FOOD VOLUME ESTIMATION FROM A SINGLE IMAGE USING VIRTUAL REALITY TECHNOLOGY

by

Zhengnan Zhang

Bachelor of Science, Southwest Jiaotong University, 2009

Submitted to the Graduate Faculty of

Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Master of Science in Electrical Engineering

University of Pittsburgh

2010

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This thesis was presented

by

Zhengnan Zhang

It was defended on

November 22th, 2010

and approved by

Ching-Chung Li, Ph.D., Professor, Electrical and Computer Engineering Department

John D. Fernstrom, Ph. D., Professor, Psychiatry and Pharmacology

Robert J. Sclabassi, Ph. D., M.D., Professor Emeritus, Neurological Surgery, Electrical and Computer Engineering Department

Zhi-Hong Mao, Ph. D, Assistant Professor, Electrical and Computer Engineering Department

Thesis Advisor: Mingui Sun, Ph. D., Professor, Neurological Surgery, Electrical and Computer Engineering Department

FOOD VOLUME ESTIMATION FROM A SINGLE IMAGE USING VIRTUAL REALITY TECHNOLOGY

Zhengnan Zhang, M.S.

University of Pittsburgh, 2010

Obesity has become a widespread epidemic threatening the health of millions of Americans and costing billions of dollars in health care. In both obesity research and clinical intervention, an accurate tool for diet evaluation is required. In this thesis, a new approach to the estimation of the volume of food from a single input image is presented based on the virtual reality (VR) technology. A software system is constructed for food image acquisition, camera parameters calibration, virtual reality modeling, virtual object manipulation, and food volume estimation. Our system utilizes a checkerboard to calibrate the intrinsic and extrinsic parameters of the camera using image process techniques. Once these parameters are obtained, we establish a VR space in which a virtual 3D wireframe is projected into the food image in a well-defined proportional relationship. Within this space, the user is able to scale, deform, translate and rotate the virtual wireframe to fit the food in the image. Finally, the known volume of the wireframe is utilized to compute the food volume using the proportional relationship. Our experimental study has indicated that our VR system is highly accurate and robust in estimating volumes of both regularly and irregularly shaped foods, providing a powerful diet evaluation tool for both obesity research and treatment.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

Obesity has become an epidemic that threatens the health of millions of Americans and costs billions of dollars in health care. In the last 30 years, adult obesity rates have doubled and childhood obesity rates have more than tripled [1]. Obesity has been linked to many types of diseases, including cancers, respiratory diseases, cardiovascular diseases, digestive diseases, osteoarthritis, stroke, and birth defects. And approximately 300,000 deaths each year are related to obesity [2]. According to a number of studies, obesity increases health care costs within the U.S. by more than 25% [3].

The primary causes of obesity are excessive calorie intake and inadequate physical activity. Thus, in studies of obesity and its potential treatments, accurate dietary assessment is essential. Currently, the most common type of dietary assessment used is derived from self-reports by respondents [4,5,6]. However, this method is subject to significant error [7] because it depends on the accuracy of respondents' memories and their willingness to report their true dietary intake. Nutritional experts have questioned whether the full number of calories ingested by subjects is recorded. Lack of accurate dietary assessment has been a major stumbling block in both researches on obesity and treatment programs for the condition.

With recent rapid advances in the fields of image processing and electronic engineering, many sophisticated technologically based methods have been introduced into the process of dietary assessment. One of the main differences between these and the traditional self-report method is the use of recorded images or videos. This technology makes it possible to record the actual eating process in front of an individual throughout the entire recording period. There is no doubt that the visual electronic memory can provide highly accurate data on participants' food intake.

In addition to the use of recorded images or video, in recent years, computer vision techniques have been developed to estimate food volume and the use of image-based computational algorithms and software has been proposed to monitor and manage dietary intake. Approaches to food volume estimation based on fixed relative positions of camera and objects were reported by Rashidi [8] and Koc [9].These approaches require restrictive camera parameters and are suitable only for certain kinds of fruit (e.g., watermelon and kiwifruit). The calibration of camera parameters based on checkerboard [10] and spherical objects [11] have been proposed to reconstruct 3D objects from 2D images. These camera calibration methods which provide intrinsic and extrinsic parameters, are preliminary features of most approaches to volume estimation. For example, Woo [12] proposed a portion estimation approach that used a checkerboard pattern card inside the food image to provide scale for estimating food volume in the same image. Their estimation objects consisted of prismatic and spherical like food. These researchers also provided a method to derive the nutritional value from the volume of the foods based on the USDA Food and Nutrient Database for Dietary Studies (FNDDS) [13]. Similarly, Puri et al presented an approach to estimating food volume using a checkerboard as a dimensional reference [14]. Jia et al. [15] proposed another approach to estimating food dimensions using a circular object, such as a dining plate, as a dimensional reference without depending on restrictive camera parameters. Yao [16] proposed a similar approach for food-dimension estimation using structured lights including an LED spotlight and a simple laser beam pattern. In all these approaches, food volumes have been calculated indirectly and the errors of intermediate dimensional estimation have produced large errors that undermine the final results of volumetric estimation.

This thesis aims to propose a new approach to reduce errors caused by indirect estimation of food volume. Our system utilizes virtual reality (VR) techniques, an advanced simulation technology that represents experience in an imaginary world [17] to design and construct software based on this approach. Eliminating the intermediate processes, our approach estimated specific food volume directly, thus producing more accurate estimates of the volume of food from a single image. The software constructed a virtual space based on real camera intrinsic and extrinsic parameters obtained by the checkerboard method. Within this space, a virtual 3D wireframe was constructed and projected into the food image in a well-defined proportional relationship. The user was able to deform, scale, translate, or rotate the virtual wireframe until it fits the food inside the image, allowing the volume of food to be derived from the volume of the wireframe and the proportional relationship between the virtual space and real space.

Next, experiments were conducted to test and analyze the performance of this approach, comparing the affects of different experimental variables including camera position, object position, object shape, and so on. Statistical results of the experiments have indicated that our VR system is highly accurate and robust in estimating volumes of both regularly and irregularly shaped foods under various experimental conditions. Our system's positive performance in estimating food volume suggests that our approach has the potential to act as a powerful diet evaluation tool for obesity research as well as treatment.

2.0 BACKGROUND

2.1 VIRTUAL REALITY TECHNOLOGY

Virtual reality (VR) is an advanced technology that uses computer-generated environments to simulate places in the real world. As business, industry, and medicine have developed over the past ten years, more and more applications of virtual reality have been proposed to solve problems in these and other fields. The major advantages of VR is that it enables the users to function in a virtual environment dynamically with virtual objects, feel virtual environment by their hands, and obtain virtual feedback, all of which are created by computer and a number of advanced input and output devices.

The VR technology has brought computer-aided design (CAD) into a completely new environment based on a so-called VR-based CAD system [18,19,20], in which users can directly build and modify 3D models through 3D manipulation. In the business field, the collaborative VR (CVR) is an approach that enables a number of users to interact in a shared virtual environment, such as a Web3D applications X3D that constructs practical platforms for customers and companies to interact in the same virtual world. To date, a large number of approaches have been proposed in this field [21,22]. In the mechanical field, VR techniques have been extensively utilized for functions such as virtual layout design [23], assembly process simulation [24,25], and internet-based fault manufacturing [26]. However, one problem related to the application of VR application in mechanics is the conflict between the requirements of high-quality images and real-time operations. In the medical field, VR has been widely used in surgical training [27,28,29], psychotherapy [30,31,32], and disability rehabilitation [33,34]. Furthermore, the increasing trend of globalized manufacturing environments requires real-time exchange of information between different nodes in a product-development cycle, including design, setup planning, production, scheduling, machining, and assembly [35]. To date, VR technology has been applied to simulate these manufacturing processes before they are carried out, which has greatly enhanced the efficiency of manufacturing and made it more economically competitive.

Augmented reality (AR) is a new form of VR that overlays computer-generated models on real-world environments. The difference between AR and VR is that whereas the first one simply enhances the existing environment, the second one actually creates the whole environment. The design of an emerging type of see-through Head-Mounted Display (UMD) [36,37,38] is a typical AR technique which allows users to observe the virtual object in an existing scene, rather than creating a new world in front of the users. Moreover, many approaches are proposed for using AR to enhance manufacturing and industrial processes, e.g., industrial training [39], interior design and modeling [40], assembly planning [41], and computer-aided instruction [42].

The food volume estimation system described here is based on AR techniques, using virtual objects to enhance real scenes and calculate the volume of food in the scenes. Since AR technique is a part of VR technique, we will use only the term VR in the remainder of this thesis.

2.2 CAMERA PINHOLE MODEL

The *pinhole camera model* is use to represent the relationship between the coordinates of a 3D point and its projection on the image plane. A pinhole camera is a camera without a lens but has a single small aperture, which functions as a point to focus light [43]. The mathematical pinhole camera model is extensively used in the computer vision community. The principle of a pinhole camera is illustrated in Figure 1, where light from a scene passes through the pinhole and projects an inverted image on the image plane.

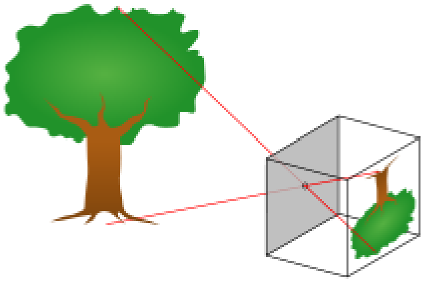


Figure 1. Principle of pinhole camera.

2.2.1 Perspective projection

Figure 2 represents a frontal pinhole imaging model. The reference frame $\boldsymbol{R} : (X, Y, Z)$ centers at the camera's optical center o, and the z axis is the optical axis of the camera's lens. The image point P with the coordinates $x = [x, y]^T$ is the projection of the point p with the coordinates $\boldsymbol{X} = [X, Y, Z]^T$. Since the point p is on a line determined by the point o and the point P, the relationship between point p and its image with the similar triangles method can be easily written as

$$x = f\frac{X}{Z}$$

$$y = f\frac{Y}{Z} \tag{2.1}$$

where f indicates the focal length.

This camera model is a so-called *ideal pinhole camera model*, and this projection method is known as *perspective projection*. [44]

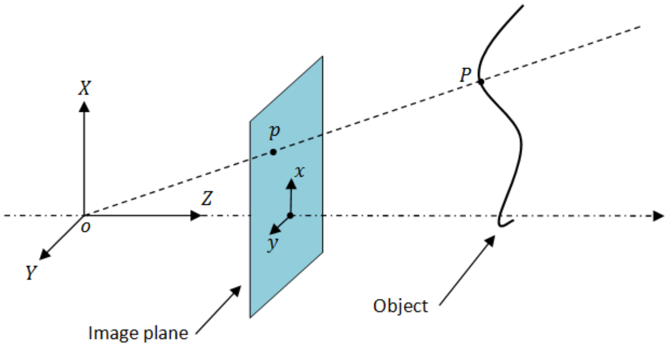


Figure 2. Frontal pinhole imaging model.

2.2.2 Camera parameters

Camera parameters consist of *intrinsic parameters* and *extrinsic parameters*. In general, intrinsic parameters describe the internal properties of the camera and extrinsic parameters describe the camera's posture and position relative to the world.

The intrinsic parameters encompass skew, principal point, image format, and focal length. In most cameras the skew parameter is negligible, the principal point is simply $(a/2\,,b/2)$ for the width and height (a,b) of the image plane, the image format depends on the pixel size, and the most important focal length can be obtained using several approaches, which will be described in Section 2.4.

The extrinsic parameters specify the position and rotation of a camera in the world. In practice, people always use the translation matrix in combination with the rotation matrix to transfer the real-world coordinate frame to a camera coordinate frame, as will be explained in Section 2.3.

2.3 VIRTUAL CAMERA MODEL

2.3.1 Representation of model

The fundamental problem for virtual geometric modeling is model representation. In this thesis, the model representation method is *triangle-based polygon representation* (also known as, “triangle mesh”). This triangle-based method describes each face of a geometric model using a series of single triangle units. Figure 3 shows an example of how triangle mesh can be used to approximate a teapot.

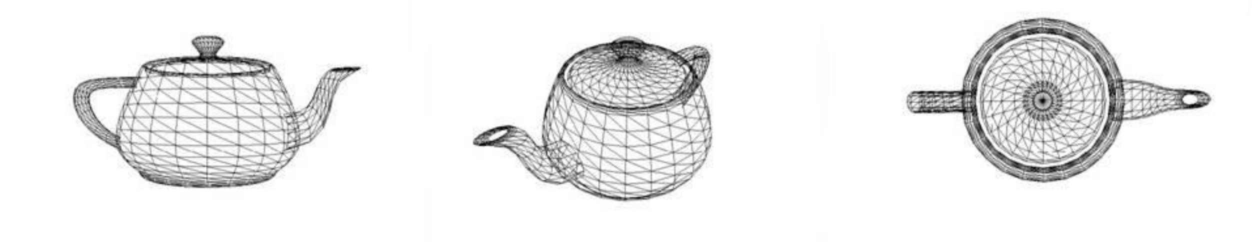


Figure 3. A Teapot described by triangle mesh.

With this model representation method, the intersection of two adjacent borders of triangle is called the vertex. Once the positions of the vertex are identified, the corresponding triangle unit is determined. In real applications of virtual model construction, the vertex is also used to store information of color, material, texture and the normal properties of each triangle unit.

2.3.2 Object coordinate frame and world coordinate frame

Object coordinate frame

In a virtual environment, the geometric object modeling is always associated with the *object coordinate frame* (also known as the *modeling coordinate frame* or *body coordinate frame*). In the object coordinate frame, an object modeling will be entirely specified, and every point in an object coordinate frame can be identified with three coordinates

$$\boldsymbol{X} = [x_1, x_2, x_3]^T = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

In some cases, we may also use $[x, y, z]^T$ to represent the three individual coordinates of a single point instead of $[x_1, x_2, x_3]^T$.The reason for constructing a model in an object coordinate frame instead of a *world coordinate frame* is that this approach allows people to construct the model without needing to consider its position, scale, and orientation relative to a world coordinate frame. Figure4 illustrates a teapot model which has been constructed in such an object coordinate frame.

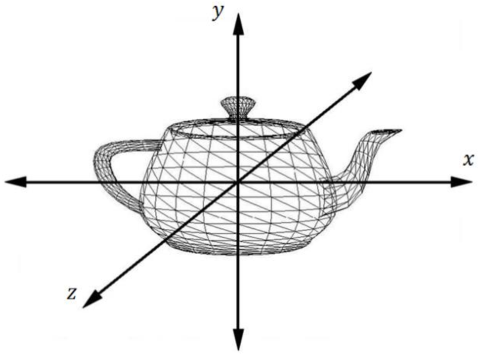


Figure 4. A teapot defined in an object coordinate frame.

World coordinate frame

As the first step of object modeling, models are constructed in an object coordinate frame. One must then organize and transfer these models to the world coordinate frame $\boldsymbol{W}$: (X, Y, Z) in order to build a virtual world scene. Each model in the object coordinate frame must undergo a *world transformation* to be introduced into the world coordinate frame.

This world transformation includes translation, rotation and scaling. Figure 5 illustrates that the teapot has been transformed into a world coordinate frame. If this teapot had applied only the translation and rotation operations, but not the scaling operation, the transform would be the so-called *rigid-body motion*, which means that the individual points will not move relative to each other within the object coordinate frame. Thus, the distance between any two points of the teapot will never change, unless we introduced the scaling operation to the world transformation.

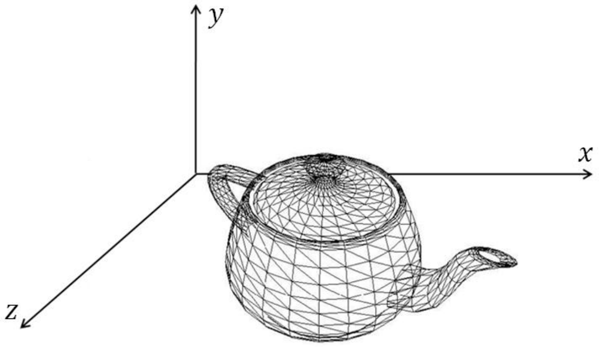


Figure 5. A teapot transformed to the world coordinate frame.

World transformation

In real application of virtual camera model construction, one needs a 4×4 matrix **G** to indicate a world transformation. The matrix **G,** having different values of elements, demonstrates the different transformations. In order to implement the matrix manipulation for the world transformation, we must express a 3D point as a 4×1 vector. This can be done by rewriting the three coordinates of a point as the first three elements of the new vector and then appending a "1" to the new vector, which is called the *homogeneous coordinate transformation*. Thus, if we have the a point $\boldsymbol{X} = [x_1, x_2, x_3]^T$, in homogeneous coordinates, it should be denoted by

$$\overline{\boldsymbol{X}} = \begin{bmatrix} \boldsymbol{X} \\ 1 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{bmatrix}$$

Thus, the product of the world transformation matrix $\boldsymbol{G}$ and the point $\boldsymbol{X}$ equals a new vector $\overline{\boldsymbol{X_W}} = [X_1, X_2, X_3, 1]^{\boldsymbol{T}}$ that indicates the coordinates of the point relative to the world frame $\boldsymbol{W}$,

$$\overline{\boldsymbol{X_W}} = \boldsymbol{G}\overline{\boldsymbol{X}} \tag{2.2}$$

As mentioned earlier, we use the triangle-based polygon to represent the models here, and the triangle-based method is demonstrated by the position information of vertexes in coordinates *x*, *y* and *z*. Then, if we create a vector set $\boldsymbol{V} = [\boldsymbol{X_1}\ \boldsymbol{X_2}\ \dots\ \boldsymbol{X_n}]$ to indicate all the point s (replace the terms of vertex) of a modal, the model $\boldsymbol{V}$ undergoing a world transformation $\boldsymbol{G}$ can be simply expressed as

$$\overline{\boldsymbol{V_W}} = \overline{\boldsymbol{V}}\boldsymbol{G} \tag{2.3}$$

where $\boldsymbol{V_W}$ indicates the new vector set that describes the model relative to world frame.

Translation matrix

During a translation process, the coordinates $\boldsymbol{X}$ of a fixed 3D point p relative to its object frame are transformed to its coordinates relative to world frame $\boldsymbol{W}$ by

$$\overline{\boldsymbol{X_W}} = \boldsymbol{T}\overline{\boldsymbol{X}} \tag{2.4}$$

where the translation matrix $\boldsymbol{T}$ can be written as

$$\boldsymbol{T} = \begin{bmatrix} 1 & 0 & 0 & -p_x \\ 0 & 1 & 0 & -p_y \\ 0 & 0 & 1 & -p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

where p_x, p_y and p_z are the values of the translation vector $\boldsymbol{A}$ from the object frame to the world frame, as shown in Figure 6.

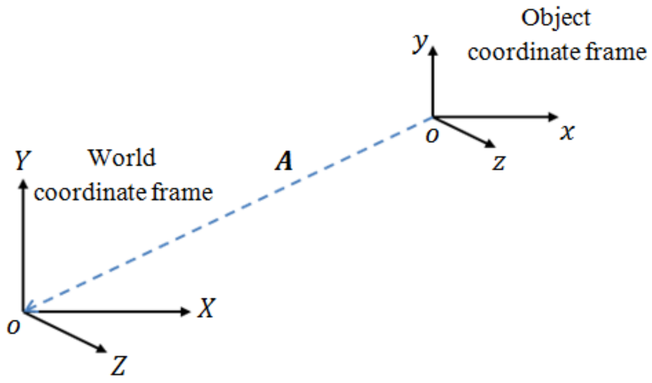


Figure 6. A translation between an object frame to a world frame.

Rotation matrix

For the rotation transformation from the object coordinates $\boldsymbol{X}$ of a fixed 3D point p to its coordinates relative to world frame $\boldsymbol{W}$, the configuration of rotating can be written by

$$\overline{\boldsymbol{X_W}} = \boldsymbol{R}\overline{\boldsymbol{X}} \tag{2.6}$$

where matrix $\boldsymbol{R}$ indicates the rotation matrix and vector $\boldsymbol{X_W}$ indicates the coordinates of point p relative to frame $\boldsymbol{W}$. We have three different kinds of rotation matrix, for rotation along x axis, y axis and z axis, respectively.

The matrix that indicates a rotation about x axis by an angle θ is

$$\boldsymbol{R_x}(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The matrix that indicates a rotation about y axis by an angle θ is

$$\boldsymbol{R_y}(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The matrix that indicates a rotation about z axis by an angle θ is

$$\boldsymbol{R_z}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Scaling matrix

For the object coordinates $\boldsymbol{X}$ of a fixed point, the configuration of the scaling, along x axis by q_x times, along y axis by q_y times and along z axis by q_z times, can be written as

$$\overline{\boldsymbol{X_W}} = \boldsymbol{S}\overline{\boldsymbol{X}} \tag{2.7}$$

where vector $\boldsymbol{X_W}$ indicates the coordinates of the same point relative to world frame $\boldsymbol{W}$, and matrix $\boldsymbol{S}$ represents the scaling matrix:

$$\boldsymbol{S} = \begin{bmatrix} q_x & 0 & 0 & 0 \\ 0 & q_y & 0 & 0 \\ 0 & 0 & q_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Combination of different world transformations

For the entire process of world transformation, one model must always implement several different world transformations instead of implementing only one. In such a case, the order of matrix manipulation must be the same as the order of the different world transformations. For example, if we have the task to use a translate matrix $\boldsymbol{T}$ to move a virtual model along x axis by two units, and then use a rotation matrix $\boldsymbol{R}$ to make it rotate about x axis by π rad, and finally use a scaling matrix $\boldsymbol{S}$ to scale the y coordinate of all the point of the model by two times, the object coordinates $\boldsymbol{X}$ of a single point of the model should be transformed by the following equations, in the order,

$$\overline{\boldsymbol{X}'} = \boldsymbol{T}\overline{\boldsymbol{X}}$$

$$\overline{\boldsymbol{X}''} = \boldsymbol{R}\overline{\boldsymbol{X}'}$$

$$\overline{\boldsymbol{X_W}} = \overline{\boldsymbol{X}'''} = \boldsymbol{S}\overline{\boldsymbol{X}''} \tag{2.8}$$

where vector $\boldsymbol{X_W}$ indicates the world coordinates of the point, and vectors $\boldsymbol{X}'$, $\boldsymbol{X}''$, $\boldsymbol{X}'''$ are the intermediate variables for the transformation.

In this formula, the order of multiplication is identified. Changing the order of the multiplication involves taking a totally different operation to the vector $\boldsymbol{X}$**,** which would introduce a large error. The advantage of matrix operation is that the different world transformation matrix can be synthesized. For the same example, we can use matrix multiplication to multiply the three different matrixes $\boldsymbol{T}$, $\boldsymbol{R}$*,* and $\boldsymbol{S}$ to produce a single world transformation matrix $\boldsymbol{P}$, and then multiply $\boldsymbol{X}$ with $\boldsymbol{P}$ to gain $\boldsymbol{X_W}$, as following

$$\boldsymbol{P} = \boldsymbol{TRS}$$

$$\overline{\boldsymbol{X_W}} = \boldsymbol{P}\overline{\boldsymbol{X}} \quad (2.9)$$

As mentioned before, one can express a model as a vector set. Then, the product of this vector set and the world transformation matrix $\boldsymbol{P}$ is the new vector set that can determine the result coordinates of all the points of the model relative to the world frame $\boldsymbol{W}$.

2.3.3 Camera coordinate frame

Visualization is the main task following the identification of a geometric model. The fundamental of visualization is to configure a virtual camera in the world frame $\boldsymbol{W}$. As the world transformation of a model requiring position information, one must determine the coordinates of a camera and its shooting target (here we use a single point which determines the camera's optical axle with the camera's optical center) in Frame $\boldsymbol{W}$. The purpose of fixing these two points is to identify the relationship between the camera and a scene. Furthermore, one must identify the rotation angle of the camera around its optical axis. Because the rotation angle is small in most cases, the default angle is set to zero in the following work, which means that the horizontal center line of the camera's image plane is parallel to the experiment platform, e.g. a table surface used for locating specific food.

After the configuration, the virtual camera has been fixed in the frame $\boldsymbol{W}$. In order to explicitly demonstrate this orientation process of the virtual camera, we use another object coordinate frame, in this case a camera coordinate frame $\boldsymbol{C} : (x_c, y_c, z_c)$, to determine the posture and position of the camera relative to the frame $\boldsymbol{W} : (X, Y, Z)$, as illustrated in the Figure 7. We define the pinhole of the virtual camera as the origin of the frame $\boldsymbol{C}$, the principal axis x_c parallel with the horizontal center line of the image plane of the camera, the principal axis y_c parallel with the vertical center line of the image plane, and the principal axis z_c coinciding with the camera's optical axis.

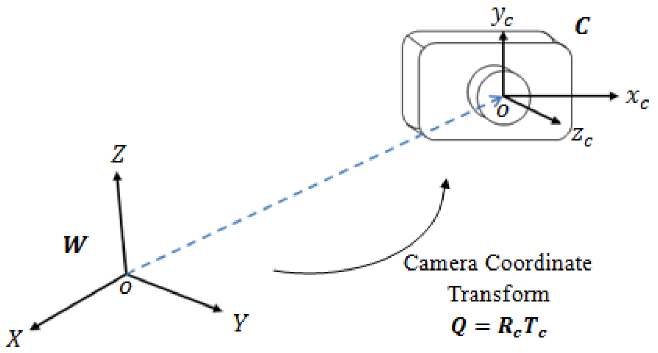

y_c
$\boldsymbol{C}$
x_c
o
z_c
Z
$\boldsymbol{W}$
o
X
Y
Camera Coordinate Transform
$\boldsymbol{Q} = \boldsymbol{R}_c\boldsymbol{T}_c$

Figure 7. A transformation between a world frame and a camera frame.

Once the frame $\boldsymbol{C}$ has been fixed, the *camera coordinate transformation* can be implemented. In order to simplify the mathematic manipulation of the virtual camera model in future work, people always transform the camera relative to the frame $\boldsymbol{W}$ to the frame $\boldsymbol{C}$. Following this step, all the geometric models in the frame $\boldsymbol{W}$ should be transformed along with the camera, to keep the relationship between the camera and all the geometric models in frame $\boldsymbol{W}$. This kind of transform is the rigid-body motion because we only implement translation and rotation transform here, which will not change the relationship between any two points in the virtual model. The camera coordinate transformation could also be implemented by matrix manipulation, similar to the world coordinate transformation. We define the camera coordinate transformation matrix as

$$\boldsymbol{Q} = \boldsymbol{R}_c\boldsymbol{T}_c \tag{2.10}$$

where $\boldsymbol{R}_c$ indicates the rotation matrix and $\boldsymbol{T}_c$ indicates the translation matrix, for the transform from the world frame $\boldsymbol{W}$ to the camera frame $\boldsymbol{C}$.

Camera coordinate transformation is a rigid-body motion; therefore, the relative position of the camera and the objects will not change during the camera coordinate transformation.

2.3.4 Projection in virtual camera model

When finishing the rigid-body motion from world frame $\boldsymbol{W}$ to camera frame $\boldsymbol{C}$, the main task is to obtain the 2D projection from the 3D virtual models. Here we also adopt the pinhole camera model to construct the virtual camera model, so that the images we obtain will be the perspective projection of the virtual objects, as illustrated in Figure 2. A point $\boldsymbol{X} = [X, Y, Z]^T$ in frame C is projected onto the image plane at the point

$$\boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{z}\begin{bmatrix} X \\ Y \end{bmatrix} \tag{2.11}$$

In homogeneous representation, the above equation can be written as

$$\overline{\boldsymbol{x}} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{f}{Z}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

We can rewrite the above equation as

$$\overline{\boldsymbol{x}} = \begin{bmatrix} \frac{f}{z} & 0 & 0 & 0 \\ 0 & \frac{f}{z} & 0 & 0 \\ 0 & 0 & \frac{f}{z} & 0 \end{bmatrix}\overline{\boldsymbol{X}} \tag{2.12}$$

$\boldsymbol{M} = \begin{bmatrix} \frac{f}{z} & 0 & 0 & 0 \\ 0 & \frac{f}{z} & 0 & 0 \\ 0 & 0 & \frac{f}{z} & 0 \end{bmatrix}$ is defined as the *perspective projection matrix*.

Furthermore, in virtual camera model construction, one must identify the field of vision for the camera. To determine the field of vision, as illustrated in Figure 8, parameters are required:

- focal length f
- angle α of vertical field of vision
- aspect ratio r, the value of image width a / image height b
- distance from camera to near plane
- distance from camera to far plane

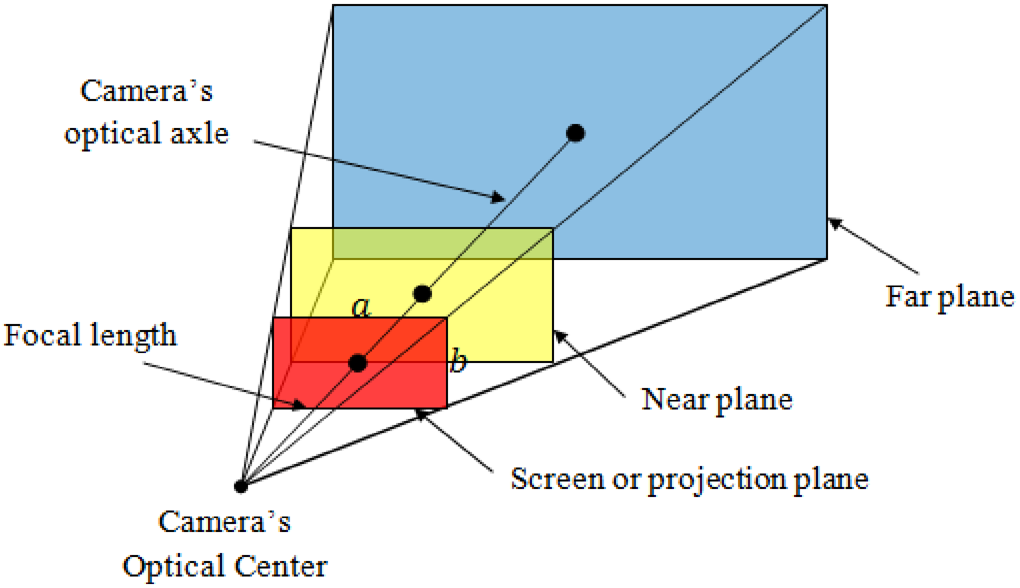


Figure 8. Identification of the field of vision of a fixed virtual camera.

For a virtual camera model, the angle of field of vision describes the angular extent of the image plane, which can be measured horizontally, vertically, or diagonally. These angles in the field of vision can identify the image (screen or projection) plane with a given focal length f. We also require the angle of vertical field of vision here because the horizontal field of vision will be determined by the vertical field of vision and another parameter, the aspect ratio. A single angle of field of vision of a camera can be written by

$$\theta = 2arctan[d/(2f)] \tag{2.13}$$

where f is the camera's focal length and d is the dimension corresponding to the angle θ.

In the case of the angle of vertical field of vision, the above equation can be rewritten as

$$\alpha = 2arctan[b/(2f)]$$

where α is the angle of vertical field of vision and b is the height of the image plane. Then, the height b can be obtained by the above equation

$$b = 2ftan(\alpha/2)$$

And the width a of the projection window can be fixed by

$$a = br$$

where r is the given aspect ratio.

In this way, the projection screen can be determined by the fixed values of a and b. In Figure 8, we can see a pyramidal tetrahedron determined by the screen and the fixed camera's optical center. The field of vision of a virtual camera is the part of the pyramidal tetrahedron which is limited by the near plane, the nearest plane that can be imaged, and the far plane, the furthest plane which can be imaged. Thus, one must provide the distances from the camera's optical center to the near plane and to the far plane in order to identify the field of vision of the virtual camera. [45]

2.4 CAMERA CALIBRATION

There are two world coordinate frames in our system: one is the real world coordinate frame, and the other is the virtual world coordinate frame. Before creating the virtual world, we must implement the camera calibration to identify the camera's intrinsic and extrinsic parameters relative to the real world coordinate frame. In our approach, we used a checkerboard approach based on an Matlab toolbox provided by Jean-Yves Bouguet [46].

At first, we calculated the camera's intrinsic parameters based on a total of twenty images of a planar checkerboard, partially shown in Figure 9. The major processes included images reading, corners extracting, and main calibration. The final result included all intrinsic parameters of the camera, including the focal length which was our primary interest.

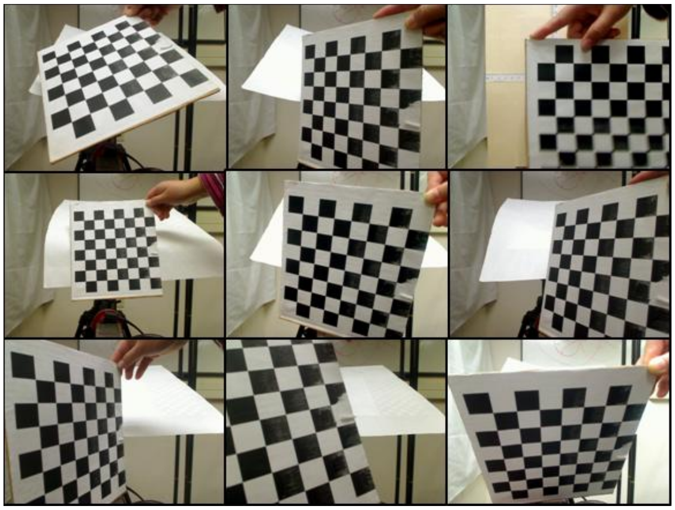


Figure 9. Checkerboard images used for camera intrinsic parameters calibration.

The next step was to construct the world coordinate and identify the camera's extrinsic parameters. Most processes were similar with the calibration of the intrinsic parameters. Figure 10 shows the world coordinate frame constructed based on a checkerboard which was placed on the table. Simultaneously, the coordinates of the camera's optical center and its orientation relative to the world frame were provided in the final results. Furthermore, we needed the position of the intersection point of the camera's optical axis and the table surface in the future process, so the checkerboard pattern card was adjusted on the table, satisfying the condition that a grid corner was presented on the camera's optical axis. And then, this grid corner's coordinates could be abstracted to describe the position of the intersection point. Figure 10 shows the world coordinate frame constructed by the checkerboard method, in which the camera's target point has been described. Unlike the first step, we needed to provide only one image to the toolbox in order to do the extrinsic parameters calibration.

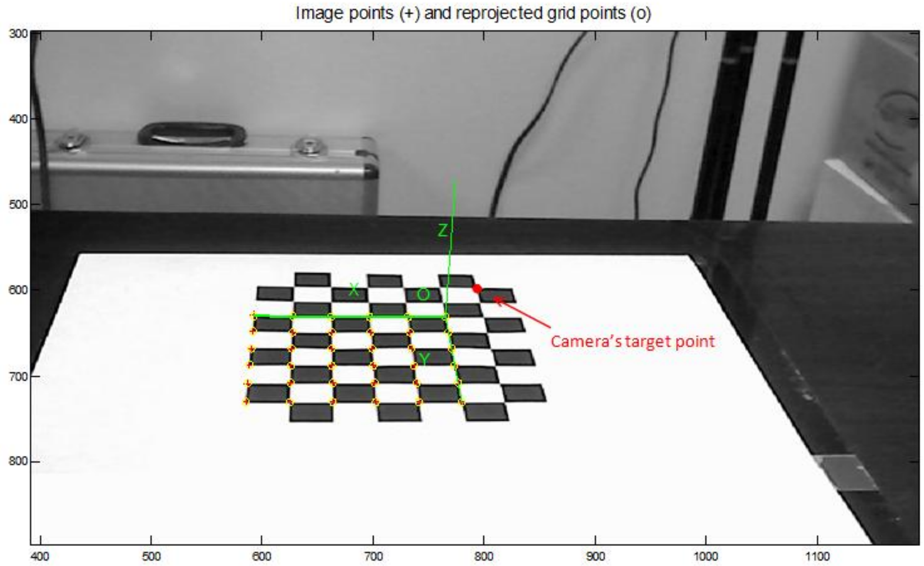


Figure 10. A world coordinate frame constructed by Checkerboard method.

3.0 RESEARCH DESIGN AND STRATEGY

The term virtual reality (VR) describes the use of computer-simulated environments to represent similar scenes in the real world as well as in imaginary worlds [47]. We propose a VR approach to simulate a food volume measurement process that cannot be accomplished easily in the real world using a single digital image as the input. The motivation for using VR in research design is described in Section 3.1, the mathematic model is presented in Section 3.2, and the application of the software based on the VR approach is discussed in Section 3.3.

3.1 MOTIVATION OF USING VR IN RESEARCH DESIGN

Once the correspondence between the real world frame and the camera frame has been established after the camera calibration, food volume can be estimated based on this correspondence. Our initial approach required human-computer interaction. For a food image, one must first select the image pixels that define the relevant dimensions (e.g., length, height, and/or diameter) and then use them to calculate the volume of the food. However, with this method, a number of obstacles prevented the accurate estimation of volume. First, the selection of image pixels was severely affected by image quality. For example, when the boundaries of the food were unclear, the measured dimensions became ambiguous. Second, in most cases, it was difficult to select a pair of points to represent the object’s height or radius. Figure 11(a) shows how a circular surface became an ellipse in most cases, and Figure 11(b) shows that selecting two points to represent the height of the hamburger produced ambiguous results. Third, the bottom surfaces of the food images were often difficult to extract due to occlusion, as illustrated in Figure 11(c). Finally, our analytical and experimental studies showed that when the food volume was computed from dimensional measurements, the volumetric errors increased significantly, especially for irregularly shaped objects. The reason for this was that when each dimension was measured separately, all the relative errors combined to create a large error in the final volume measurement.

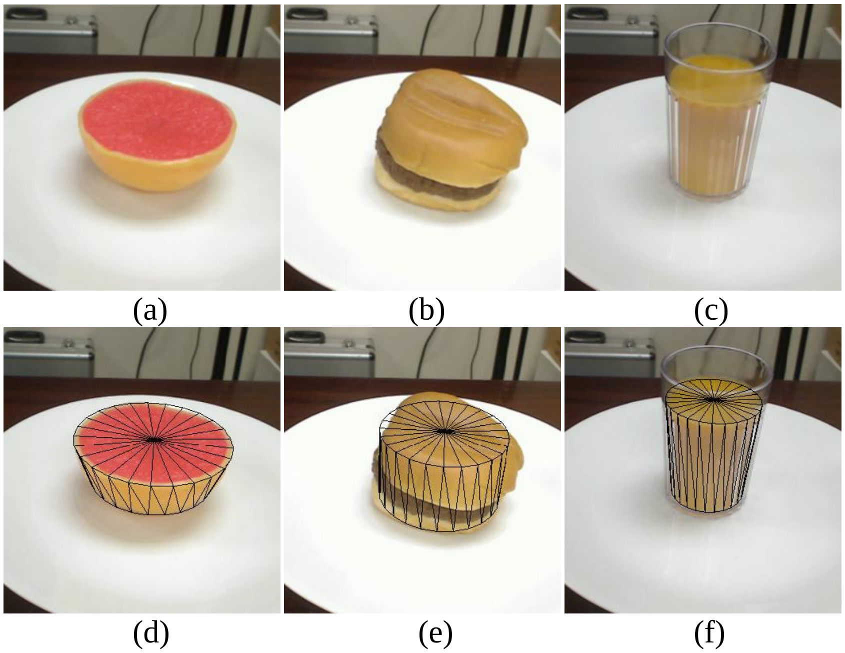


Figure 11. The original food images and the effects of images after volume estimation.

In order to estimate food volume more accurately, we designed the software based on the proposed VR approach. This software created a virtual world to simulate the real world and then created a number of 3D wireframe models to fit the specific food item within a digital image, as illustrated in the bottom row of Figure 11. By assessing the relationship between and on the volume of the virtual object, the software can calculate the volume of real foods. The relationship between these two worlds and the estimation process will be described in the remainder of this chapter. Using this method, the 3D volumetric model was directly with respect to the volume of specific food. Hence, dimensional measurements can be bypassed and the error-accumulation effect is avoided. Errors arising from occlusion and/or irregular shapes are also reduced because people can adjust the shape of the virtual object based on the whole picture and obtain the optimal fit and then get the results. Our experimental study has indicated that our VR system is highly accurate and robust in estimating the volume of both regularly and irregularly shaped foods.

3.2 MECHANISM OF THE VR APPROACH

3.2.1 Basic description of the VR system

The general idea of the proposed system is to use only one food image together with a virtual object built by computer graphics to approximate and estimate the food volume, as illustrated in Figure 11. The method is to create a virtual camera in the virtual world to simulate the camera in the real world, and make the two camera optical centers, as well as the two image planes, coincide. Inside the virtual world, virtual object can be built to simulate the interested object. When the images of virtual object and real object overlapped, the volume of the real object can be estimated by the volume of the virtual object and a scale ratio between the real world and the virtual world.

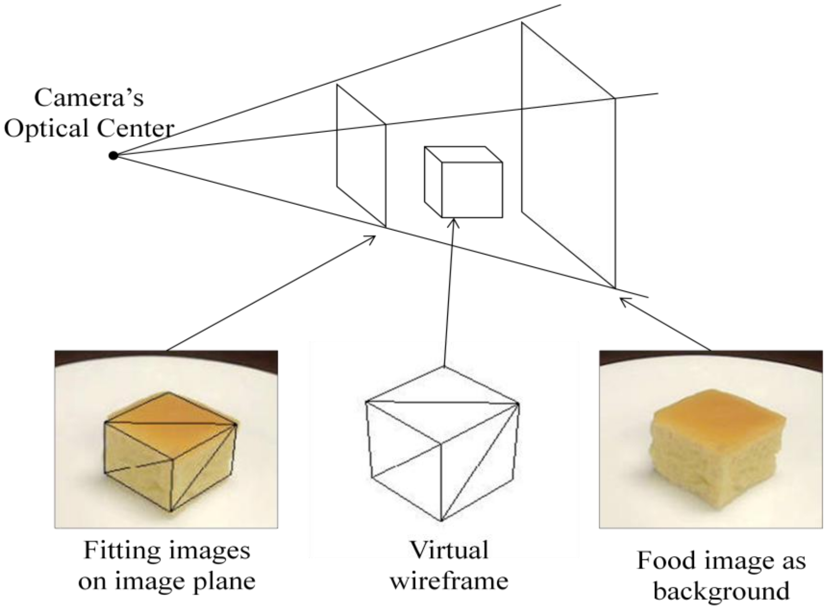


Figure 12. The proposed virtual camera model based on the VR approach.

In real application, we set the digital food image as the background of the scene for a virtual camera, and then the virtual objects will be constructed between the virtual camera's image plane and the background as shown in Figure 12. Without considering the virtual object, under a given intrinsic parameters setting, the virtual camera will provide us the same image which has been taken by the real camera. When the virtual object is considered, we are able to use a real camera and a virtual camera, respectively, to capture objects in the real and virtual worlds, and then make the two cameras coincide to create an integrated image. In the following mathematic deduction, we will combine both the two world coordinate frames and the two cameras to form a model with a single camera that can image the objects in both worlds.

In order to achieve this goal, we distinguish two world coordinate frames, one is the *virtual world coordinate frame* $\boldsymbol{W} : (X, Y, Z)$, and the other one is the *real world coordinate frame* $\boldsymbol{W'} : (X', Y', Z')$, as shown in Figure 13. In the whole process of this VR method, we always make the virtual camera's intrinsic parameters equivalent to the real camera, and at the mean time, we set the two camera coincide with each other, which means the two optical center and two image plane are overlapped completely. Thus, we will use a same *camera coordinate frame* $\boldsymbol{C} : (x, y, z)$, to indicate both the virtual camera coordinate frame and the real camera coordinate frame.

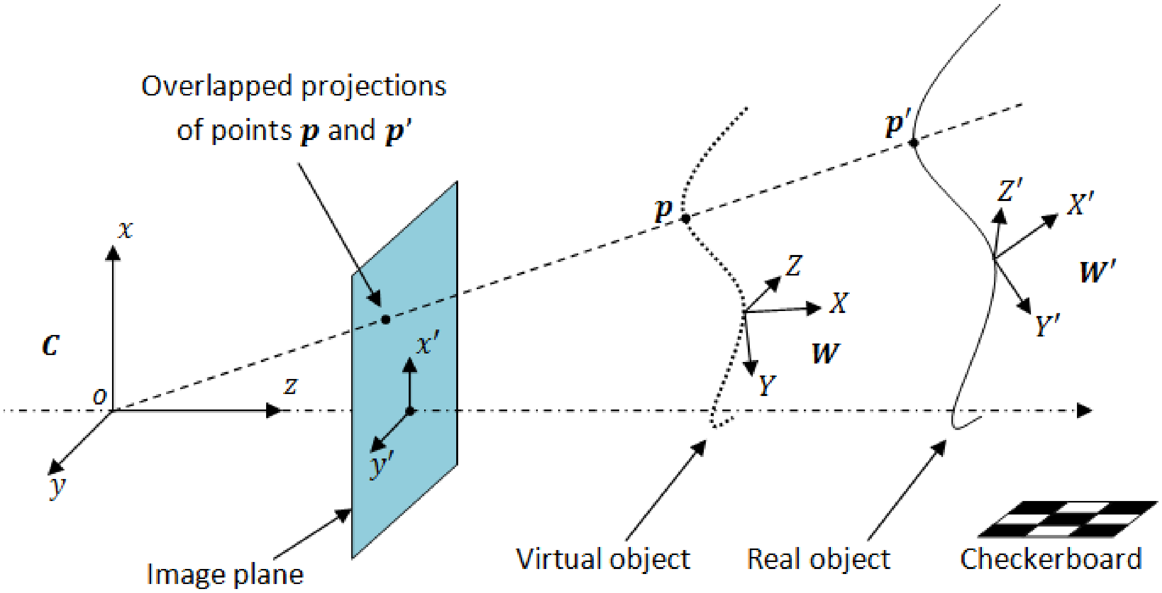


Figure 13. Geometric model of the VR approach.

Let us consider a generic point $p'$of the real object, with the coordinates $\boldsymbol{X'} = [X', Y', Z']^T$ relative to the frame W'. In order to implement camera coordinate transformation $\boldsymbol{Q'}$, we transfer $\boldsymbol{X'}$ into homogeneous coordinates. Then, after the camera coordinate transformation, we have the coordinates $\overline{\boldsymbol{x_{p'}}}$ of the point p' relative to the camera coordinate frame $\boldsymbol{C}$, written as

$$\overline{\boldsymbol{x_{p'}}} = \begin{bmatrix} x_{p'} \\ y_{p'} \\ z_{p'} \\ 1 \end{bmatrix} = \boldsymbol{Q'}\overline{\boldsymbol{X'}} \qquad (3.1)$$

Then, using (2.12), we will have the perspective projection of the point $\boldsymbol{x_{p'}}$ on the image plane $x'o'y'$,

$$\overline{\boldsymbol{x'_{p'}}} = \begin{bmatrix} x'_{p'} \\ y'_{p'} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f}{Z_{P'}} & 0 & 0 & 0 \\ 0 & \frac{f}{Z_{P'}} & 0 & 0 \\ 0 & 0 & \frac{f}{Z_{P'}} & 0 \end{bmatrix} \overline{\boldsymbol{x_{p'}}} = \frac{f}{z_{p'}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \overline{\boldsymbol{x_{p'}}}$$

Now, suppose we construct a virtual object together with the frame $\boldsymbol{W}$ and point p is a generic point in it. Similar to the above equation, the perspective projection of the point p on the image plane can be written by

$$\overline{\boldsymbol{x'_{p}}} = \begin{bmatrix} x'_{p} \\ y'_{p} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f}{z_{p}} & 0 & 0 & 0 \\ 0 & \frac{f}{z_{p}} & 0 & 0 \\ 0 & 0 & \frac{f}{z_{p}} & 0 \end{bmatrix} \overline{\boldsymbol{x_{p}}} = \frac{f}{z_{p}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \overline{\boldsymbol{x_{p}}}$$

where $\boldsymbol{x_p}$ is the coordinate of point p relative to the frame $\boldsymbol{C}$.

In order to make the projection of virtual object and real object overlap on the image plane, the system must satisfy

$$\boldsymbol{x'_p} = \boldsymbol{x'_{p'}}$$

which means

$$\frac{f}{z_p} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} = \frac{f}{z_{p'}} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{p'} \\ y_{p'} \\ z_{p'} \end{bmatrix}$$

The above equation can be reduced to

$$f \begin{bmatrix} \frac{x_p}{z_p} \\ \frac{y_p}{z_p} \\ 1 \end{bmatrix} = f \begin{bmatrix} \frac{x_{p'}}{z_{p'}} \\ \frac{y_{p'}}{z_{p'}} \\ 1 \end{bmatrix}$$

or in fraction equation form

$$\frac{x_p}{x_{p'}} = \frac{y_p}{y_{p'}} = \frac{z_p}{z_{p'}} \tag{3.2}$$

Then, if we can fix the generic point p, with its coordinates $\boldsymbol{x_p}$ relative to the frame C satisfying (3.2), the perspective projection of point p and point p' will overlap on the image plane.

Most important, when the coordinates $\boldsymbol{x_p}$ of a generic point of the virtual object and its corresponding coordinates $\boldsymbol{x_{p'}}$ of a generic point of the real object satisfies

$$\frac{x_p}{x_{p'}} = \frac{y_p}{y_{p'}} = \frac{z_p}{z_{p'}} = r$$

or in vector form

$$\boldsymbol{x_p} = r\boldsymbol{x_{p'}} \tag{3.3}$$

where r is an constant ratio, satisfying $1 > r > 0$, the volume of real object can be written as

$$v' = \iiint\limits_{v'} dx_{p'}dy_{p'}\,dz_{p'}$$

Use *Jacobian* matrix to change the variables of the above equation according to (3.3), we have

$$v' = \iiint\limits_{v} dx_p dy_p\,dz_p \begin{vmatrix} 1/r & 0 & 0 \\ 0 & 1/r & 0 \\ 0 & 0 & 1/r \end{vmatrix} = \frac{1}{r^3} \iiint\limits_{v} dx_p dy_p\,dz_p = v/r^3$$

where v indicates the volume of the virtual object.

In the above equation, we can obtain a fixed relationship between the volumes of the virtual object and the real object, written as

$$v' = v/r^3 \tag{3.4}$$

Thus, if we can construct a virtual camera model to simulate the real cameral model, make the two cameras coincide with each other, and also make the virtual geometric object satisfy (3.3), the volume of the real object can be estimated by the volume of the virtual geometric object and (3.4).

3.2.2 Construction method of the VR model

The objective of the method presented in this section is to construct a VR model that satisfies the three requirements for the volume estimation system, as indicated at the end of Section 3.2.1.

In practical applications, a virtual camera can be constructed with any intrinsic parameters as a real camera, so we start from the point that we have already had the two cameras that are identical. Thus, the problem need to be solved here is how to create a VR model, in which the two cameras are coincided with each other and the virtual geometric object satisfies (3.3). The method to solve this problem is to construct a virtual table on which virtual objects could be located, simulating the real foods on a real table. The details will be described as follows.

For a real camera model, we use the checkerboard method (Section 2.4) to calculate its extrinsic parameters. Since foods are essentially always located on a table to be imaged, the checkerboard calibration pattern should also be placed upon the table and then undertake the camera calibration. After the implement of the checkerboard method, the real world coordinate frame $\boldsymbol{W'}$ will be constructed, where the table's upper surface will be described as the plane $X'O'Z'$, as shown in Figure 14. The camera's optical center C position will also be described in the frame $\boldsymbol{W'}$ as

$$\boldsymbol{X_c'} = [X_c', Y_c', Z_c']^T.$$

The checkerboard method would also provide us with the position of the camera target point, which is the same as the intersection point of the camera's optical axis and the table surface, described in section 2.4. It is straightforward to translate the origin O' of the frame $\boldsymbol{W'}$ to the camera target point, and then use the new coordinate frame to describe the real world. This operation simplifies the identification of virtual camera model in the next step. Thus, in the following, we will use the coordinates $\boldsymbol{X'_c} = [X'_c, Y'_c, Z'_c]^T$ to describe the real camera's optical center, and use point $O' = [0,0,0]$ to indicate the real camera's target point, all relative to the frame $\boldsymbol{W'}$.

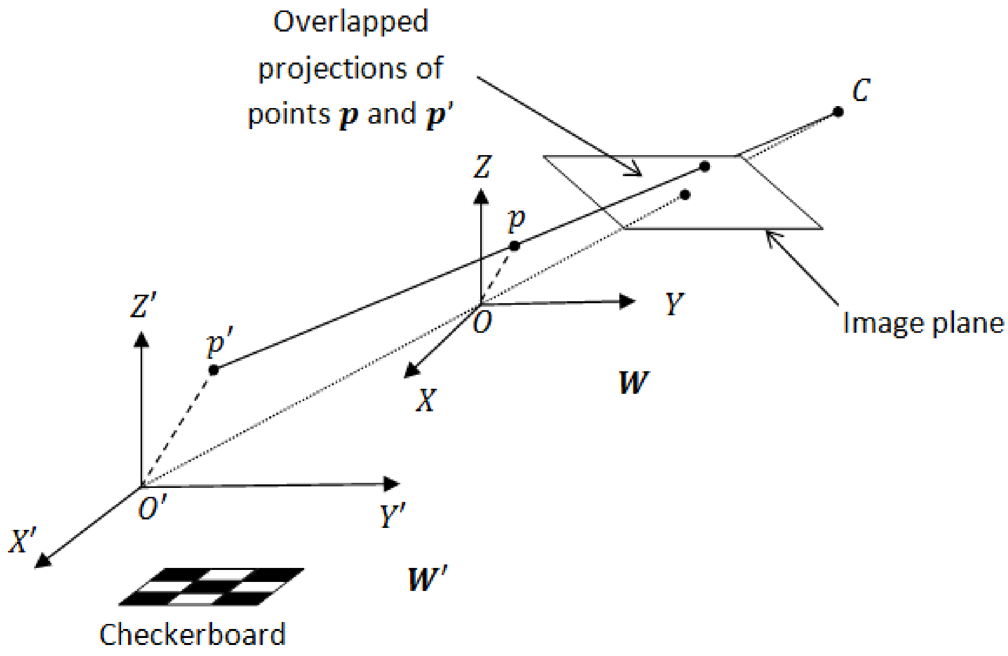


Figure 14. Construction method of VR model.

Now, a virtual camera model should be constructed and combined with the real camera model. As mentioned previously, the virtual camera's optical center and image plane will be located coincide with the ones of the real camera, which means that the two cameras have the same camera coordinate frame $\boldsymbol{C}$ here. Then, we set the origin O of the virtual coordinate frame, which is also the virtual camera's target point, on the real camera optical axis $O'C$, satisfies

$$\overline{OC} = R\overline{O'C}, 1 > R > 0$$

where R is a constant factor providing the proportional relationship between the virtual world and the real world. Since the constant ratio r in (3.4) also satisfies$1 > r > 0$, we are able to make $R = r$. Then, the above equation can be written as,

$$\overline{OC} = r\overline{O'C}, 1 > r > 0 \tag{3.5}$$

It is straightforward to obtain the coordinates $\boldsymbol{X'_o}$ of point O relative to frame $\boldsymbol{W}$', written by

$$\boldsymbol{X'_o} = (1-r)\boldsymbol{X'_c} = [(1-r)X'_c, (1-r)Y'_c, (1-r)Z'_c]^T$$

Then, based on the known point O, we construct the virtual frame $\boldsymbol{W}$, where the three principle axis X, $Y$and Z are parallel with the frame $\boldsymbol{W'}$ principle axis X', $Y'$and Z' of frame $\boldsymbol{W'}$, respectively. Actually, the plane XOY is the virtual table surface, which will be discussed at the end of this section.

Let us consider a generic point p' with the coordinates $\boldsymbol{X'_p}$ relative to the frame $\boldsymbol{W'}$ (see Figure 14). In homogeneous representation, the coordinates $\boldsymbol{X_p}$ of the same point relative to the frame $\boldsymbol{W}$, can be written as

$$\overline{\boldsymbol{X_p}} = \begin{bmatrix} X_p \\ Y_p \\ Z_p \\ 1 \end{bmatrix} = \boldsymbol{RT}\overline{\boldsymbol{X'_p}}\,,$$

where $\boldsymbol{R}$ and $\boldsymbol{T}$ are the rotation matrix and the translation matrix of the coordinate transformation, respectively.

Since the three principle axes of frame $\boldsymbol{W}$ are parallel with the three principle axes of frame $\boldsymbol{W'}$, there are no rotations between these two coordinates, which means that $\boldsymbol{R}$ is an identity matrix. Moreover, since the coordinates of origin O of frame $\boldsymbol{W}$ relative to frame $\boldsymbol{W'}$ is $\boldsymbol{X'_o} = [(1-r)X'_c, (1-r)Y'_c, (1-r)Z'_c]^T$, referring (2.5), the translation matrix can be written as

$$\boldsymbol{T} = \begin{bmatrix} 1 & 0 & 0 & -(1-r)X'_C \\ 0 & 1 & 0 & -(1-r)Y'_C \\ 0 & 0 & 1 & -(1-r)Z'_C \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.6}$$

Then, the coordinates $\boldsymbol{X_c}$ of the camera's optical center C relative to frame $\boldsymbol{W}$ can be written as

$$\overline{\boldsymbol{X_c}} = \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \boldsymbol{T}\overline{\boldsymbol{X'_c}} = \begin{bmatrix} 1 & 0 & 0 & (r-1)X'_C \\ 0 & 1 & 0 & (r-1)Y'_C \\ 0 & 0 & 1 & (r-1)Z'_C \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X'_c \\ Y'_c \\ Z'_c \\ 1 \end{bmatrix} = \begin{bmatrix} rX'_c \\ rY'_c \\ rZ'_c \\ 1 \end{bmatrix}.$$

Now, we have finished the construction of the virtual world coordinate frame. The next step is to translate both of the real world frame and the virtual world frame into the single camera frame $\boldsymbol{C}$, and consider how to construct the virtual geometric objects.

As mentioned previously in this section, there are no 3D rotations between frames $\boldsymbol{W}$ and $\boldsymbol{W'}$. Therefore, we can implement the same rotation matrix $\boldsymbol{R_c}$ on both of the two frames. The two translation matrixes of the camera coordinate transformation can be determined by the coordinates $\boldsymbol{X_c} = [rX'_c, rY'_c, rZ'_c]^T$ and the coordinates $\boldsymbol{X'_c} = [X'_c, Y'_c, Z'_c]^T$, both of which are the coordinates of the camera's optical centers, but relative to different frames. Then, similar to (3.6), we can write the two translation matrixes as

$$\boldsymbol{T_{VC}} = \begin{bmatrix} 1 & 0 & 0 & -rX'_C \\ 0 & 1 & 0 & -rY'_C \\ 0 & 0 & 1 & -rZ'_C \\ 0 & 0 & 0 & 1 \end{bmatrix}, \boldsymbol{T_{RC}} = \begin{bmatrix} 1 & 0 & 0 & -X'_C \\ 0 & 1 & 0 & -Y'_C \\ 0 & 0 & 1 & -Z'_C \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where $\boldsymbol{T_{VC}}$ provides the translation matrix between virtual world frame $\boldsymbol{W}$ and camera frame $\boldsymbol{C}$, and $\boldsymbol{T_{RC}}$ indicates the translation matrix between real world frame $\boldsymbol{W'}$ and camera $\boldsymbol{C}$.

Then, let us consider two generic point p' of a real object and point p of a virtual object, described in the frame $\boldsymbol{W'}$ and frame $\boldsymbol{W}$ as $\boldsymbol{X'_{p'}} = [X'_{p'}, Y'_{p'}, Z'_{p'}]^T$ and $\boldsymbol{X_p} = [X_p, Y_p, Z_p]^T$ respectively. In order to describe the relationship between the virtual object and real object in frame $\boldsymbol{C}$, as expressed by (3.3), we use the rotation matrix $\boldsymbol{R_c}$ and the two translation matrix $\boldsymbol{T_{RC}}$ and $\boldsymbol{T_{VC}}$ to obtain the coordinates $\boldsymbol{x_{p'}}$ and $\boldsymbol{x_p}$ of the two points relative to the frame $\boldsymbol{C}$, written by

$$\overline{\boldsymbol{x_{p'}}} = \begin{bmatrix} x_{p'} \\ y_{p'} \\ z_{p'} \\ 1 \end{bmatrix} = \boldsymbol{R_C T_{RC}} \overline{\boldsymbol{X'_{p'}}} = \boldsymbol{R_C} \begin{bmatrix} 1 & 0 & 0 & -X'_C \\ 0 & 1 & 0 & -Y'_C \\ 0 & 0 & 1 & -Z'_C \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X'_{p'} \\ Y'_{p'} \\ Z'_{p'} \\ 1 \end{bmatrix} = \boldsymbol{R_C} \begin{bmatrix} X'_{p'} - X'_c \\ Y'_{p'} - Y'_c \\ Z'_{p'} - Z'_c \\ 1 \end{bmatrix} \tag{3.7}$$

$$\overline{\boldsymbol{x_p}} = \begin{bmatrix} x_p \\ y_p \\ z_p \\ 1 \end{bmatrix} = \boldsymbol{R_C T_{VC}} \overline{\boldsymbol{X_p}} = \boldsymbol{R_C} \begin{bmatrix} 1 & 0 & 0 & -rX'_C \\ 0 & 1 & 0 & -rY'_C \\ 0 & 0 & 1 & -rZ'_C \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_p \\ Y_p \\ Z_p \\ 1 \end{bmatrix} = \boldsymbol{R_C} \begin{bmatrix} X_p - rX'_c \\ Y_p - rY'_c \\ Z_p - rZ'_c \\ 1 \end{bmatrix} \tag{3.8}$$

As described in Section 3.2.1, if and only if the coordinates $\boldsymbol{x_p}$ of a generic point p of the virtual object and its corresponding coordinates $\boldsymbol{x_{p'}}$ of a generic point p' of the real object satisfy $\boldsymbol{x_p} = r\boldsymbol{x_{p'}}$ (3.3), the real object volume can be estimated by (3.4). Therefore, we must construct a virtual object that can satisfy (3.3), described as

$$\boldsymbol{R_C} \begin{bmatrix} X_p - rX'_c \\ Y_p - rY'_c \\ Z_p - rZ'_c \\ 1 \end{bmatrix} = \boldsymbol{R_C} \begin{bmatrix} r(X'_{p'} - X'_c) \\ r(Y'_{p'} - Y'_c) \\ r(Z'_{p'} - Z'_c) \\ 1 \end{bmatrix}$$

Multiplying $\boldsymbol{R_C}^{-1}$ on both sides, we obtain

$$\begin{bmatrix} X_p - rX'_c \\ Y_p - rY'_c \\ Z_p - rZ'_c \\ 1 \end{bmatrix} = \begin{bmatrix} r(X'_{p'} - X'_c) \\ r(Y'_{p'} - Y'_c) \\ r(Z'_{p'} - Z'_c) \\ 1 \end{bmatrix}$$

or in the form of fraction equations

$$\frac{X_p - rX_c'}{X_{p'}' - X_c'} = \frac{Y_p - rY_c'}{Y_{p'}' - Y_c'} = \frac{Z_p - rZ_c'}{Z_{p'}' - Z_c'} = r \tag{3.9}$$

It can be found that the above equation is satisfied when the generic point p of the virtual object with coordinates $\boldsymbol{X_p}$ has been related with the corresponding generic point p' of the real object with the coordinates $\boldsymbol{X'_{p'}}$ as

$$\boldsymbol{X_p} = \begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} = r \begin{bmatrix} X'_{p'} \\ Y'_{p'} \\ Z'_{p'} \end{bmatrix} = r\boldsymbol{X'_{p'}} \tag{3.10}$$

Satisfying the above constrains, the image of virtual object will perfectly fit the image of the real object, and similar to (3.4), the volume of the real object v' can be obtained by the volume of the virtual object v with the equation (3.4).

Since we have set the table surface as the plane $X'O'Y'$ and set the axis Z' to be perpendicular to the table surface in the upward direction, the generic point p' of the real object on the table surface has a positive Z' value. Thus, according to (3.5) and $r > 0$, it is necessary to set the Z value of any point of the virtual object larger than or equal to 0, which means that we "put" the virtual object on the "virtual table surface", the plane XOY.

A question arises whether we can construct a virtual object satisfying (3.10). The answer is yes. The reason lies in that the software only provides typical geometric objects to estimate the volume of the real food in a digital image. When we adjust the virtual object in the virtual world until comprehensively the images of the virtual object and an interested real object are overlapped, we can guarantee that their relationship will satisfy (3.10). The reason is that in the estimation process, we use typical geometric object to fit an approximately typical geometric object. If they did not satisfy (3.10), the virtual object would be an anomalistic 3D object which is in conflict with the fact that the software only provides typical geometric objects.

3.3 APPLICATION OF THE VR SYSTEM

This section describes how the volume-estimation software constructed using the VR approach is implemented. Figure 15 below presents an overview of the system. It consists of two sections, one for the pre-estimation process and the other for the estimation process. The pre-estimation process includes image acquisition, calibration of camera parameters, and virtual camera modeling and construction. The estimation process includes virtual object manipulation, food volume estimation, and data storage.

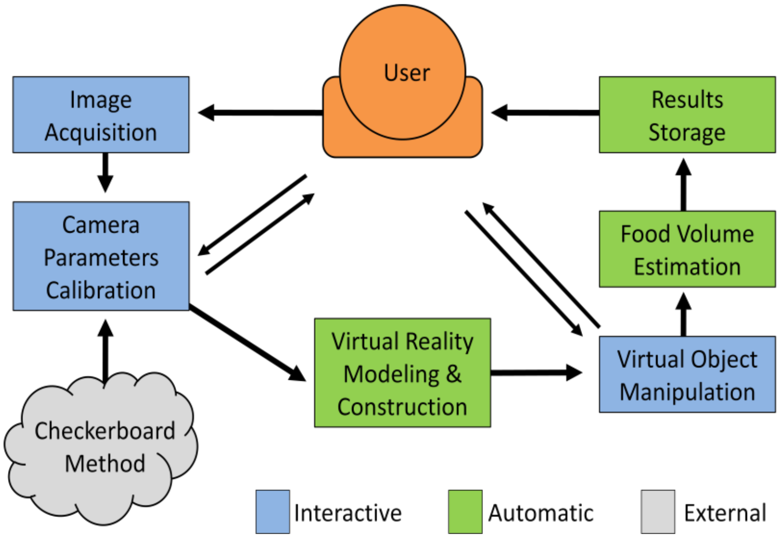


Figure 15. Volume estimation system based on the VR approach.

3.3.1 Pre-estimation process

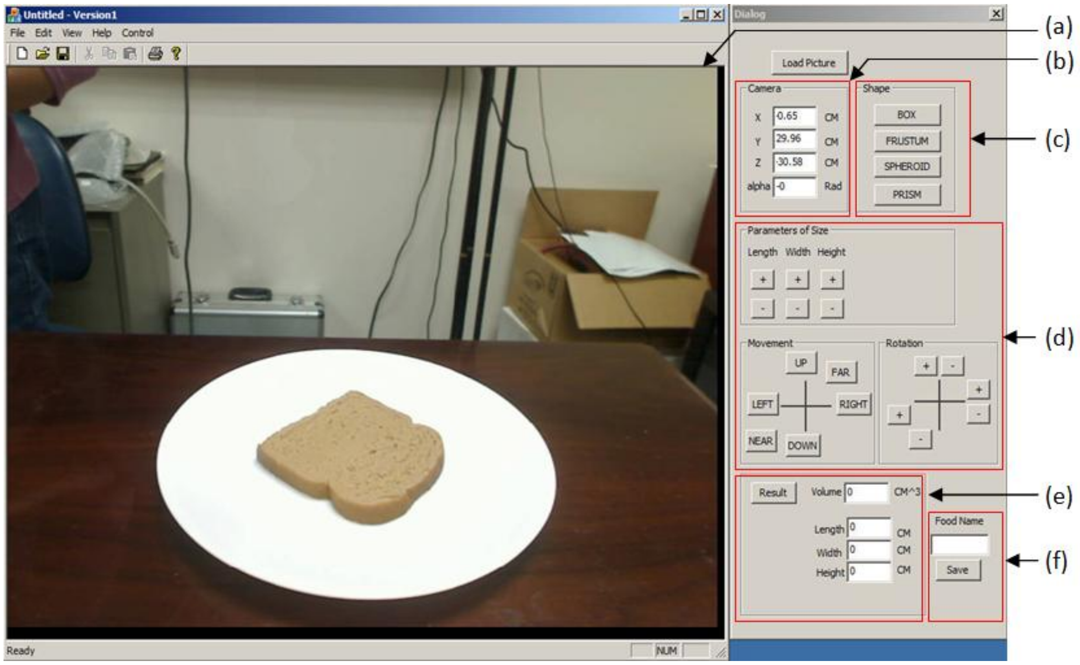
As the first part of the food volume estimation system, the pre-estimation process attempts to: 1) calibrate the parameters of the real camera, 2) create a virtual camera identical to the real camera, and 3) build a virtual environment that can be used to implement manipulation of virtual objects.

We used the checkerboard method, introduced in Section 2.4, to calibrate camera parameters, including the focal length, angle of field of vision, target point position, and position of optical center. All the camera parameters obtained with this method were relative to the real-world coordinate frame, in which the upper table surface was described as a plane XOY, as stated in Section 3.2.2 above. Also, we had to refer to the camera's user manual to find the specific dimensions of the camera’s image plane.

The second part of the pre-estimation process involved construction of the virtual-camera model. Section 2.3.3 has defined the four parameters required to specify a virtual camera. In practice, we set the angle of field of vision in the virtual camera model to that of the real camera model. The aspect ratio was calculated using the image width and height obtained from the camera user manual. Finally, we set the near plane to the camera’s optical center to coincide with the image plane, and set the distance from the far plane to the camera’s optical center, ensuring that it was long enough for the view field to contain any virtual geometric object required for use with the application.

As the final step of the pre-estimation process, the construction of the virtual coordinate frame followed the method described in Section 3.2. In the real application, both the real camera target point and the virtual camera target point have been translated to the origin of the coordinate frame, relative to the virtual world frame and the real world frame, respectively. Then, presuming that the given position of the real camera’s optical center is $\boldsymbol{X}_{\boldsymbol{c}}' = [X', Y', Z']^T$

relative to the real world frame $\boldsymbol{W'}$, we should provide a proper ratio R, to obtain the position of the virtual camera's optical center $\boldsymbol{X_c} = [rX', rY', rZ']^T$ relative to the virtual world frame $\boldsymbol{W}$. For greater convenience, the value of R was set inside the software and the user only needed to input the position of the real camera's optical center. Then, a specific digital food image was loaded into the operation window as the background of the scene for the virtual camera. The settings of camera position and virtual world background are presented in Figure 16.



X -0.65 CM
Y 29.96 CM
Z -30.58 CM
alpha -0 Rad

Volume 0 CM^3
Length 0 CM
Width 0 CM
Height 0 CM

Food Name

Figure 16. A screen shot of the interface of our software: (a) Operation window; (b) Camera setting panel; (c) Geometric class selection panel; (d) Virtual object manipulation panel; (e) Result Panel; (f) Result storage Panel.

Although the value of r doesn't affect the mathematic model used to described this VR system, it should be set as a proper ratio, say $r = 0.1$, because if the r is too big, the virtual object will be hidden by the digital image and if the r is too small, the virtual camera may not show the images of all the required virtual objects during the experiment.

3.3.2 Estimation process

This section describes the estimation process, including virtual object manipulation, food volume estimation, and feedback with the proposed system.

3.3.2.1 Virtual model manipulation

Virtual object manipulation is an essential component of the entire estimation system. To approximate the food item of interest, the software provides adjustable and rotatable virtual geometric wireframes that can fit real objects presented on digital images, as described in Section 3.2. And then, when an excellent fitting is achieved between a virtual wireframe and an object in the original image, we were able to use the (3.4) to estimate the food volume.

The food volume-estimation software partitioned the virtual objects into *geometric classes*, each with its own set of parameters. The software provided four geometric classes including the rectangular box, the conical frustum, the spheroid, and the triangular prism. After loading a real food image, users select the geometric class that is most similar to the food item in the digital image. Figure 17 below presents the four geometric classes.

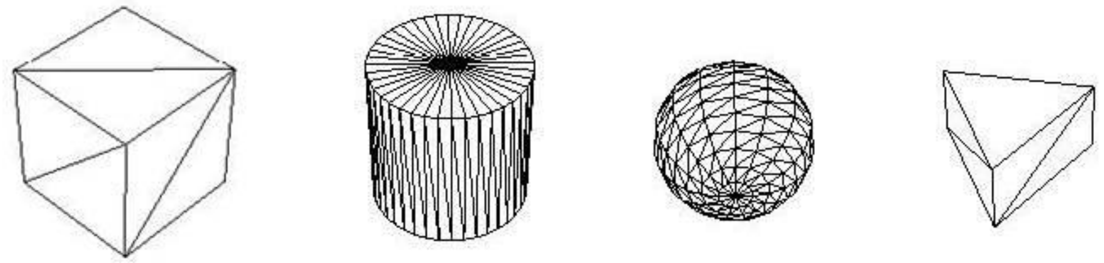


Figure 17. Four geometric classes of virtual wireframe provided by the VR system.

Once a geometric class is selected, a default wireframe corresponding to the selection will appear on the surface of the virtual table, as illustrated at the end of Section 3.2.2. Then, users can use the control panel (Figure 16), the mouse, or the keyboard to deform, scale, move or rotate the object on the virtual table surface until a good fit with the target food is achieved. All these manipulations are carried out using the transform matrix introduced in Section 2.3.3. Care must be taken that the virtual object always be located on the virtual plane XOY during the adjustment process. Figure 18 shows an example of virtual model manipulation, using a conical frustum model to fit half a grapefruit inside a digital image.

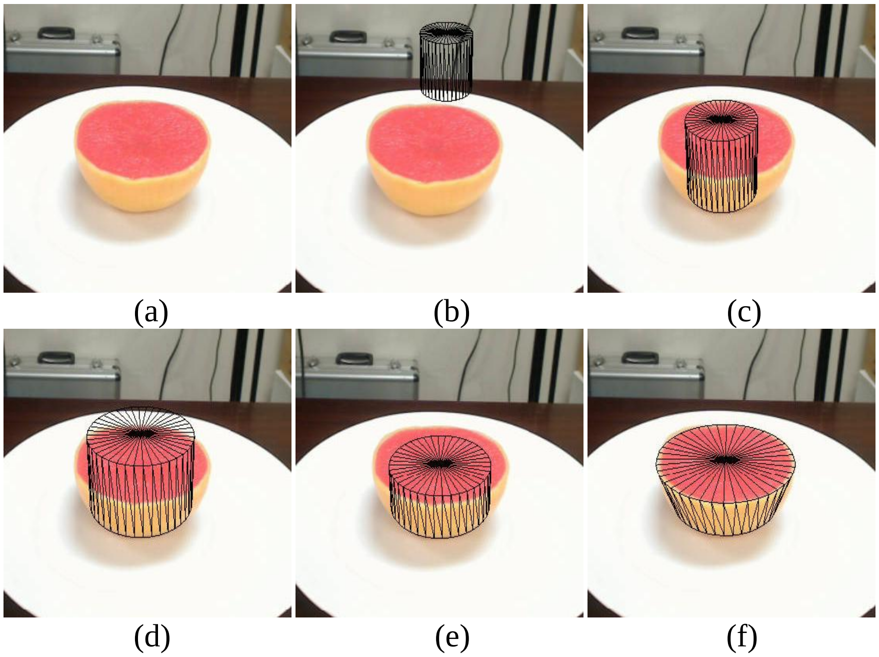


Figure 18. An example of virtual model manipulation: (a) Original image of half a grapefruit; (b) A selected conical frustum wireframe is located on the virtual table; (c) The wireframe has been dragged into the grapefruit; (d) The radius of bottom surface is adjusted; (e) The height is adjusted; (f) The ratio of the radius of top surface over the radius of bottom surface is adjusted and an excellent fitting is achieved.

3.3.2.2 Volume estimation

This section explains how the software carried out volume estimation. After manipulating the virtual objects, we obtained the overlapped images of the virtual object and the specific object presented in the operation window (Figure 16), indicating that the proportional relationship between the two objects had been obtained. Once this occurred, the volume of real objects could be obtained using the ratio R and the volume of the virtual object using (3.4). Formulas to calculate the volumes of the four virtual geometric classes are listed below.

The volume of a rectangular box is

$$v = abc \tag{3.11}$$

where a, $b,$ and c indicate the dimensions of the rectangular box.

The volume of a conical frustum is

$$v = \pi \left[\frac{r(1+\beta)}{2}\right]^2 h \tag{3.12}$$

where r and h indicate, respectively, the radius of the bottom surface and the height of the conical frustum, and β is the ratio equal to the radius of the upper surface over the radius of the bottom surface of the conical frustum.

The volume of a spheroid is

$$v = \frac{4}{3}\pi r_1 r_2 r_3 \tag{3.13}$$

where $r_1,\ r_2$ and r_3 are the lengths of the three semi-axes.

And the volume of a triangular prism is

$$v = \frac{1}{2}mnh \tag{3.14}$$

where m and n are the lengths of hemline and altitude of the triangular surfaces, and h is the height of the triangular prism.

Then, substituting equation (3.11), (3.12), (3.13) or (3.14) into (3.4), we can obtain the estimated volume of the real object, written as

$$v^{'} = v/r^3$$

where r is the constant ratio indicates the proportional relationship between the virtual world and the real world (Section 3.2.2).

Our system has the ability to do the volumetric estimation automatically. Together with the final result of the estimated volume, values for geometric parameters of the real object are also presented on the result panel (Figure 16).

3.3.2.3 Feedback system

The existing camera calibration method is essentially an open-loop system in which a checkerboard feature pattern produces calibration parameters. The accuracy of these camera parameters is not checked, which may introduce error. In order to reduce this kind of error, we propose a feedback system (Figure 19). At first, an initial rough estimate of the camera parameters is provided by the checkerboard method to the system, enabling the camera model to estimate the area and principal axes of a regular dinner plate expressed by a vector P'. The difference between P' and its true value P produces a multidimensional error e which adjusts the camera parameters for creating the new estimate of P'. The feedback system is repeated continually until $|e|$ is minimized, at which point the parameters are then utilized to estimate the volume of real food in the image.

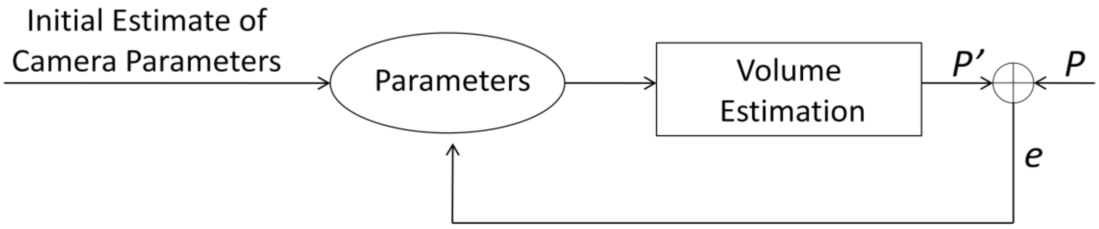


Figure 19. Proposed closed-loop calibration system.

In practice, we use focal length as the controllable parameter, which affects the image of virtual objects, according to (2.1). Actually, the value of focal length f obtained by the checkerboard method is described as a range $[a, b]$. Therefore, the feedback system has been designed to scan the range $[a, b]$ for the optimal value of f that leads to the minimum $|e|$. Once this value has been found, the virtual camera's focal length is set as the optimal value of f to estimate food volume.

In order to prove the feasibility of the proposed feedback system, we performed an experiment in which the focal length f was changed in small steps while a fitting of the plate area was maintained and $e = P' - P$ was evaluated. We found that varying f brought e across zero in a nearly linear fashion (left panel in Figure 20). An excellent fitting was also achieved between the virtual plate and the plate in the original image at $e \approx 0$. This experiment demonstrated that the scanning method can be used to obtain the optical focal length that will reduce error of the estimators. Thus, we constructed the feedback system based on this scanning method. Section 4.2 will show the volume estimation results of the proposed feedback system, which outperformed the results of none feedback system by a large margin.

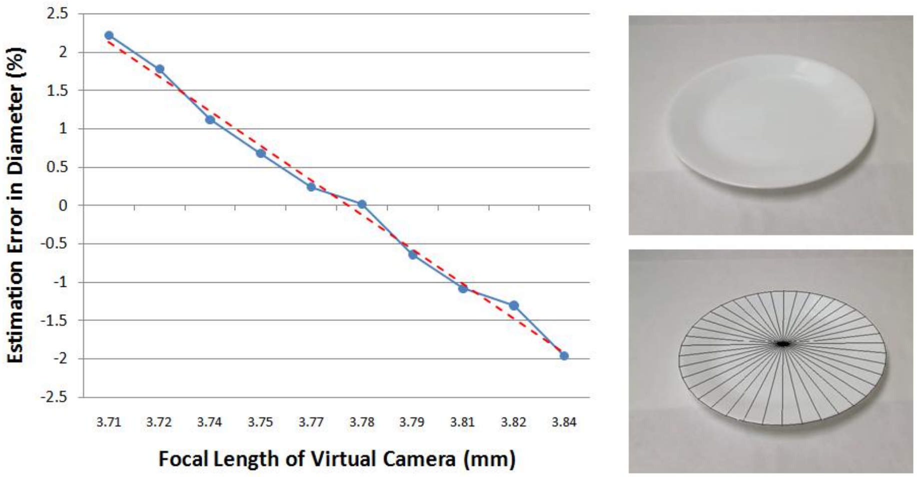


Figure 20. Results of the experiment proving the feasibility of the feedback system: (Left) estimation error vs. focal length; (Top right) Original image of a circular plate; (Bottom right) Overlapped images of the circular plate and the virtual wireframe when the estimation error is zero.

4.0 EXPERIMENTAL DESIGN AND RESULTS

This section describes the experimental design and results of our VR system. Section 4.1 describes the experiment design. Section 4.2 discusses the findings of our comparative experiments with and without the use of the feedback system. Next, Section 4.3 presents experiments using different camera positions. Finally, Section 4.4 discusses the effect of different object positions within the food image on the estimation accuracy.

4.1 EXPERIMENTAL DESIGN

Experiments using twenty-one regularly shaped objects and food replicas were designed to test the performance of the proposed approach and system. The selected objects include a small box, a big box, a cornbread, a cream cake, a piece of chicken breast, rice, a slice of bread, a baked potato, half of a hard-boiled egg, half of a grapefruit, a hamburger, a scoop of ice cream, a bowl of jello, a glass of orange juice, a glass of milk, a slice of onion, an orange and a peach (shown in Figure 21). Three different groups of experiments were performed to test the effect of three parts of the design on the outcome, including the feedback, the camera position and the location of subject item. Figure 22 shows the equipment used in our experiment.

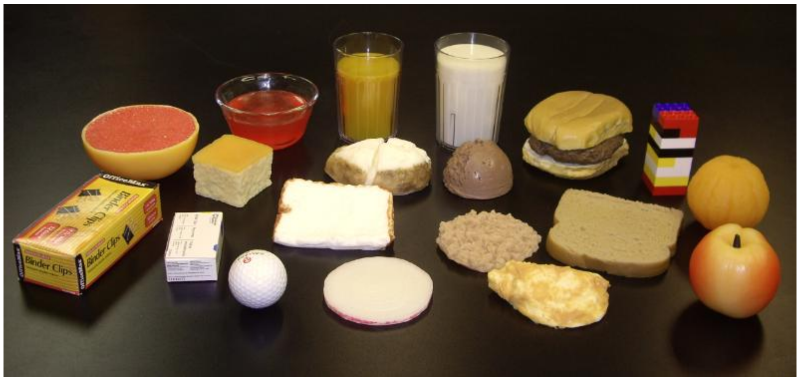


Figure 21. Objects used in experiments.

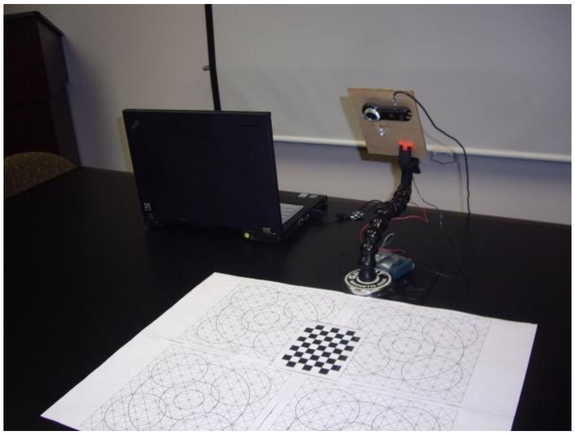


Figure 22. Equipment used in experiments.

In the statistical analysis of the experiment results, the mean value and standard deviations (STD) were used to describe the accuracy and the stability of the estimators. To precisely quantify measurement errors, Root Mean Squared Error (RMSE) was also used in the experiments to measure the differences between an estimator and the values actually observed. The RMSE is defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(\frac{\hat{V}_j - V}{V}\right)^2} \times 100 \tag{4.1}$$

where n is the sample size, $\hat{v}_j$ is the estimated value, and v is the true value, measured in centimeters.

For food replicas, the true volumes were measured using the water displacement method, as illustrated in Figure 23.

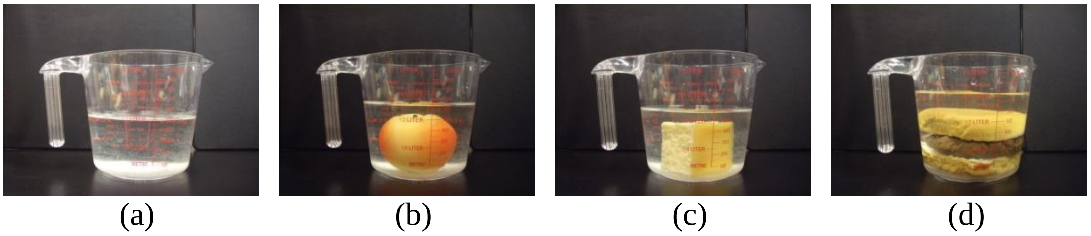


Figure 23. Volume measurements of artificial food models using water displacement method: (a) Baseline; (b) Peach; (c) Cornbread; (d) Hamburger.

4.2 EXPERIMENT WITH AND WITHOUT FEEDBACK SYSTEM

An experiment was performed to compare the results of using the open-loop system vs. the closed-loop system. The volumes of eighteen food items were estimated based on readings from four different camera positions. The camera was set to simulate a reading taken by patient sitting in front of the table on which the items were placed. The angle α between the plane of the camera image and the plane of the table was between $25°$ and $45°$. Parameters of the camera were obtained using the checkerboard method (Section 2.4). The range of focal length obtained for the camera was between 3.6596 cm and 3.7940 cm. Table 1, below, lists the coordinates of the four different camera optical points relative to the world coordinate frame constructed by the checkerboard method.

Table 1. Camera's extrinsic parameters for four positions.

	Coordinates of Camera Optical Center			
	x coordinate (mm)	y coordinate (mm)	z coordinate (mm)	α (°)
Camera Position 1	36.09	536.20	261.067	25.96
Camera Position 2	46.77	462.30	256.46	29.02
Camera Position 3	25.42	324.25	212.36	33.22
Camera Position 4	40.50	303.94	239.07	38.19

For the non-feedback system, we used the average value of the focal length range to serve as the virtual camera's focal length. For example, for a range between 3.6596 mm and 3.7940 mm, we have

$$f = \frac{3.6596 + 3.7940}{2} = 3.7268 \text{ (mm)}$$

For the feedback system, we used the diameter of a regular-sized dinner plate in the image as feedback to estimate volumes, as explained in Section 3.4. The estimation results for both the non-feedback system and feedback system are listed below in Tables 2-5. The RMSE in the volume estimation results for all the samples are shown in Figure 24.

Table 2. Volume estimates derived using non-feedback system with camera positions 1 and 2.

Estimated Object	True Value (cm^3)	Camera Position 1		Camera Position 2	
		Estimated Value (cm^3)	Relative Error (%)	Estimated Value (cm^3)	Relative Error (%)
Small box	56.65	64.75	14.30	63.84	12.69
Big box	248.57	284.86	14.60	283.57	14.08
Bread	106	126.17	19.03	127.69	20.46
Cornbread	93	94.58	1.70	101.86	9.53
Cream cake	93.67	99.41	6.13	86.77	-7.37
Chicken breast	64	44.84	-29.94	39.68	-38.00
Rice	52.33	31.45	-39.91	29.43	-43.75
Potato	112.67	161.18	43.05	143.29	27.18
Egg	20.67	17.97	-13.06	20.56	-0.53
Grapefruit	272	319.10	17.32	309.54	13.80
Hamburger	307.67	341.17	10.89	336.72	9.44
Ice Cream	79.67	81.01	1.68	77.92	-2.20
Jello	109.67	139.31	27.03	137.15	25.06
Juice	180	207.11	15.06	217.29	20.72
Milk	240	308.59	28.58	306.79	27.83
Onion Slice	32.63	26.42	-19.02	25.11	-23.04
Orange	151.67	163.55	7.83	167.74	10.60
Peach	151.67	163.55	7.83	172.01	13.41

Table 3. Volume estimates derived using non-feedback system with camera positions 3 and 4.

Estimated Object	True Value (cm^3)	Camera Position 3		Camera Position 4	
		Estimated Value (cm^3)	Relative Error (%)	Estimated Value (cm^3)	Relative Error (%)
Small box	56.65	59.70	5.38	63.95	12.88
Big box	248.57	265.70	6.89	289.77	16.57
Bread	106	119.16	12.42	125.71	18.59
Corn Bread	93	90.34	-2.86	99.37	6.85
Cream cake	93.67	86.95	-7.17	103.01	9.97
Chicken breast	64	36.64	-42.75	51.79	-19.07
Rice	52.33	40.92	-21.81	35.24	-32.65
Potato	112.67	135.39	20.16	144.67	28.40
Egg	20.67	17.69	-14.43	21.77	5.30
Grapefruit	272	278.67	2.45	337.07	23.92
Hamburger	307.67	322.65	4.87	372.26	20.99
Ice Cream	79.67	75.57	-5.14	78.22	-1.82
Jello	109.67	130.57	19.06	144.68	31.92
Juice	180	195.15	8.42	215.23	19.57
Milk	240	272.69	13.62	308.67	28.61
Onion Slice	32.63	27.24	-16.53	26.09	-20.04
Orange	151.67	163.55	7.83	172.01	13.41
Peach	151.67	167.74	10.60	176.34	16.27

Table 4. Volume estimates derived using feedback system with camera positions 1 and 2.

Estimated Object	True Value (cm^3)	Camera Position 1		Camera Position 2	
		Estimated Value (cm^3)	Relative Error (%)	Estimated Value (cm^3)	Relative Error (%)
Small box	56.65	55.61	-1.83	58.57	3.38
Big box	248.57	252.05	1.40	247.52	-0.42
Bread	106	96.62	-8.85	106.67	0.63
Corn Bread	93	87.86	-5.53	86.49	-7.01
Cream cake	93.67	101.75	8.62	105.02	12.12
Chicken breast	64	50.10	-21.72	50.10	-21.71
Rice	52.33	41.38	-20.93	46.82	-10.52
Potato	112.67	128.61	14.15	139.21	23.55
Egg	20.67	18.04	-12.71	17.64	-14.64
Grapefruit	272	262.25	-3.58	267.70	-1.58
Hamburger	307.67	290.37	-5.62	293.11	-4.73
Ice Cream	79.67	80.05	0.48	81.19	1.90
Jello	109.67	122.74	11.92	108.99	-0.62
Juice	180	186.04	3.35	188.04	4.47
Milk	240	263.74	9.89	262.26	9.28
Onion Slice	32.63	26.52	-18.73	26.16	-19.81
Orange	151.67	159.43	5.11	155.37	2.44
Peach	151.67	143.63	-5.30	159.43	5.11

Table 5. Volume estimates derived using feedback system with camera positions 3 and 4.

Estimated Object	True Value (cm^3)	Camera Position 3		Camera Position 4	
		Estimated Value (cm^3)	Relative Error (%)	Estimated Value (cm^3)	Relative Error (%)
Small box	56.65	53.30	-5.91	56.55	-0.17
Big box	248.57	236.52	-4.85	246.18	-0.96
Bread	106	99.57	-6.07	104.92	-1.02
Corn Bread	93	84.75	-8.87	81.45	-12.42
Cream cake	93.67	95.40	1.85	90.86	-3.00
Chicken breast	64	46.94	-26.65	50.55	-21.01
Rice	52.33	45.26	-13.51	37.53	-28.27
Potato	112.67	132.32	17.44	126.12	11.93
Egg	20.67	18.72	-9.43	17.61	-14.82
Grapefruit	272	249.89	-8.13	281.97	3.67
Hamburger	307.67	318.70	3.58	285.56	-7.19
Ice Cream	79.67	71.36	-10.44	76.20	-4.36
Jello	109.67	112.46	2.55	121.08	10.41
Juice	180	184.02	2.23	194.49	8.05
Milk	240	255.06	6.28	266.71	11.13
Onion Slice	32.63	28.64	-12.21	27.24	-16.53
Orange	151.67	159.43	5.11	155.37	2.44
Peach	151.67	147.47	-2.77	159.43	5.11

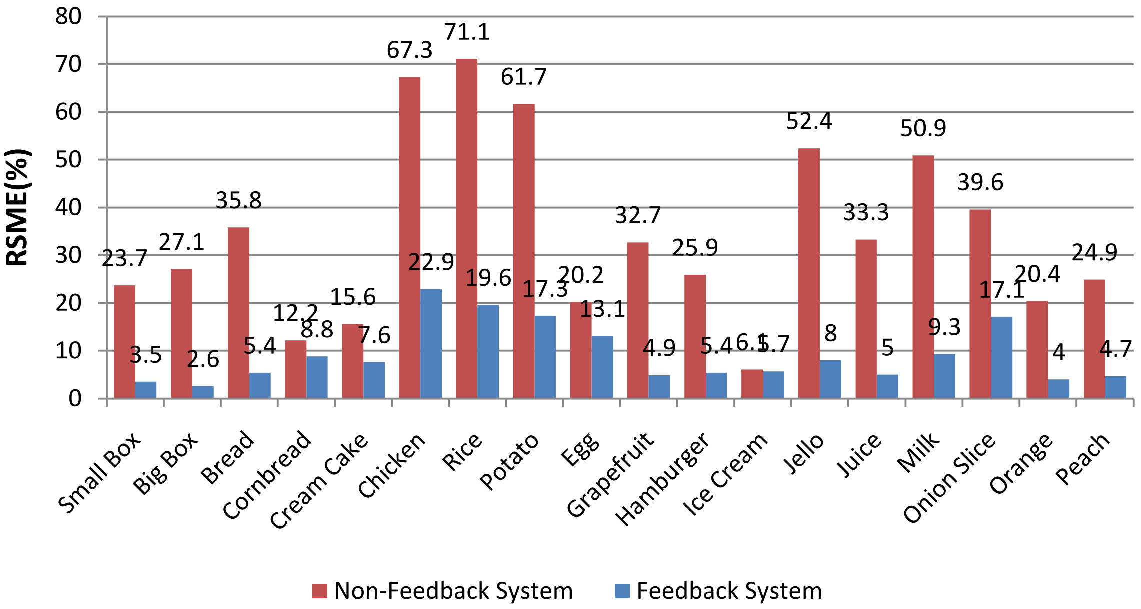


Figure 24. Comparison of volume estimates using non-feedback and feedback systems.

It can be seen that in general the estimation error rate with the non-feedback system is much larger than the rate with the feedback system. The averaged RMSE of volume estimation with the non-feedback system is 34.49%, but with the feedback system, it is reduced to only 9.16%, a very large improvement in volume estimation. Given that the system was so greatly improved by the feedback, the following experiments were all performed with feedback systems.

4.3 EXPERIMENT WITH DIFFERENT CAMERA POSITIONS

In the last section, we analyzed the data came from four camera positions to compare the results obtained using the feedback vs. non-feedback system, but we did not consider the effects of different camera positions on the results. In this section, we will use the same data obtained with the feedback system described in the last section, but will now analyze the effect of camera positions on the estimation results.

Figure 25 below shows the images of the glass of milk, the piece of cornbread and the peach from four different camera positions that were used for volume estimation. Figure 26 shows the relative error for each food item obtained with four different camera positions. Table 6 lists the means and STDs of the relative errors for the four camera positions. Figure 27 shows the RMSE and STD for each food item across the four groups of experiments.

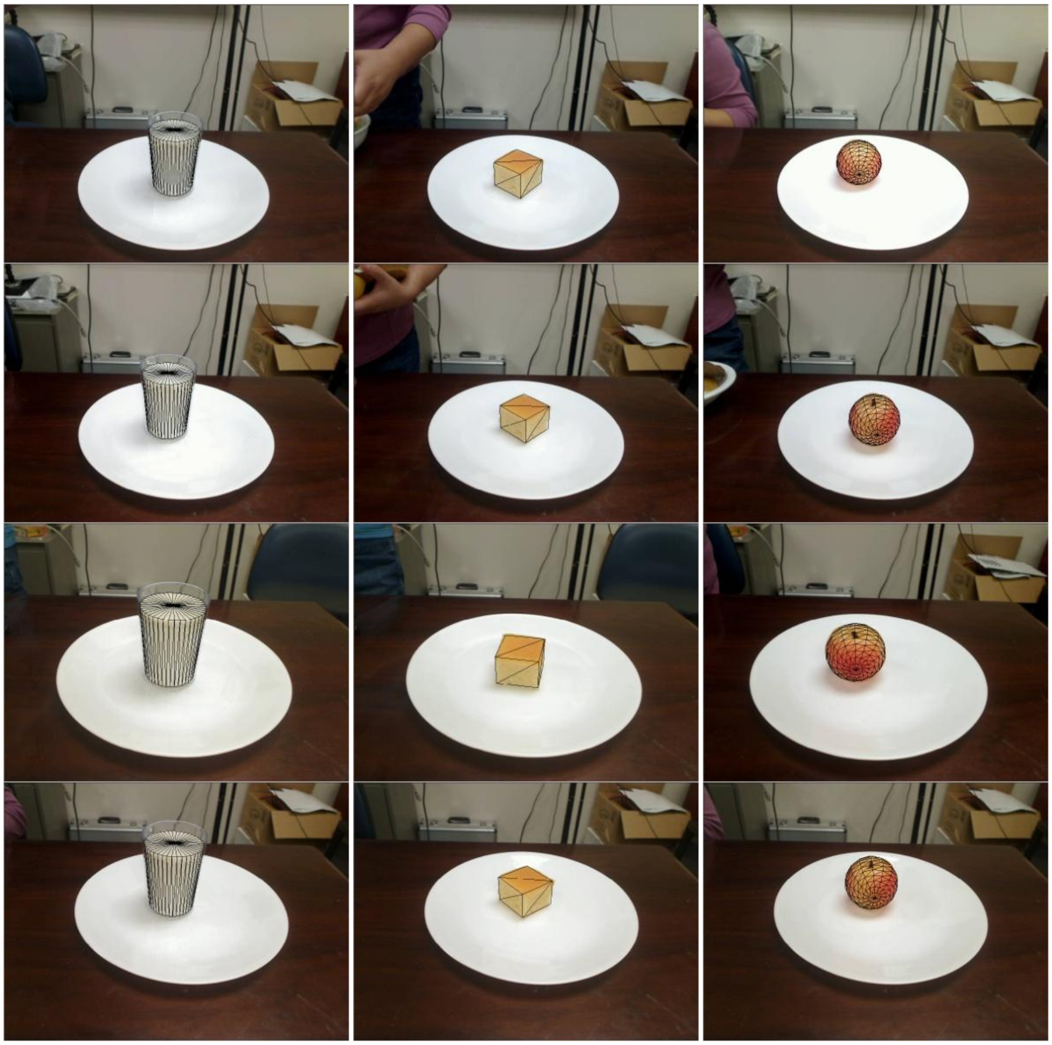


Figure 25. Result images of food items following volume estimation using four camera positions.

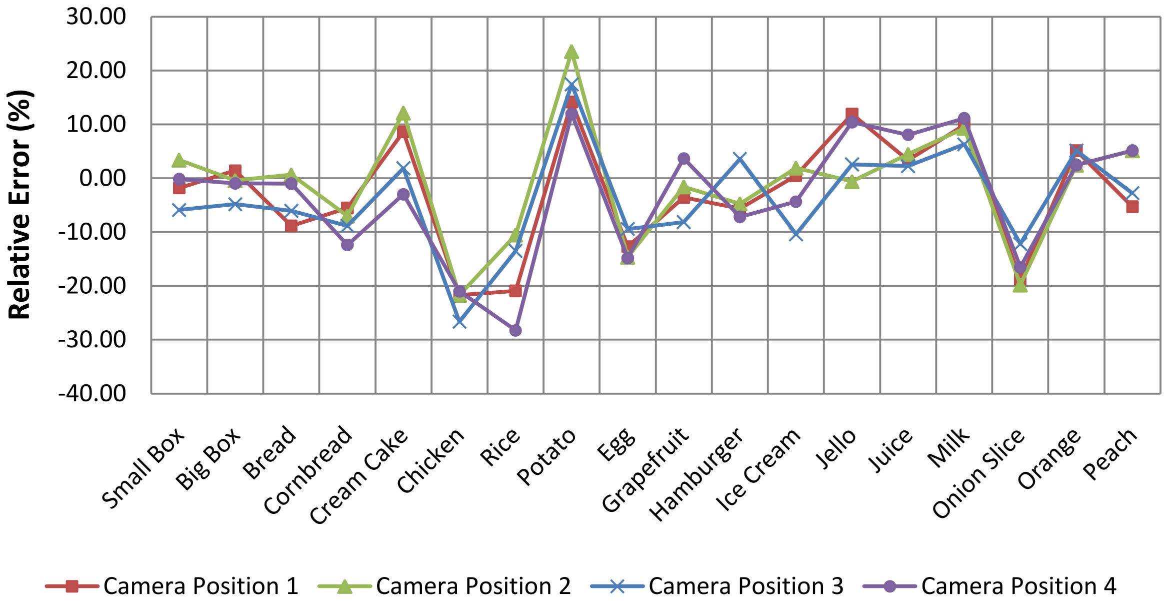


Figure 26. Percentage of relative error in volume estimates of each item for four camera positions.

Table 6. Mean and STD of relative errors for each item obtained with four camera positions.

	Mean of relative error (%)	STD of relative error (%)
Camera Position 1	-0. 19	8.54
Camera Position 2	0. 74	8.35
Camera Position 3	-2.5	8.6
Camera Position 4	-0.22	6.21

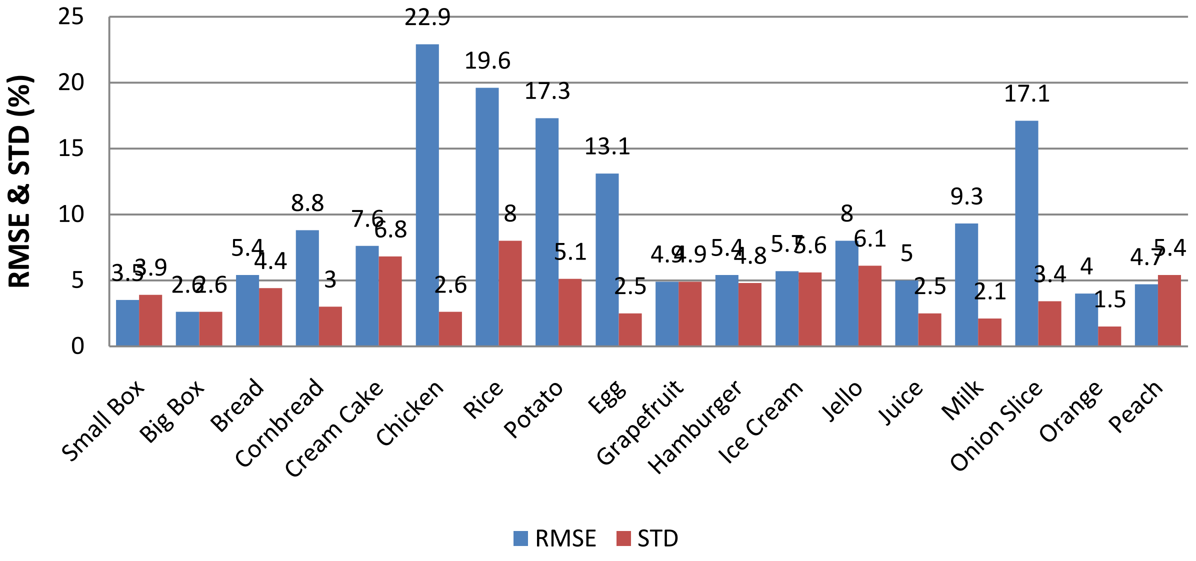


Figure 27. RMSE and STD of relative errors in volume estimates for each item obtained from four different camera positions.

Figure 26 shows no strong trend for the four lines, indicating that the relative percentage of errors in volume estimation from the four camera positions was very close. Table 6 shows the means and STDS of relative errors for all objects across four camera positions. It can be observed that the means and STDs of the relative errors do not show explicit trends for the four camera positions. Combining these two sets of results, we are able to conclude that our system for food-volume estimation exhibits no particular sensitivity to varying camera positions.

Figure 27 shows that the STDs for all of the samples were less than 10% and that most of the RMSEs were below 8% except for the chicken breast, the rice, the potato, the egg and the onion slice. The strong results for STD show the high degree of stability in our system. For the items with an RMSE less than 10%, the food-volume estimation results shows high accuracies, and the five biggest errors all occurred with objects that were irregularly shaped or small. During the estimation process for irregularly shaped objects, users used their knowledge to approximately fit regularly shaped virtual wireframe into the irregularly shaped object presented in images, instead of exactly overlapping projections of objects and models, which produced a higher error rate. The egg and the onion slice were relatively small objects, so the volume estimation for each of these objects were very sensitive to the appearance of absolute errors in the experiment, which introduced larger relative errors into the results. In summary, the estimation results obtained were sensitive to irregularly shaped objects as well as small objects.

4.4 EXPERIMENT WITH DIFFERENT OBJECT LOCATIONS

We performed another experiment with a fixed camera position to analyze how locating the objects in different locations affected the accuracy of estimates. Eleven food items were used in this experiment, and for each item, we estimated its volume at sixteen different locations. As shown in Figure 28, we used a circular reference feature to equally divide a table surface in front of the camera into sixteen quadrants $Q_{i,j}$, where i and j indicate the integers in the range [1,4]. We calculated the volumes of each sample for each of the resulting sixteen quadrants and then analyzed the results. The camera was set at a position with the angle α around $45°$, and the coordinates of the camera optical center in the experiment were $\boldsymbol{X} = [40{,}303.9357{,}239.0729]$, obtained using the checkerboard method.

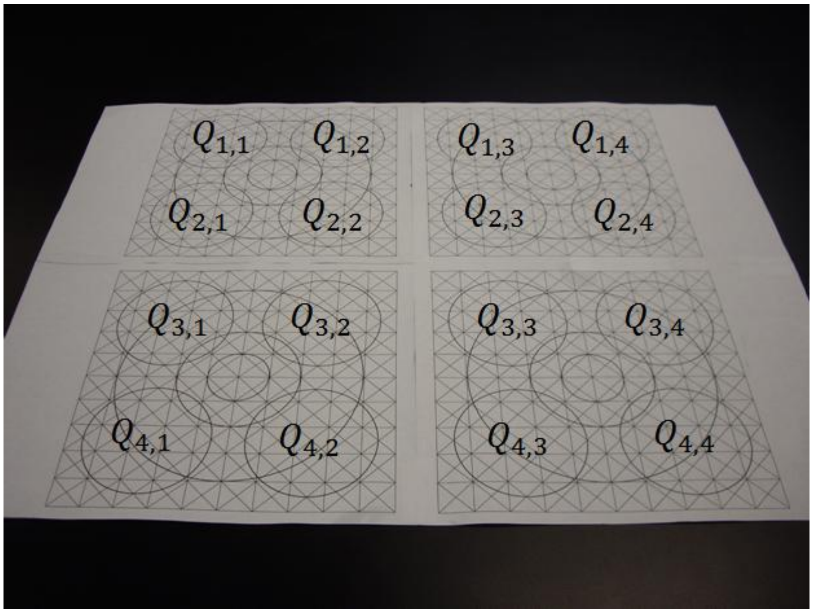


Figure 28. Sixteen locations of items defined by circular reference features.

Tables 7-17 show the estimation results for each item at sixteen locations. Table 18 shows the RMSEs of the volume estimates for items at sixteen locations. Table 19 shows the mean values of the relative percentage of errors for volume estimates at each of sixteen locations. Figure 29 shows the RMSE and STD for each food item at sixteen locations.

Table 7. Volume estimates for LEGOs at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j=1$	$j=2$	$j=3$	$j=4$	$j=1$	$j=2$	$j=3$	$j=4$
$i=1$	72.07	78.47	80.71	86.87	-7.96	0.22	3.08	10.94
$i=2$	72.52	79.27	83.44	84.80	-7.38	1.24	6.56	8.30
$i=3$	72.62	76.49	81.87	86.98	-7.25	-2.32	4.56	11.08
$i=4$	74.53	78.67	82.77	84.62	-4.81	0.47	5.71	8.07

Table 8. Volume estimates for small box at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	51.82	57.93	56.78	59.76	-11.43	-0.97	-2.94	2.16
$i = 2$	54.14	56.05	56.67	59.05	-7.46	-4.18	-3.13	0.94
$i = 3$	51.03	54.67	57.48	59.70	-12.78	-6.55	-1.75	2.05
$i = 4$	54.20	55.38	59.31	58.17	-7.35	-5.33	1.39	-0.57

Table 9. Volume estimates for onion slice at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	21.67	25.46	26.88	23.52	-33.58	-21.97	-17.63	-27.93
$i = 2$	24.43	25.81	26.52	25.81	-25.14	-20.89	-18.73	-20.89
$i = 3$	24.77	25.11	32.71	27.24	-24.09	-23.03	0.23	-16.53
$i = 4$	24.43	28.25	32.26	29.83	-25.14	-13.41	-1.12	-8.57

Table 10. Volume estimates for cornbread at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	78.74	87.69	87.07	82.18	-15.33	-5.71	-6.38	-11.64
$i = 2$	80.22	83.43	86.80	83.50	-13.74	-10.29	-6.67	-10.21
$i = 3$	77.33	86.23	80.75	87.76	-16.85	-7.28	-13.17	-5.63
$i = 4$	94.22	81.54	90.35	88.53	1.31	-12.32	-2.85	-4.80

Table 11. Volume estimates for orange at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	147.47	143.63	163.55	172.01	-2.77	-5.30	7.83	13.41
$i = 2$	143.63	147.47	163.55	176.34	-5.30	-2.77	7.83	16.27
$i = 3$	155.37	151.39	163.55	172.01	2.44	-0.19	7.83	13.41
$i = 4$	151.39	151.39	167.74	167.74	-0.19	-0.19	10.60	10.60

Table 12. Volume estimates for peach at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	155.37	155.37	151.39	163.55	2.44	2.44	-0.19	7.83
$i = 2$	147.47	151.39	155.37	163.55	-2.77	-0.19	2.44	7.83
$i = 3$	159.43	155.37	163.55	172.01	5.11	2.44	7.83	13.41
$i = 4$	143.63	159.43	151.39	167.74	-5.30	5.11	-0.19	10.60

Table 13. Volume estimates for big box at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	233.03	236.01	249.80	256.08	-6.25	-5.05	0.50	3.02
$i = 2$	230.37	237.14	247.01	256.95	-7.32	-4.60	-0.63	3.37
$i = 3$	230.97	244.14	251.06	260.49	-7.08	-1.78	1.00	4.79
$i = 4$	237.00	242.24	253.42	259.80	-4.66	-2.55	1.95	4.52

Table 14. Volume estimates for hamburger at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	268.18	284.84	298.23	310.70	-12.83	-7.42	-3.07	0.98
$i = 2$	287.01	288.92	302.36	315.79	-6.71	-6.09	-1.73	2.64
$i = 3$	284.54	292.65	323.39	325.87	-7.52	-4.88	5.11	5.92
$i = 4$	274.06	272.46	318.07	335.95	-10.92	-11.44	3.38	9.19

Table 15. Volume estimates for golf ball at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	38.72	41.99	40.34	38.72	-4.81	3.23	-0.84	-4.81
$i = 2$	38.72	37.15	40.34	47.24	-4.81	-8.67	-0.84	16.12
$i = 3$	38.72	40.34	41.99	43.70	-4.81	-0.84	3.23	7.41
$i = 4$	38.72	38.72	41.99	45.44	-4.81	-4.81	3.23	11.71

Table 16. Volume estimates for juice at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	166.49	170.69	184.02	194.66	-7.51	-5.17	2.23	8.14
$i = 2$	171.73	191.54	197.95	197.95	-4.60	6.41	9.97	9.97
$i = 3$	172.09	189.06	182.93	197.95	-4.40	5.03	1.63	9.97
$i = 4$	180.00	198.86	193.63	196.89	0.00	10.48	7.57	9.38

Table 17. Volume estimates for toy ball at sixteen locations.

	Estimated Value (cm^3)				Relative Error (%)			
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 1$	$j = 2$	$j = 3$	$j = 4$
$i = 1$	118.56	121.94	128.91	132.49	-0.05	2.81	8.68	11.70
$i = 2$	118.56	121.94	128.91	132.49	-0.05	2.81	8.68	11.70
$i = 3$	136.14	121.94	128.91	121.94	14.78	2.81	8.68	2.81
$i = 4$	121.94	125.39	128.91	136.14	2.81	5.72	8.68	14.78

Table 18. RMSEs of the volume estimates for items at sixteen locations.

	RMSE in the Volume Estimation (%)				
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	Average
$i = 1$	12.98	7.84	6.89	11.75	9.86
$i = 2$	10.04	8.27	7.94	11.49	9.44
$i = 3$	11.60	7.97	6.29	9.59	8.86
$i = 4$	9.10	7.92	5.35	9.22	7.90
Average	10.93	8.00	6.62	10.51	

Table 19. Means of relative errors of volume estimates for items at sixteen locations.

	Mean of Relative Error of Volume (%)				
$Q_{i,j}$	$j = 1$	$j = 2$	$j = 3$	$j = 4$	Average
$i = 1$	-9.10	-3.90	-0.79	1.26	-3.13
$i = 2$	-7.75	-4.29	0.34	4.19	-1.88
$i = 3$	-5.68	-3.33	2.29	4.43	-0.57
$i = 4$	-5.37	-2.57	3.49	5.90	0.36
Average	-6.97	-3.52	1.33	3.94	

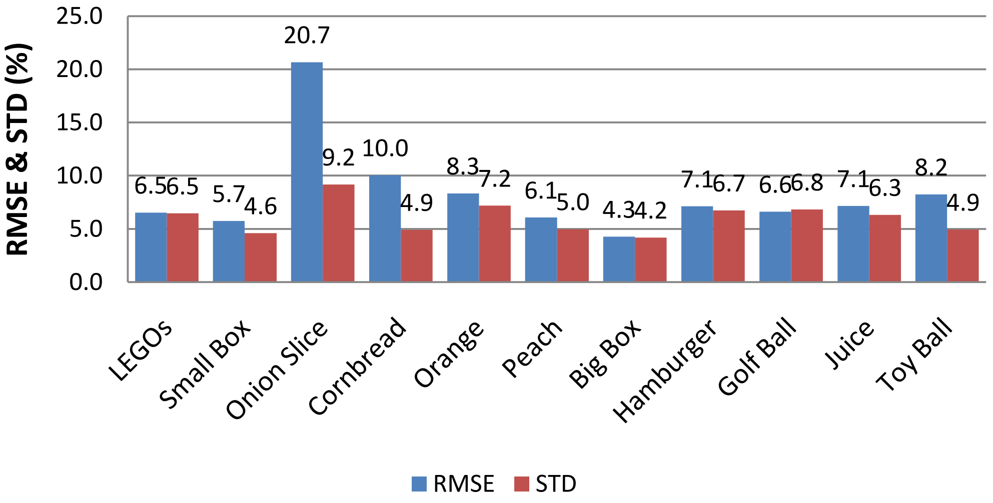


Figure 29. RMSEs and STDs of relative errors of volume estimates for each item at sixteen locations.

Table 18 shows all the RMSEs and means of relative errors for all objects with sixteen locations, where i and j indicates, respectively, the regions $Q_{i,j}$ in Figure 28. We found that when $j = 1$ and $j = 4$, the average value of the RMSEs for the four corresponding vertical quadrants were 10.93% and 10.51%, notably larger than the average RMSE values of 8.00% and 6.62% obtained when $j = 2$ and $j = 3$, respectively. We also found that in Table 19, the average value of the mean for relative errors in estimation volumes changed from -6.97% to 3.94%, monotonously with j changing from 1 to 4. The reason for this result is that the camera rotated away from its default position. When the horizontal center line of the camera's image plane is parallel with the experiment table surface, the camera is at the default position. In experiments, there must have been a small rotation angle of the camera around its optical axis, which means that first, the camera rotated away from its default position, and then, this rotation angle produced different sized images of a single object at two bilateral symmetry quadrants relative to the camera. As shown in the rows of Table 19, the errors change from a negative maximum to a positive maximum monotonously when the samples moved from one side of the image to the other along a horizontal line. The objects imaged near the vertical center line of the screen were characterized by more accurate estimates of their volume.

In Table 18, we also found that the average values of each row varied monotonously. The results were more accurate when the objects were positioned close to the camera. This is because a closer object provided a larger image with relatively clear borders for the fitting operation, which reduced estimation errors. However, there was no significant trend in the average values, and the difference between the maximum value and minimum value was only 1.96%. Thus, we can conclude that, the distance between the camera and the object will affect the accuracy of volume estimates to a minor degree.

Figure 29 shows that most of the RMSEs of volume estimates for each item at sixteen locations were equal to or less than 10%, and that all the STDs were less than 10%, showing that our system is also highly accurate when testing volume of foods at various locations. The larger error came from the estimation of the onion slice, which occurred because its tiny height made it sensitive to absolute error, as described in the previous section.

5.0 DISCUSSION

Our experiments and analysis have shown that the proposed approach and system enable highly accurate food-volume estimation based on the VR technology. The average relative error in volume estimation is less than 7% for regularly shaped food and less than 19% for irregularly shaped food, suggesting that our system can be used appropriately as a powerful tool for tracking caloric intake in research on, and treatment of, obesity. Also, since in our experiments the estimators gave the same readings regardless of the camera position or the distance between the camera and the food items, this shows that the system is robust and suitable for a wide variety of camera positions and foods. Our results also showed that our VR-based approach is sensitive to the rotation angle of camera around its optical axis. One problem we encountered was that, during various experiments, we had intended to set the horizontal center line of the camera's image plane parallel with the experiment surface, which can only represent the ideal situation of picture taking. We believe this problem can be solved by using the checkerboard method to calibrate the camera's rotation angle and then adding this variable into the system's pre-estimation process.

In out experiments, we found the major errors generally occurred with the estimation of irregularly shaped food. This is because the volumes of certain foods may not be defined correctly because of the non-uniqueness in treating the air spaces near the exterior, e.g., a bowl of leafy salad or a hamburger. Currently users of our system can only roughly define the surface as a boundary that smoothes out peaks and valleys of the visible exterior of food. This process introduced some error and uncertainty, but the degree of each was acceptable for irregular foods.

Although the experimental results of volume estimation using our VR approach and system have shown an acceptable degree of accuracy, future research should focus on the following four ways in order to improve the performance and the practicality of our system:

1) Introduce a camera calibration method with higher performance on the calibration of camera's intrinsic and extrinsic parameters, including the rotation angle;
2) Design a multi-dimensional feedback system to replace our current closed-loop system, which uses only one variable as a feedback error. The difference between the estimated reference and its true value provides knowledge of the type of multi-dimensional feedback errors which could be used to improve our system;
3) Provide virtual geometric objects with higher flexibility. As illustrated in our experiments, the volume estimation of irregular shaped always lacked accuracy because our system provided users with only regularly shaped objects. Providing virtual objects that could be deformed more flexibly would greatly enhance the performance of our system.

6.0 CONCLUSION

In research on the treatment of obesity, it is essential to accurately assess the volume of food consumed. However, thus far, the tools required for such close estimation have proved inadequate. This thesis presents a new approach to estimating the volume of food based on the use of a single input image, which is made possible through the use of virtual reality (VR) software/applications. The basic mechanism of the proposed system involves a two-step process: In the first step, a virtual pinhole camera model is created to simulate the picture-taking process by a real camera. In the second step, an imaginary 3D frame is created as a powerful measurement tool in the virtual space to cover or fit over the foods in the image in order to estimate the volume of the real food.

The construction of the virtual model was based on a well-defined mathematic model describing the relationship between the volume of objects in the virtual model and real objects. We designed and constructed software to realize the mathematical model, and the resulting system consisted of the following five functional units for: 1) food image acquisition, 2) camera parameters calibration, 3) virtual reality modeling and construction, 4) virtual object manipulation, and 5) food volume estimation.

Our system used the checkerboard method to determine the intrinsic and extrinsic parameters of the camera. Once these parameters were obtained, we established a VR space in which a virtual 3D wireframe was projected onto the food image in a well-defined proportional relationship. Within this space, the user could scale, deform, translate, and/or rotate the virtual wireframe to better fit the food in the image. Finally, the volume of the wireframe was utilized to compute the food volume based on the established proportional relationship. Finally, the volume of the real food was derived from the volume of the virtual wireframe.

Our experimental study indicated that out system very accurately estimated the volume of foods with relatively regular shapes (e.g., cornbread, cake, juice and hamburger), but less accurately estimated the volume of foods with irregularly shapes or very small sizes (e.g., chicken thigh and onion slice); however, even for the second group of foods, the volume estimation results were acceptable. We also found that in most cases our results were not sensitive to camera position or the distance from the objects to the camera's optical center; however, we did find that even a slight change in the camera-rotation angle about its optical axis could affect the results. In general, the results of our estimates were found to be satisfactory in that they provided an accurate measurement tool for dietary assessment in obesity research and treatment.

To enhance the performance of the current system future work in this area should focus on improving the accuracy in camera parameter estimation, testing the system using more food models and real foods, and designing a multidimensional feedback system to enhance system performance. It is hoped that, with the advancement in food portion size measurement using the VR technology, overweight and obese individuals can improve their awareness in energy intake, implement a more effective plan to lose weight, and live healthier lives.

BIBLIOGRAPHY

[1] http://www.healthyamericans.org, Trust for American Health, Robert Wood Johnson Foundation.

[2] U.S. Department of Health and Human Services. "Overweight and obesity: a major public health issue," *Prevention Report* 2001; 15-17.

[3] R.A. Hammond and R. Levine, "The economic impact of obesity in the United States," *Diabetes, Metabolic Syndrome and Obesity: Targets and Therapy* 2010; 3:285–295.

[4] C.K. Martin , S.D. Anton, E. York-Crowe, L.K. Heilbronn, C. VanSkiver, L.M. Redman, F.L. Greenway, E. Ravussin, D.A. Williamson and for the Pennington CALERIE Team, "Empirical evaluation of the ability to learn a calorie counting system and estimate portion size and food intake," *British Journal of Nutrition* 2007; 97:1-7.

[5] T. Trabulsi and D. A. Schoeller, "Evaluation of dietary assessment instruments against doubly labeled water, a biomarker of habitual energy intake," *Am. J. Physiol. Endocrimol. Metab*. 2001; 281(5):E891-E899.

[6] M.M. Most, AG Ershow, BA Clevidence, "An overview of methodologies, proficiencies, and training resources for controlled feeding studies," *J. Am. Diet Assoc*. 2003; 103:729-735.

[7] D.A. Schoeller, L. G. Bandini, W. H. Dietz, "Inaccuracies in self-reported intake identified by comparison with the doubly labeled water method," *Can J Physiol Pharmacol*. 1990; 68:940-950.

[8] M. Rashidi and M. Gholami, "Determination of kiwifruit volume using ellipsoid approximation and image-processing methods," *Int. J. Agri. Biol.*2008; 10:372-381.

[9] A.B. Koc, "Determination of watermelon volume using ellipsoid approximation and image processing," *Postharvest Biology and Technology,* Sep. 2007; 45:366-371.

[10] C. Yu and Q. Peng, "Robust recognition of checkerboard pattern for camera calibration," *Optical Engineering*. 2006; 45(9).

[11] H. Zhang, K. Y. K. Wong and G. Zhang, "Camera calibration from images of spheres," *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2007; 29(3):499-502.

[12] I. Woo, K. Otsmo, S. Kim, D. Ebert, E. Delp and C. Boushey, "Automatic portion estimation and visual refinement in mobile dietary assessment," *Proc. SPIE, Computational Imaging VIII,* Jan. 2010; 7533, 75330O.

[13] Food Surveys Research Group, "USDA Food and Nutrient Database for Dietary Studies," *3.0, Agricultural Research Service,* 2008.

[14] M. Puri, Zhiwei Zhu, Qian Yu, A. Divakaran, H. Sawhney, "Recognition and volume estimation of food intake using a mobile device," *Workshop on Applications of Computer Vision (WACV),* Dec. 2009; 3-8.

[15] W. Jia, R. Zhao, N. Yao, J. D. Fernstrom, M. H. Fernstrom, R. J. Sclabassi, and M. Sun, "A Food Portion Size Measurement System for Image-Based Dietary Assessment," *Proc. IEEE 35th Northeast Biomedical Engineering Conference*, Cambridge, MA, April 3-5, 2009.

[16] Y. Yue, W. Jia, J. D. Fernstrom, R. J. Sclabassi, M. H. Fernstrom, N. Yao, M. Sun, "Food Volume Estimation Using a Circular Reference in Image-Based Dietary Studies," *Proc 36th Northeast Biomedical Engineering Conference,* New York, NY, March 26-28, 2010.

[17] G.C. B**urdea,** P. Coiffet, ***Virtual Reality Technology***, 2nd Ed, John Wiley & Sons, New York, NY, 2003.

[18] A. Stork, M Maidhof, "Efficient and Pricise Solid Modelling Using a 3D Input Device," *Proceeding of Fourth Symposium on Solid Modelling and Applications,* Atlanta, Georgia, 1997; 180-195.

[19] Y. Zhong, H. Yang. W. Ma, "A Constraint-based Approach for Intuitive and Pricise Solid Modelling in a Virtual Reality Environment," *Proceedings of the Sixth International Conference on Computer Aided Design & Computer Graphics*, Shanghai, China, 1999; 1164-1171.

[20] S. Gao, H. Wan, Q. Peng, "An approach to Solid Modeling in a Semi-Immersive Virtual Environment," *Computer & Graphics* 2000; 24:191-202.

[21] R. Hawkes, "A Software Architecture for Modeling and Distributing virtual environments," Ph.D. Thesis, University of Edinburgh, UK, 1996.

[22] K.P. Berier, "Web-Based Virtual Reality in Design and Manufacturing Applications," *Proceedings of First International Conference on Computer Application and IT in the Maritime Industies,* Germany, 2000; 45-55.

[23] M.S. Hassan, R.G. Askin, A.J. Vakharia, "Cell Formation in Group Technology: Review Evaluation and Directions for Future Research," *Computers and Industrial Engineering*. 1998; 34:1-20.

[24] G. Chryssolouris, D. Mavrikios, D. Fragos, V. Karabatsou, "A Virtual Reality based Experimentation Environment for the Verification of Human Related Factors in

Assembly Processes," *Robotics & Computer Integrated Manufacturing*. 2000; 16:266-276.

[25] R. Grandl, "Virtual Process Week in the Experimental Vehicle Built at BMW AG," *Robotics & Computer Integrated Manufacturing*. 2001; 17:65-71.

[26] Z.D. Zhou, J.D. Zhou, Y.P Chen, S.K. Ong, A.Y.C. Nee, "Geometric Simulation of NC Machining Based on STL Models," *Annals of CIRP*. 2003; 52(1):129-134.

[27] B. Reitinger, D. Schmalstieg, A. Bornik, R. Beichel, "Spatial Analysis Tools for Virtual Reality-based Surgical Planning," *Proc. of the IEEE Symposium on 3D User Interfaces*, 2006.

[28] H. R. Malone, O.N. Syed, M.S. Downes, A. L. D'Ambrosio, D. O. Quest, M. G. Kaiser, "Simulation in neurosurgery: a review of computer-based simulation environments and their surgical applications," *Neurosurgery*. 2010; 67(4):1100-14.

[29] P. D. Van Hove, G.J. Tuijthof, E.G., Verdaasdonk, L.P. Stassen, J. Dankelman, "Objective assessment of technical surgical skills," *Br J Surg*. 2010; 97(7):970-986.

[30] G. Riva, S. Raspelli, D. Algeri, F. Pallavicini, A. Gorini, B.K. Wiederhold, A. Gaggioli, "Bridging Virtual and Real Worlds in the Treatment of Posttraumatic Stress Disorders," *Cyberpsychology, Behavior, and Social Networking*. Feb 2010; 13(1):55-65.

[31] O. Gervasi, R. Magni, M. Zampolini, "Virtual Reality in Neuro Tele-rehabilitation of Patients with Traumatic Brain Injury and Stroke," *Virtual Reality*. Jun 2010; 14(2):131-141.

[32] A. Londero, I. Viaud-Delmon, A. Baskind, O. Delerue, S. Bertet, P. Bonfils, O. Warusfel, "Auditory and visual 3D virtual reality therapy for chronic subjective tinnitus: theoretical framework," *Virtual Reality*. Jun 2010; 14(2):144-152.

[33] R. Korpela, "Disability and Rehabilitation," *Virtual reality: Opening the way*. 1998; 20:105-107.

[34] H.G. Hoffman, J.N. Doctor, D.R. Patterson, G.J. Carrougher, T.A.I. Furness, "Use of virtual reality for adjunctive treatment of adolescent burn pain during wound care: A case report," *Pain*, 2000; 85:304-309.

[35] S.K. Ong, A.Y.C. Nee, *Virtual and Augmented Reality Applications in Manufacturing*, Springer-Verlag London, 2003.

[36] R. Fisher, "Head-Mounted Projection Display System Featuring Beam Splitter and Method Of Making Same," *US Patent 5572229*. November 5, 1996.

[37] J. Parsons, J.P. Rolland, "A Non-Intrusive Display Technique for Providing Real-time Data Within a Surgeons Critical Area of Interest," *Proceeding of Medicine Meets Virtual Reality*. IOS Press, San Diego, CA, 1998; 235-256.

[38] H. Hua, A. Girardot, C. Gao, L.D. Brown, N. Ahuja, J.P. Rolland, "Engineering of Head-mounted Projective Displays," *Applied Optics* 2002; 39:3815-3824.

[39] T.P. Caudell, D.W. Mizell, "Augmented Reality: An Application of Heads-Up Display Technology to Manul Manufacturing Processed," *International Conference on System Sciences*, Kauai, HI, 1992; 650-670.

[40] G. Klinker, D. Stricker, D.Reiners, "Augmented Reality for Exterior Construction Applications," *Fundamentals of Wearable Computers and Augmented Reality.* Lawrence Erlbaum Associates Publishers, Mahwah, NJ, 2001.

[41] R. Sharama, R.J. Beveridge, "Computer vision-based Augmented Reality for Guiding Manual Assembly," *Presence: Teleoperators and Virtual Environments* 1997; 6(3):291-317.

[42] M. Bajura, H. Fuchs, R. Ohbuchi, "Merging Virtual Reality with the Real World: Seeing Ultrasound Imagery within the Patient," *IEEE Computer Graphics* 1992; 26(2):2003-2012.

[43] http://www.wikipedia.org, Pinhole Camera Model.

[44] Y. Ma, S. Soatto, J. Kosecka, S.S. Sastry, *An Invitation to 3-D Vision From Images to Geometric Models*, Springer-Verlag New York, 2004.

[45] F.D. Luna, R. Lopez, *Introduction to 3D Game Programming with DirectX 9.0*, Wordware Publishing, 2003.

[46] Jean-Yves Bouguet, Camera Calibration Toolbox for Matlab.

[47] P. Banerjee, D. Zetu, *Virtual Manufacturing*, John Wiley & Sons, New York, USA, 2001; 1-12.