Mathematical Biosciences and Engineering

MBE, 20(4): 6294–6311.
DOI: 10.3934/mbe.2023271
Received: 30 August 2022
Revised: 10 December 2022
Accepted: 09 January 2023
Published: 31 January 2023

http://www.aimspress.com/journal/MBE

Research article

Food volume estimation by multi-layer superpixel

Xin Zheng[1], Chenhan Liu[2], Yifei Gong[3], Qian Yin[1,]*, Wenyan Jia[5] and Mingui Sun[4,5]

[1] School of Artificial Intelligence, Beijing Normal University, Beijing 100875, China
[2] School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China
[3] Beijing Sankuai Online Technology Co., Ltd., Beijing 100190, China
[4] Department of Neurosurgery, University of Pittsburgh, PA 15260, USA
[5] Department of Electrical and Computer Engineering, University of Pittsburgh, PA 15260, USA

* **Correspondence:** Email: yinqian@bnu.edu.cn; Tel: +861058807943.

Abstract: Estimating the volume of food plays an important role in diet monitoring. However, it is difficult to perform this estimation automatically and accurately. A new method based on the multi-layer superpixel technique is proposed in this paper to avoid tedious human-computer interaction and improve estimation accuracy. Our method includes the following steps: 1) obtain a pair of food images along with the depth information using a stereo camera; 2) reconstruct the plate plane from the disparity map; 3) warp the input image and the disparity map to form a new direction of view parallel to the plate plane; 4) cut the warped image into a series of slices according to the depth information and estimate the occluded part of the food; and 5) rescale superpixels for each slice and estimate the food volume by accumulating all available slices in the segmented food region. Through a combination of image data and disparity map, the influences of noise and visual error in existing interactive food volume estimation methods are reduced, and the estimation accuracy is improved. Our experiments show that our method is effective, accurate and convenient, providing a new tool for promoting a balanced diet and maintaining health.

Keywords: food volume estimation; multi-layer superpixel; stereo vision; disparity map

1. Introduction

The World Health Organization (WTO) has classified obesity as a disease. In 2016, more than

1.9 billion adults (39%) in the world aged 18 and above were overweight (BMI > 25), among which more than 650 million were obese. [1]. Being overweight or obese can have serious adverse health effects. Excessive adipose tissue accumulation can lead to many serious chronic diseases, such as cardiovascular disease (mainly heart disease and stroke), type 2 diabetes, musculoskeletal disorders and some forms of cancer (e.g., endometrial cancer, breast cancer and colon cancer). It has been found that obesity may lead to disability and even premature death [2,3].

The key to prevent overweight or obesity is to control the daily calorie intake and keep it in balance with the daily calorie expenditure. Therefore, self-monitoring of diet is of great importance in reducing body fat and preventing obesity.

A difficult part of diet monitoring is estimating the volume of food. With the recent advances in computer vision and artificial intelligence, a variety of image-based dietary assessment methods have been proposed [4–9], which can be further divided into model based [10–13], 3D reconstruction based [14–18] and learning based [19–20] methods. Despite the effectiveness of these methods, they still face many problems. Model-based methods need manual interaction; noise, visual errors, and other factors negatively impact the accuracy of 3D reconstruction based methods; and learning based methods are plagued by lack of training data. Currently, it is still difficult to estimate the volume of food automatically and accurately. However, food size (or portion size) is directly related to the calorie/nutrition intake. Thus, the significance of food volume estimation is self-evident.

In this paper, we propose a new method for estimating food volume. The food and plate are separated based on food pictures with depth information obtained by a stereo camera. The view of the camera is rotated virtually so that it is parallel to the plate plane. Then, the food coordinates are mapped and transformed accordingly. Different methods are adopted to volumetrically slice food according to its thickness and other characteristics. The slices are accumulated, and the total volume is obtained. The combination of image and depth data greatly reduces the influence of noise and visual error which have been significant problems in the conventional food volume estimation method. Our method improves food estimation accuracy, helps users estimate the nutrition content, and improves self-monitoring of diet in daily life.

2. Related works

2.1. Model based method

For food volume estimation from images, the conventional approach was to determine a food template based on feature points, and the volume is estimated from the selected template using certain algorithms [4–7]. For example, Zhu et al. [10] designed a model-based method to nest food with a specific geometric model, calculated parameter values of the matching geometric model with a checkerboard as a scale reference, and inferred the volume of food according to the volume of the geometric model. This method requires the food to have certain geometric characteristics. Thus the accuracy of this method is higher for food that conforms well to the model. However, due to the large varieties of food and cooking methods, it is difficult to prepare a set of models for general forms of food. Therefore, this method is not universal in use, but instead it is suitable for food with certain geometric shapes.

Chen et al. [12] improved the selection of reference objects. They proposed using the plate as a reference to calculate the size and position of the food relative to the camera, separated the food from its container, and matched the food shape with a library of templates stored in a database. All these procedures were performed in a semiautomatic way. Then, fine adjustments were conducted to adapt to irregular food shapes. This method greatly improves the accuracy of food volume estimation, but the radius of the plate as a scale reference needs a manual measurement. If the plates utilized in a dietary study are not standardized, this measurement must be performed for every meal, which increases the complexity of operation.

2.2. 3D reconstruction based method

Another method is to estimate food volume from multiple images in different views [4,14]. The mainstream idea is to calculate the parallax diagram based on pixel matching of binocular vision to carry out 3D reconstruction and estimate food volume. Most 3D reconstruction processes require an external calibration first. However, external calibration may lead to low resolution because, in the reconstructed model, each 3D point needs to correspond to a pair of points in the input images, which makes the 3D data sparse. To solve this problem, density reconstruction is carried out, in which all available pixels are used to build a 3D model. Among all these methods, stereo matching is commonly used. This method simplifies the one-to-one pixel matching between images by using the epipolar rectification.

Currently, there are two popular types of 3D reconstruction schemes. One is to build a 3D point cloud and then estimate the food volume from the cloud [15–17]. For example, Puri et al. proposed constructing the 3D point cloud of food and plates by stereo matching, obtaining food surface information through the point cloud, and extracting the food depth information using the RANSAC algorithm [15]. Finally, the food volume is estimated from the depth data. The other scheme obtains shape information and estimates food volume from multiple images in different perspectives [18]. For example, three different images were used with a checkerboard as a reference to obtain the scale information. Then, the food volume is estimated from the three perspective images.

However, these 3D reconstruction methods have four major disadvantages. First, during the 3D reconstruction process, considerable noise is present, and this noise significantly impacts the estimation of food volume. Second, in the absence of prior knowledge, food is difficult to separate from the image background. In addition, contour completion is required in the reconstruction process. Although this procedure works well for food with regular geometric shapes, the estimation error increases as the irregularity of food shape increases. Finally, the selection of scale reference is affected by many factors, and errors are produced when the operator uses improper scale references.

2.3. Learning based method

Due to the rapid development of the AI technology, food volume estimation methods based on deep learning have been proposed recently [19,20]. Convolutional Neural Networks (CNN) has been used in food recognition and volume estimation. An advantage of using deep learning is that the scale of the image can be learned from the global cues of the scene without the needs of camera calibration and scale reference. Although these methods have achieved reasonable results, it is still challenging to use deep learning for food volume estimation, mainly due to the insufficient 3D shape information from a single image. In addition, the accuracy of these methods relies heavily on the quality and availability of training data, which are difficult to obtain.

3. Materials and methods

3.1. Overview

We present a food volume estimation method based on stereo vision, multi-layer superpixel segmentation, and disparity maps. The estimation process is highlighted in Figure 1.

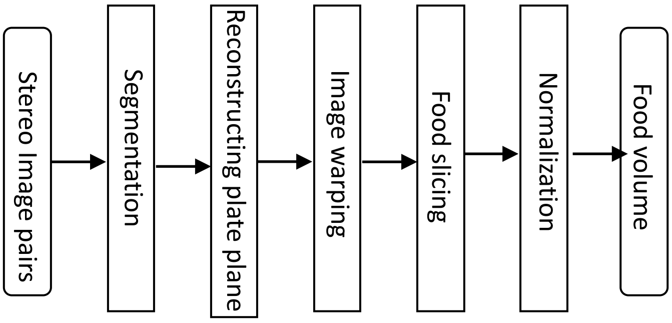

Figure 1. Algorithm flowchart.

3.2. Segmentation

First, the pair of stereo vision images is segmented into superpixels using the Simple Linear Iterative Cluster (SLIC) algorithm [21]. The SLIC algorithm performs local clustering of pixels based on the k-means technique in a 5-D space (l, a, b, x, y), where (*l*, *a*, *b*) represents the lightness scale, hue, and saturation in the CIELAB colour space, and (*x*, *y*) represents the coordinates of the pixel.

Each pixel P_i $(l_i, a_i, b_i, x_i, y_i)$ is clustered to the nearest clustering centre C_k $(l_k, a_k, b_k, x_k, y_k)$ by computing the distance measure D_k from P_i to C_k:

$$D_k = d_{lab} + \frac{m}{s} d_{xy} \tag{1}$$

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2} \tag{2}$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \tag{3}$$

where *m* is the superpixel compactness control parameter and *S* is the superpixel grid interval.

Then, the Density Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm is used to cluster the superpixels into several regions and separate the plate and food. The DBSCAN algorithm performs clustering relying on a density-based notation of clusters. It is designed to discover clusters of arbitrary shapes [22].

3.3. Reconstructing plate plane

By matching points in stereo images, a disparity map is obtained which contains depth information. Next, the Maximum Likelihood Estimation Sample Consensus (MLESAC) algorithm [23] is used to reconstruct the plate plane and calculate the camera orientation relative to the plate plane. The purpose of this calculation is to reconstruct an image within which the angle of view becomes parallel to the plate plane.

In order to determine the parameters of the plane (defined by $Ax+By+Cz+D = 0$, where A,B,C,D are parameters) in which the plate resides, an error cost, given by E, is minimized:

$$E = \sum_i p(e_i^2) \tag{4}$$

where e_i is the distance from each 3D point in plate region to the plate plane. The error is modelled by a mix of Gaussian and uniform distributions:

$$P_r(e^2) = \left(\gamma \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{e^2}{2\sigma^2}\right) + (1-\gamma)\frac{1}{\vartheta}\right) \tag{5}$$

where $0 < \gamma <= 1$ is the mixing factor, ϑ is the size of the search window, and σ is the standard deviation of the Gaussian distribution. Maximizing $P_r(e^2)$ is equivalent to minimizing the negative log likelihood:

$$-\mathrm{L} = -\log(\gamma \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{e^2}{2\sigma^2}\right) + (1-\gamma)\frac{1}{\vartheta}) \tag{6}$$

and

$$p(e_i^2) = \begin{cases} -L & e^2 < T^2 \\ T^2 & e^2 \geq T^2 \end{cases} \tag{7}$$

Equation (7) indicates that all points with e^2 less than T^2 are considered as the points within the plate plane, otherwise outside the plate plane.

3.4. Image warping

Taking the optical centre O as the origin of coordinates, the X, Y and Z directions are shown in Figure 2. Let the line of sight be the positive direction of the Z-axis. These establish the visual coordinate system. As shown in Figure 2, the original image I, and the responding disparity map I_d as well, are warped from the current view AOB to a new perspective view $A'OB'$ so that the new image I' and the warped disparity map $\mathrm{I_d}'$ represent the image for which the line of sight is parallel to the plate plane F.

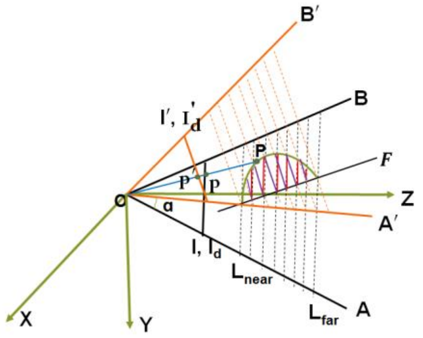

Figure 2. Schematic diagram of image warping.

Suppose $p(u,v)^T$ and $p'(u',v')^T$ are the projections of space point $P(X,Y,Z)^T$ on image I and I'. Let d and d' be the disparity values of point p and p' respectively. Based on the stereo vison principle, the depth *z* of space point *P* is inversely related to the stereo disparity value d of its projection point, given by:

$$z = \frac{f\,T_x}{d} = \frac{t}{d} \tag{8}$$

where *t* is the multiplication of focal length *f* and offset *Tx* of the stereo camera. Letting **H** be the camera projection matrix, we have:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{H}\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \end{pmatrix}\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} h_{11} & h_{12} & h_{14} \\ h_{21} & h_{22} & h_{24} \\ h_{31} & h_{32} & h_{34} \end{pmatrix}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + \begin{pmatrix} h_{13} \\ h_{23} \\ h_{33} \end{pmatrix}\frac{t}{d} = \mathbf{G}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + \begin{pmatrix} h_{13} \\ h_{23} \\ h_{33} \end{pmatrix}\frac{t}{d} \tag{9}$$

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \mathbf{G}^{-1}\begin{pmatrix} u - h_{13}\frac{t}{d} \\ v - h_{23}\frac{t}{d} \\ 1 - h_{33}\frac{t}{d} \end{pmatrix} \tag{10}$$

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \mathbf{R}_x(\alpha)\begin{pmatrix} x \\ y \\ t/d \\ 1 \end{pmatrix} \tag{11}$$

$$\begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \end{pmatrix}\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \mathbf{HR}_x(\alpha)\begin{pmatrix} x \\ y \\ t/d \\ 1 \end{pmatrix} \tag{12}$$

$$d' = \frac{t}{z'} \tag{13}$$

The mapping formula can be obtained by substituting (10) into (12).

Since the space plane corresponding to each pixel point in image *I* needs to be stretched and rotated, its projected area in transformed image *I'* may expand, resulting in non-pixel parts of the transformed image, shown as holes or gaps. Similarly, multiple pixels in image *I* may correspond to the same pixel point in image *I'* after mapping, resulting in a loss of image information. To avoid the loss in the forward warping process, we propose back projection procedure: 1) finding the corresponding point in the source image *I* for each pixel in the target image *I'*, and 2) obtaining the depth and colour values in *I'*. Like the case of forward warping, the back projection can be obtained by

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{HR}_x(-\alpha) \begin{pmatrix} x' \\ y' \\ t/_{d'} \\ 1 \end{pmatrix} \tag{14}$$

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \mathbf{G}^{-1} \begin{pmatrix} u' - h_{13}\frac{t}{d\prime} \\ v' - h_{23}\frac{t}{d\prime} \\ 1 - h_{33}\frac{t}{d\prime} \end{pmatrix} \tag{15}$$

In this formula, d' is unknown, so forward warping should be performed first to obtain the disparity value d' of each pixel on the target image:

$$d' = \frac{t}{z\prime} = \frac{t}{\frac{t}{d}\cos\alpha - y\sin\alpha} = \frac{t}{\frac{t}{d}\cos\alpha - (\mathbf{G}^{-1})_2^T \begin{pmatrix} u - h_{13}\frac{t}{d} \\ v - h_{23}\frac{t}{d} \\ 1 - h_{33}\frac{t}{d} \end{pmatrix} \sin\alpha} \tag{16}$$

where $(\mathbf{G}^{-1})_2^T$ is the second row of matrix$(G^{-1})^T$. The calculation in (16) looks complex. We propose a simplified one by replacing displacement value d' with a function of d_f(distance to plate plane *F*). Define

$$\mathbf{P}_r = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ A & B & C & D \\ h_{31} & h_{32} & h_{33} & h_{34} \end{pmatrix} \tag{17}$$

Let A, B, C and D be the parameters of the equation of plate plane F (given by $Ax + By + Cz + D = 0$). We can obtain:

$$\begin{pmatrix} wu \\ wv \\ wd_r \\ w \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ A & B & C & D \\ h_{31} & h_{32} & h_{33} & h_{34} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = P_r \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{18}$$

Here,

$$w\ d_r = Ax + By + Cz + D = d_f\ \sqrt{A^2 + B^2 + C^2} \tag{19}$$

where d_f stands for the distance to plate plane *F*, as shown in Figure 3.

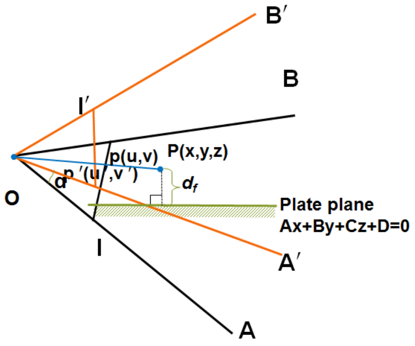

Figure 3. Schematic diagram of d_f.

From (19), we can obtain

$$d_r = d_f \ (\sqrt{A^2 + B^2 + C^2})/w \tag{20}$$

Similarly, (20) can be modified to obtain the projection formula of the target image:

$$\begin{pmatrix} w'u' \\ w'v' \\ w'd'_r \\ w' \end{pmatrix} = \mathbf{P}_d \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{21}$$

According to (18) and (21),

$$\begin{pmatrix} wu \\ wv \\ wd_r \\ w \end{pmatrix} = \mathbf{P}_r \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \mathrm{P}_r \mathbf{P}_d{}^{-1} \begin{pmatrix} w'u' \\ w'v' \\ w'd'_r \\ w' \end{pmatrix} = \mathbf{T}_{rd} \begin{pmatrix} w'u' \\ w'v' \\ w'd'_r \\ w' \end{pmatrix} \tag{22}$$

$$\begin{pmatrix} wu \\ wv \\ w \end{pmatrix} = \mathbf{H}_{rd} \begin{pmatrix} w'u' \\ w'v' \\ w' \end{pmatrix} + w'd'_r e_{rd} \tag{23}$$

where H_{rd} is the matrix of $\mathbf{T}_{rd}$ without the third row and the third column, and e_{rd} is the third column of $\mathbf{T}_{rd}$ without the third row. Since, by (19), $w'd'_r = wd = Ax + By + Cz + D$, we can obtain:

$$d'_r = \frac{w}{w'} d \tag{24}$$

3.5. Food slicing

According to the disparity information, the image is cut sequentially into a series of slices in the depth direction, as shown in Figure 4.

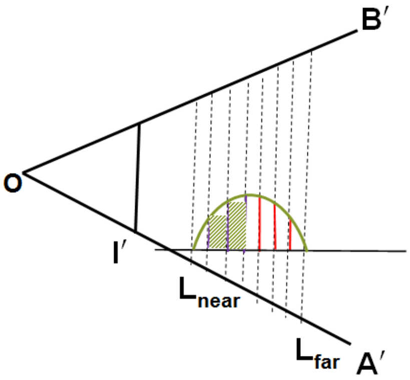

Figure 4. Slice diagram.

In order to predict the shape of the occluded part of the food effectively, the food is classified into two types determined by a user-determined threshold δ: 1) thin food, such as pizza and green beans, in which the maximum height (the distance between the highest point of the food and the plate plane) is less than δ, and 2) thick food, such as oranges and hamburgers, in which the maximum height is at least δ. We use different completion schemes for these two food types.

3.5.1. Thin food

For thin food, few part is occluded. We assume that the depth (in Z direction) of the occluded part does not exceed the height (in Y direction) of the food. In this case, we add m slices to the food, with $m = \frac{h}{s}$, where h is the height of the food and s is the slice thickness.

3.5.2. Thick food

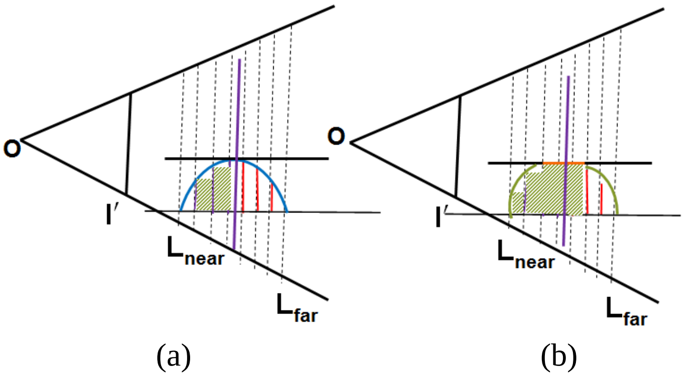

Figure 5. Cutting schemes for: (a) food without a flat top, and (b) food with a flat top.

For thick food, we assume that the food shape is symmetric and the highest point is visible. With this assumption, we compute the volume of the visible half and multiply the result by two as the estimate of food volume. Two cutting schemes are used according to the form of food top.

If the top of the food is a single point or multiple points occupying a small area, as shown in Figure 5(a**),** we find the highest point and use it to divide the food into two parts (shown as the purple vertical line in Figure 5(a)).

On the other hand, if the food has a flat top occupying a large area, as shown in Figure 5(b**)**, we compute the centroid of the flat top and use it to divide the food into two parts, shown as the purple vertical line in Figure 5(b).

3.6. Normalization

Because an object presented in an image appears larger when it is close to the camera, and smaller when it is far from the camera, the average superpixel size in each slice represents an increasing physical size as the distance of view increases, as illustrated in Figure 6.

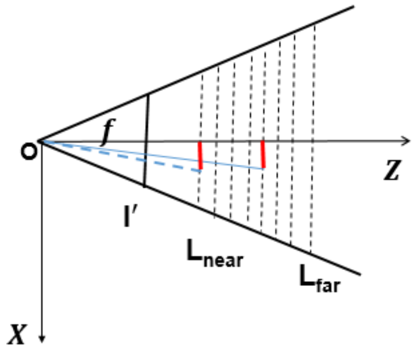

Figure 6. Schematic diagram of the normalization of superpixels.

Therefore, it is necessary to normalize the superpixels so that they represent approximately equal physical sizes regardless of the viewing distances. The following normalization formulas are utilized based on the depth information:

$$\frac{\Delta u}{\Delta x}=\frac{f}{z} \qquad \mathrm{N}S=\left(\frac{W}{\Delta u}\right)^2=\left(\frac{W}{f\Delta x}z\right)^2 \tag{25}$$

where Δu is the width increment of the superpixel, Δx is the increment in the X direction in the 3D space, f is the focal length of the camera, and z is the Z coordinate value of the superpixel. From the above formulas, the number of superpixel divisions NS in each sliced image is related to depth z. Let the number of superpixel divisions of the nearest slice layer L_{near} be NS_{near}. From the nearest slice layer L_{near} to the farthest slice layer L_{far} the food is divided into number of N_l slices with depth Δz and the number of superpixel divisions of the i-th slice is given by:

$$\frac{NS_i}{NS_{near}}=\left(\frac{z_i}{z_{near}}\right)^2=\left(\frac{z_{near}+i\,\Delta z}{z_{near}}\right)^2=\left(1+i\frac{\Delta z}{z_{near}}\right)^2=\left(1+i\frac{z_{far}-z_{near}}{z_{near}N_l}\right)^2$$

$$=\left(1+i\frac{\frac{f\,T_x}{d_{far}}-\frac{f\,T_x}{d_{near}}}{\frac{f\,T_x}{d_{near}}N_l}\right)^2=\left(1+i\frac{d_{near}-d_{far}}{d_{far}N_l}\right)^2 \tag{26}$$

$$NS_i = NS_{near}\left(1 + i\,\frac{d_{near} - d_{far}}{d_{far} N_l}\right)^2 \tag{27}$$

3.7. Volume estimation

After normalizing the superpixels in all slices, the area of each slice is calculated. Since the thickness of each slice is small, the slice volume can be approximated as the product of the slice area and the slice thickness. Finally, the food volume is estimated as the sum of all slice volumes.

4. Results

4.1. Raw image acquisition

We implemented the algorithms of our superpixel method in MATLAB®. Seven realistically shaped food replicas of known volumes (measured using water displacement) were used as the test objects. Each food was placed on a plate before a shot was taken by an Aiptek iDV stereo camera in an indoor environment illuminated by natural light. The results are shown in Figure 7. The distance between the food and the camera was approximately 1 m. The stereo image pair was separated into a left-eye image and a right-eye image. The corresponding disparity map was obtained by stereo vision matching.

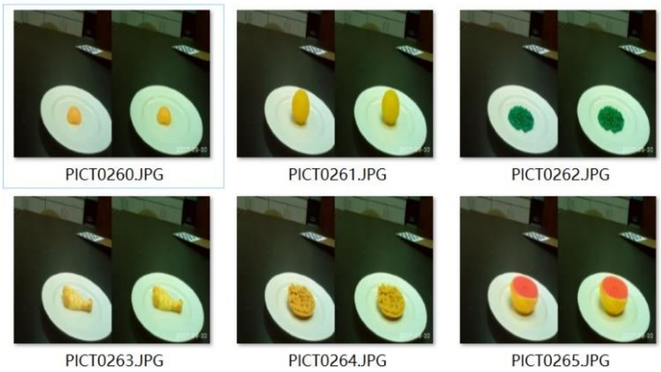

Figure 7. Raw images.

4.2. Segmentation

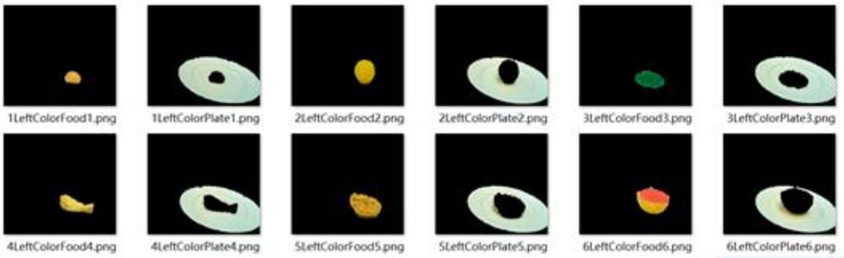

Figure 8. Segmentation results.

The results of food and plate segmentation by the SLIC and DBSCAN algorithms are shown in Figure 8.

4.3. Reconstructing plate plane

The plate plane, given by Ax + By +Cz +D = 0, was determined by the MLESAC algorithm. The resulting parameter values and the output images (where the plate plane is colored in red) are shown in Table 1 and Figure 9, respectively.

Table 1. Plane parameters of reconstructed plate plane.

Food	Plane parameters$[A, B, C, D]$
egg	[–0.02534, 0.90135, 0.43234, –0.90558]
orange	[0.12549, –0.936669, –0.32698, 0.60524]
chicken leg	[0.05850, –0.90413, –0.42323, 0.76347]
bread	[0.04248, –0.89979, –0.43425, 0.80871]
grapefruit	[–0.09693, 0.91111, 0.40060, –0.70904]
cake	[0.07609, –0.90506, –0.41841, 0.74542]
peach	[–0.07205, 0.90717, 0.41454, –0.74395]

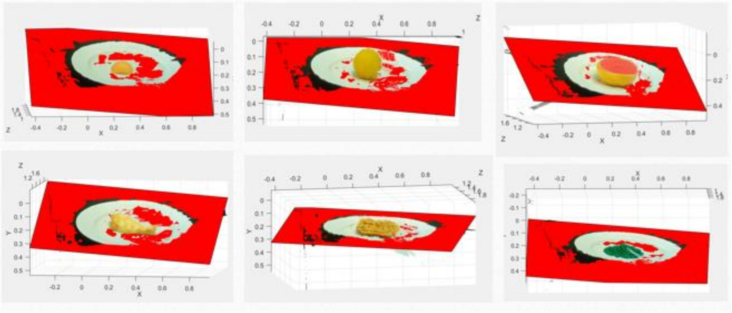

Figure 9. Results of reconstructed plate plane.

4.4. Image warping

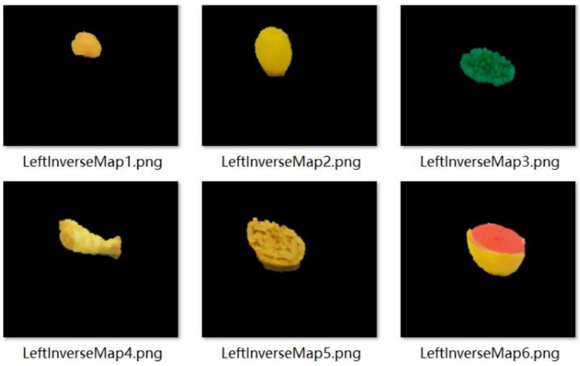

Figure 10. Results of image warping.

The forward warping/back projection procedure was used to obtain a new sight of view paralleling to the plate plane. The results of the image warping are shown in Figure 10.

4.5. Food slicing

The food after forward warping/back projection was sliced according to threshold δ. As stated in Section 3.5, different methods were used, determined by the food thickness. The results are shown in Figure 11. The first (top part) and second (bottom part) sets of pictures are the example results of thick and thin foods, respectively.

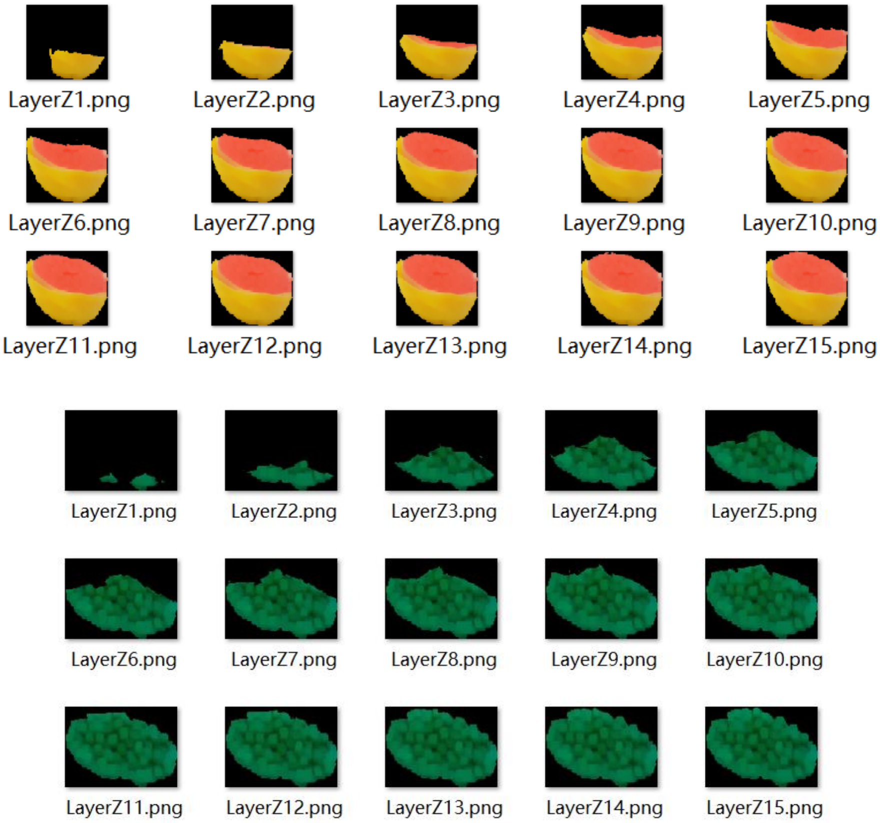

Figure 11. Results of food slicing.

4.6. Normalization

Each superpixel is normalized according to depth information to equalize physical size regardless of the viewing distance. The normalized result is shown in Figure 12.

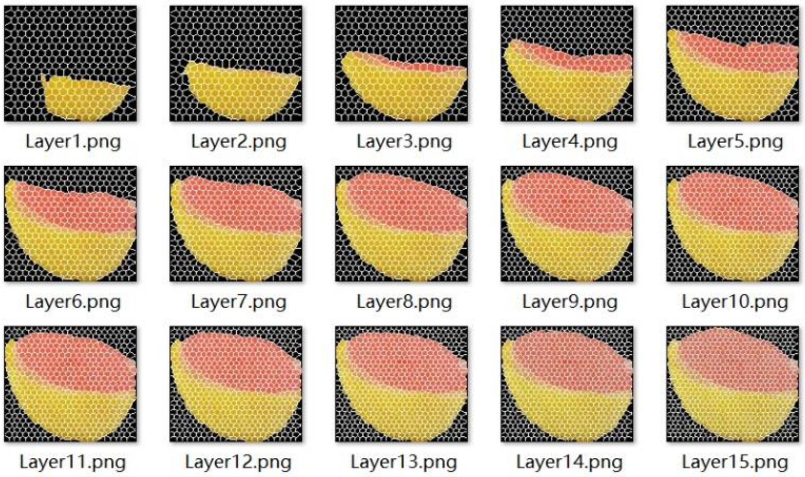

Figure 12. Results of superpixel normalization.

4.7. Volume estimation

Our experimental results are shown in Table 2. The food volumes (in cubic centimetres) obtained by averaging multiple water displacement measurements and by the proposed estimation method are denoted by V0 and V, respectively. The error rate ζ, calculated by $\zeta = \frac{|V-V0|}{V0} \times 100\%,$ is also listed in Table 2.

Table 2. Results of volume estimation and error rate.

Food type	V0	V	ζ
egg	20.67	20.79	0.58%
orange	151.67	152.61	0.62%
chicken leg	64.00	85.89	34.20%
bread	307.67	344.05	11.82%
grapefruit	272.00	255.13	6.20%
cake	93.67	82.43	12.00%
peach	151.67	128.94	14.99%

4.8. Accuracy analysis

It can be seen from the error rates that, in most cases, the food volume estimation accuracy by our method is generally high (except for the chicken leg) even only a pair of images was used in each estimation. For regularly shaped foods with high symmetry, such as eggs and oranges, more accurate results were obtained. On the other hand, for asymmetrical and thicker foods, such as chicken legs, the volume estimation error was generally larger.

4.9. Comparison

So far, model-based food volume estimation method and other manual interactive method have the highest accuracy. We compared two existing methods by Chen et al. [12] and Yang et al. [9] against water displacement measurements as the gold standard. As shown in Table 3, Chen's method has the highest accuracy, but this algorithm requires manually selecting and manipulating three-dimensional objects, which are tedious and difficult to use in practice. When compared with Yang's method which uses the smart phone, our algorithm is more accurate (except for the chicken leg) and does not require manual procedures.

Table 3. Comparison of the error rates with other methods.

Food type	Error rate		
	ours	Yang's[9]	Chen's[12]
egg	0.58%	41.10%	-
Chicken leg	34.20%	12.18%	0.85%
grapefruit	6.20%	-	1.78%
cake	12.00%	14.91%	-
peach	14.99%	17.56%	1.46%

5. Discussion

5.1. Missing data problem

As described in Section 3.4, a forward warping procedure is required to obtain the food depth information once the camera's orientation becomes parallel to the plate plane. In this step, holes will likely appear due to missing data in certain locations. This problem is exemplified in Figure 13.

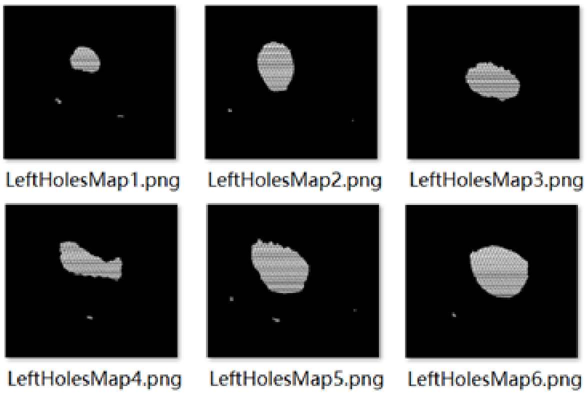

Figure 13. Results of forward warping.

Our solution is to assume that the depth information is continuous and fill the holes according to this assumption. Suppose that there is a lack of depth value at point *A* (i.e., a hole), neighbouring points of point *A* with depth values are selected, and the average value is assigned to point *A* as its depth value. If the neighbours are absent from depth values and point *A* is judged to be a food point, the depth values of the points closest to Point *A* are utilized to construct its depth value by interpolation. However, in non-smooth regions of food, the continuous assumption is invalid, which leads to a certain error.

5.2. Rounding error in coordinate data

In the process of coordinate mapping, it is best to use real-valued numbers to represent pixel image coordinates because these coordinates contain depth information. If integers are used to represent coordinates, the rounding error must be considered. Similarly, in the process of superpixel normalization, it is necessary to compute the area corresponding to each superpixel. In this case, if integers are used to represent coordinates after a rotation, the rounding operation also causes error in the estimation result.

6. Conclusions

In this paper, a food volume estimation method based on multi-layer superpixel segmentation is proposed. A food image with depth information is obtained by using a stereo camera. The superpixel segmentation method is utilized to separate the food and the plate and then reconstruct the plate plane based on the parameters of the stereo camera and the scale calibration information of the plate. Next, we computationally alter the orientation of the camera (so that it is parallel to the plate plane) and forward warp the input images to obtain new disparities. We then perform a back projection to obtain a converted image and disparity map. Subsequently, the image is divided into a series of slices according to the food thickness, each slice is represented by superpixels, and the superpixels from different slices are normalized. Finally, we accumulate the volumes of all slices to obtain the total volume of the food.

Compared with the traditional method, our method can adapt to foods of various shapes, not restricted by geometric constraints. Thus, our method has good applicability in the real-world setting. In addition, our method greatly reduces the needs of human involvements, therefore it is more convenient to use. Moreover, our method can separate food from plate, allowing more specific use of the depth information to calculate food volume.

Acknowledgments

The research work described in this paper was supported in part by the China Scholarship Council for collaborative research abroad, and the Joint Research Fund in Astronomy (U2031136) under cooperative agreement between the NSFC and CAS. We thank Dr. Hua Zou, Dr. Ruowei Qu, Dr. Linyan Cui and Dr. Yuecheng Li for their support in data preparation and algorithm discussion. We thank the anonymous reviewers for their constructive feedback.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. *World Health Organisation*, Obesity and overweight, 2018. Available from: http://www.who.int/mediacentre/factsheets/fs311/en/.

2. G. Ni, J. Zhang, F. Zheng, The current situation and trend of obesity epidemic in China, *Food Nutr. China*, **19** (2013), 70–74.
3. *World Health Organisation*, What are the health consequences of being overweight?, 2013. Available from: https://www.who.int/features/qa/49/en/.
4. F. Lo, Y. Sun, J. Qiu, B. Lo, Image-based food classification and volume estimation for dietary assessment: A review, *IEEE J. Biomed. Health Inform.*, **24** (2020), 1926–1939. https://doi.org/10.1109/JBHI.2020.2987943
5. W. Tay, B. Kaur, R. Quek, Current developments in digital quantitative volume estimation for the optimisation of dietary assessment, *Nutrients*, **12** (2020), 1167. https://doi.org/10.3390/nu12041167
6. I. Nyalala, C. Okinda, K. Chen, T. Korohou, L. Nyalala, C. Qi, Weight and volume estimation of poultry and products based on computer vision systems: A review, *Poult. Sci.*, **100** (2021). https://doi.org/10.1016/j.psj.2021.101072
7. V. B. Raju, E. Sazonov, A Systematic Review of Sensor-Based Methodologies for Food Portion Size Estimation, *IEEE Sens. J.*, **21** (2021), 12882–12899. https://doi.org/ 10.1109/JSEN.2020.3041023
8. M. Sun, Q. Liu, K. Schmidt, J. Yang, N. Yao, J. Fernstrom, et al., Determination of food portion size by image processing, *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, (2008), 871–874. https://doi.org/10.1109/EMBS10205.2008
9. Y. Yang, W. Jia, T. Bucher, H. Zhang, M. Sun, Image-based food portion size estimation using a smartphone without a fiducial marker, *Public Health Nutrition*, **22** (2018), 1180–1192. https://doi.org/10.1017/S136898001800054X
10. F. Zhu, M. Bosch, I. Woo, S. Kim, C. Boushey, D. Ebert, et al., The use of mobile devices in aiding dietary assessment and evaluation, *IEEE J. Sel. Top. Sign. Proces.*, **4** (2010), 756–766. https://doi.org/10.1109/JSTSP.2010.2051471
11. H. C. Chen, Y. Yue, Z. Li, J. Fernstrom, Y. Bai, C. Li, et al., Accuracy of food portion size estimation from digital pictures acquired by a chest-worn camera, *Public Health Nutr.*, **17** (2014), 1671–1681. https://doi.org/ 10.1017/S1368980013003236
12. H. Chen, W. Jia, Y. Yue, Z. Li, Y. Sun, J. Fernstrom, et al., Model-based measurement of food portion size for image-based dietary assessment using 3D/2D registration, *Meas. Sci. Technol.*, **24** (2013). https://doi.org/10.1088/0957-0233/24/10/105701
13. C. Xu, Y. He, N. Khanna, C. Boushey, E. Delp, Model-based food volume estimation using 3D pose, *IEEE Int. Conf. Image Process.*, (2013), 2534–2538. https://doi.org/10.1109/ICIP.2013.6738522
14. J. Dehais, M. Anthimopoulos, S. Shevchik, S. Mougiakakou, Two-view 3D reconstruction for food volume estimation, *IEEE Trans Multimedia*, **19** (2017), 1090–1099. https://doi.org/10.1109/TMM.2016.2642792
15. M. Puri, Z. Zhu, Q. Yu, A. Divakaran, H. Sawhney, Recognition and volume estimation of food intake using a mobile device, *Workshop Appl. Comput. Vis.*, (2009), 1–8. https://doi.org/10.1109/WACV.2009.5403087
16. M. Rahman, Q. Li, M. Pickering, M. Frater, D. Kerr, C. Bouchey, et al., Food volume estimation in a mobile phone based dietary assessment system, *Int. Conf. Signal Image Technol. Internet Based Syst.*, (2012), 988–995. https://doi.org/10.1109/SITIS.2012.146

17. T. Suzuki, K. Futatsuishi, K. Yokoyama, N. Amaki, Point cloud processing method for food volume estimation based on dish space, *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, (2020), 5665–5668. https://doi.org/10.1109/EMBC44109.2020.9175807
18. H. Yin, 3D reconstruction from infrared stereo image pairs, *Masters Abstr. Inte.*, 2013.
19. L. Zhou, C. Zhang, F. Liu, Z. Qiu, Y. He, Application of deep learning in food: A review. *Compr. Rev. Food Sci. Food Saf.*, **18** (2019), 1793–1811. https://doi.org/10.1111/1541-4337.12492
20. F. Lo, Y. Sun, J. Qiu, B. Lo, Food volume estimation based on deep learning view synthesis from a single depth map, *Nutr.*, **10** (2018). https://doi.org/10.3390/nu10122005
21. F. Boemer, E. Ratner, A. Lendasse, Parameter-free image segmentation with SLIC, *Neurocomputing*, **277** (2018), 228–236. https://doi.org/10.1016/j.neucom.2017.05.096
22. J. Hou, C. Sha, L. Chi, Q. Xia, N. Qi, Merging dominant sets and DBSCAN for robust clustering and image segmentation, *IEEE Int. Conf. Image Process.*, (2014), 4422–4426. https://doi.org/10.1109/ICIP.2014.7025897
23. P. Torr, A. Zisserman, MLESAC: A New Robust Estimator with Application to Estimating Image Geometry, *Comput. Vis. Image Und.*, **78** (2000), 138–156, https://doi.org/10.1006/cviu.1999.0832