.33V + 38.0$ | 0.87 |
| Yakitori (grilled chicken skewers) | $W = 0.36V + 26.0$ | 0.92 |
| Hot dog | $W = 0.17V + 23.81$ | 0.65 |
| Toast | $W = 0.15V + 22.6$ | 0.97 |
| Yakisoba (fried noodles) | $W = 0.44V + 21.35$ | 0.85 |
| Potato salad | $W = 0.77V + 5.2$ | 0.71 |
| Onigiri (rice balls) | $W = 0.49V + 20.7$ | 0.81 |
| Tamagoyaki (rolled omelet) | $W = 0.65V + 4.0$ | 0.62 |
| Sautéed vegetables | $W = 0.29V + 28.0$ | 0.85 |

### 4.3 Calorie Estimation Accuracy Evaluation

We employed both Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE) as evaluation metrics. While MAPE captures relative errors, it can be difficult to interpret for small food items; therefore, MAE is additionally reported.

Weight is estimated from volume, and calorie amounts are calculated based on food composition tables. Tables 3 and 4 show MAE and MAPE for estimated calories by food, compared with existing research DepthCalorieCam. Note that DepthCalorieCam can recognize only two food categories among the ten food categories LiDARCalorieCam can recognize.

**Table 3.** Calorie estimation errors MAE (kcal).

| **Food Category** | **DepthCalorieCam** | **LiDARCalorieCam** |
|---|---|---|
| Karaage (Japanese fried chicken) | 101.29 | 15.88 |
| Croquette | 43.09 | 12.83 |
| Yakitori (grilled chicken skewers) | — | 25.68 |
| Hot dog | — | 12.03 |
| Toast | — | 23.95 |
| Yakisoba (fried noodles) | — | 30.42 |
| Potato salad | — | 16.76 |
| Onigiri (rice balls) | — | 24.19 |
| Tamagoyaki (rolled omelet) | — | 6.01 |
| Sautéed vegetables | — | 11.52 |

**Table 4.** Calorie estimation errors MAPE (%)

| **Food Category** | **DepthCalorieCam** | **LiDARCalorieCam** |
|---|---|---|
| Karaage (Japanese fried chicken) | 52.84 | 7.87 |
| Croquette | 20.11 | 6.10 |
| Yakitori (grilled chicken skewers) | — | 11.50 |
| Hot dog | — | 5.59 |
| Toast | — | 8.64 |
| Yakisoba (fried noodles) | — | 16.25 |
| Potato salad | — | 16.81 |
| Onigiri (rice balls) | — | 12.15 |
| Tamagoyaki (rolled omelet) | — | 10.53 |
| Sautéed vegetables | — | 9.99 |

# 5 Discussion

## 5.1 Relationship Between Volume Estimation Errors and Calorie Estimation Errors

A notable phenomenon was observed in case of "karaage," where despite a large volume estimation MAPE of 64.5%, the calorie estimation error was significantly improved to 7.87%. This improvement can be attributed to the intercept term in the regression formula $W = 0.45V + 11.4$ ($\mathrm{R}^2 = 0.95$), where the constant term of 11.4g effectively compensates for systematic errors in volume estimation. Specifically, the weight increase due to volume overestimation is offset by the intercept term, ultimately improving the final calorie estimation accuracy.

The intercept term $b$ addresses practical measurement challenges including systematic volume underestimation, measurement noise, and density variations within food items. This empirical correction improves weight estimation accuracy across practical food portions.

### 5.2 Practical Applicability Assessment

The calorie estimation accuracy achieved in this research (MAPE 5.59-16.81%) is significantly below the generally acceptable error range for nutritional guidance (approximately ±20%), indicating that the system has reached a practical level of performance. The automated approach eliminates manual input requirements, making it accessible to a broader range of users.

### 5.3 Quantitative Comparison with Existing Methods

In comparison with DepthCalorieCam, significant error reductions were achieved: 85.41 kcal (83.9%) for karaage and 30.26 kcal (70.2%) for croquettes in terms of MAE reduction. These improvements can be attributed to the high-precision depth information acquisition using LiDAR sensors and the effectiveness of the confidence-based integration algorithm. The LiDAR technology provides more accurate 3D reconstruction compared to stereo camera-based depth estimation, while the multi-method volume calculation approach with confidence scoring ensures robust results across different food types and shapes.

### 5.4 System Robustness and Limitations

The system demonstrates good robustness across different food categories, with most foods achieving MAPE values below 15% for calorie estimation. However, certain limitations were identified: foods with elongated shapes (yakitori, hot dogs) showed higher sensitivity to photography angles, and foods with complex internal structures (yakisoba with mixed ingredients) exhibited increased estimation variance. The confidence-based volume selection mechanism successfully identified unreliable estimates, with all experimental data exceeding the 0.8 confidence threshold.

Cases of overestimation when measuring multiple foods together were confirmed. This is considered due to insufficient point cloud acquisition for spaces between foods or inaccurate convex approximation of overlapping parts. Figure 9 shows an example with large estimation errors where two croquettes were measured simultaneously.

Considering photography conditions and introducing photography from multiple viewpoints could suppress these errors. Advancing point cloud outlier removal algorithms and incorporating finer meshes could achieve further accuracy improvements.

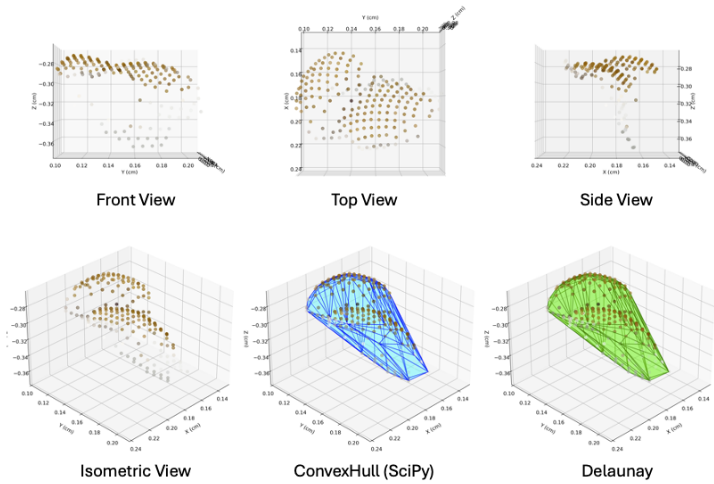



**Fig. 9.** Example with large estimation errors (simultaneous measurement of two "croquettes").

### 5.5 Future Research Directions

For practical implementation, several improvements are considered effective: (1) substantial expansion of the food database to cover a wider variety of cuisines and cooking methods; (2) improvement of 3D reconstruction accuracy through multi-view photography; (3) introduction of deep learning-based direct calorie regression models that can learn complex relationships between visual features and nutritional content; and (4) exploration of piecewise linear models or volume-dependent density functions to address the theoretical limitations of the current regression approach while maintaining practical accuracy. Additionally, integration with comprehensive nutritional databases and personalized dietary tracking systems could enhance the overall utility for health management applications.

## 6 Conclusion

This paper proposed a system "LiDARCalorieCam" that achieves real-time calorie estimation without using reference objects by utilizing iPhone's LiDAR sensor and deep learning models to accurately measure meal three-dimensional shapes. Experimental results confirmed accuracy improvements in volume estimation and calorie estimation compared to conventional methods, demonstrating application possibilities in daily meal management.

However, challenges remain such as limited recognizable meal categories and issues arising from photography condition variations and food diversity. Future work aims to realize practical meal management systems by addressing these limitations through improved algorithms and expanded training data.

**Acknowledgments**

This work was supported by JSPS KAKENHI 22H00548 and JST CRONOS JPMJCS24K4.

## References

1. Akpa, E., Suwa, H., Yasumoto, K.: Smartphone-based food weight and calorie estimation method for effective food journaling. SICE Journal of Control, Measurement, and System Integration (2017)
2. Ando, Y., Ege, T., Cho, J., Yanai, K.: Depthcaloriecam: A mobile application for volume-based food calorie estimation using depth cameras. In: Proc. of ACMMM Workshop on Multimedia Assisted Dietary Management (MADIMA) (2019)
3. Chen, L.C., Papandreou, G., et al.: Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
4. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1800–1807 (2017)
5. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. pp. 226–231 (1996)
6. Fang, S., Liu, C., et al.: Single-view food portion estimation based on geometric models. In: IEEE International Symposium on Multimedia (ISM). pp. 385–390 (2015)
7. Matsuda, Y., Hoashi, H., Yanai, K.: Recognition of multiple-food images by detecting candidate regions. In: Proc. of IEEE International Conference on Multimedia and Expo (ICME) (2012)
8. Okamoto, K., Yanai, K.: An automatic calorie estimation system of food images on a smartphone. In: Proc. of ACM MM Workshop on Multimedia Assisted Dietary Management (MADiMa) (2016)
9. Okamoto, K., Yanai, K.: Uec-foodpix complete: A large-scale food image segmentation dataset. In: Proc. of ICPR Workshop on Multimedia Assisted Dietary Management (MADiMa). pp. 647–659 (2021)
10. OpenCV Developers: Opencv: Camera calibration and 3d reconstruction (2025), https://docs.opencv.org/4.11.0/d9/d0c/group__calib3d.html
11. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI). pp. 234–241 (2015)
12. Tai, C., Keller, M., Nair, S., Chen, Y., Wu, Y., Markham, O., Parmar, K., Xi, P., Keller, H., Kirkpatrick, S., Wong, A.: Nutritionverse: Empirical study of various dietary intake estimation approaches. arXiv preprint arXiv:2309.07704 (2023)
13. Tanno, R., Ege, T., Yanai, K.: Ar deepcaloriecam v2: Food calorie estimation with cnn and ar-based actual size estimation. In: Proc. of the 24th ACM Symposium on Virtual Reality Software and Technology (2018)
14. Thames, Q., Karpur, A., Norris, W., Xia, F., Panait, L., Weyand, T., Sim, J.: Nutrition5k: Towards automatic nutritional understanding of generic food. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8903–8911 (2021)