Heliyon 9 (2023) e14722

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Learning metric volume estimation of fruits and vegetables from short monocular video sequences

Jan Steinbrener [a,*], Vesna Dimitrievska [b], Federico Pittino [b], Frans Starmans [c], Roland Waldner [c], Jürgen Holzbauer [c], Thomas Arnold [b]

[a] Control of Networked Systems Group, University of Klagenfurt, Universitaetsstr. 65- 67, Klagenfurt, 9020, Carinthia, Austria
[b] Silicon Austria Labs GmbH, Europastraße 12, Villach, 9524, Carinthia, Austria
[c] Philips Domestic Appliances Austria GmbH, Koningsbergerstr. 11, Klagenfurt, 9020, Carinthia, Austria

ARTICLE INFO

Keywords:
Volume estimation
Image recognition
Deep learning
Sensor
Fusion
Food datasets

ABSTRACT

We present a novel approach for extracting metric volume information of fruits and vegetables from short monocular video sequences and associated inertial data recorded with a hand-held smartphone. Estimated segmentation masks from a pre-trained object detector are fused with the predicted change in relative pose obtained from the inertial data to predict the class and volume of the objects of interest. Our approach works with simple RGB video frames and inertial data which are readily available from modern smartphones. It does not require reference objects of known size in the video frames. Using a balanced validation dataset, we achieve a classification accuracy of 95% and a mean absolute percentage error for the volume prediction of 16% on untrained objects, which is comparable to state-of-the-art results requiring more elaborated data recording setups. A very accurate estimation of the model uncertainty is achieved through ensembling and the use of Gaussian negative log-likelihood loss. The dataset used in our experiments including ground-truth volume information is available at https://sst.aau.at/cns/datasets.

1. Introduction

Documenting daily food consumption plays an important role in personal health [1,2] and dietary research [3,4] both to track caloric intake and uptake of macro- and micro-nutrients. Traditional methods relying on self-report and self-assessment of food intake have been found to be inaccurate with large variations across population groups [5] thus motivating technical solutions for accurate quantification of food intake. Images have been shown to be an effective means to extract quantitative information about food object properties [6,7], and, while other solutions have been proposed (including measuring food items with a ruler or weighing them with a scale), most methods rely on images for extracting volume or weight information of food objects. Earlier approaches required sending images of food items to dietitians for manual analysis of dietary content which proved difficult in terms of accuracy of the assessment from images alone and hampered wide dissemination of the technique due to the manual, labor-intensive nature of the approach. With the advent of modern smartphones and IoT devices, fully automated approaches have been developed to facilitate dietary assessment. The approaches differ in terms of the methodology used for food object detection and volume estimation. While the former can be reliably solved with modern object detection algorithms based on neural networks, the latter is more challenging and most approaches either require a reference object of known size to be present in the images or rely on elaborated data recording setups such as multiple

* Corresponding author.
E-mail address: jan.steinbrener@aau.at (J. Steinbrener).

https://doi.org/10.1016/j.heliyon.2023.e14722
Received 6 July 2022; Received in revised form 14 March 2023; Accepted 15 March 2023
Available online 21 March 2023
2405-8440/© 2023 Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

calibrated cameras, stereo rigs or depth cameras to achieve acceptable performance. This limits their application to lab-based setups as most suitable consumer devices such as smartphones do not contain depth cameras or multiple cameras with the required baseline for accurate metric volume estimation.

We present here a different approach for volume estimation that only relies on data readily available from any modern smartphone consisting of short videos recorded while moving around the object of interest. Apart from video frames, inertial data encoding information about the motion of the phone is also recorded from its onboard inertial measurement unit (IMU). The video frames are processed by a neural network trained to estimate segmentation masks and class labels of objects of interest. The IMU sequences are processed with a recurrent neural network architecture that was trained in an unsupervised fashion to predict 6°-of-freedom transformation between subsequent frames. These predictions are then fused in another recurrent neural network architecture that predicts the volume and class of the objects of interest. Our approach does not require reference objects of known size to be present in the video frames and works with simple monocular RGB video frames while still achieving competitive accuracy in volume estimation compared to the state-of-the-art. We also do not rely on assumptions about typical shapes of object instances, e.g., in the form of a typical 3D model that can be fitted. Since no existing public data set for food volume estimation provides inertial data in combination with sequential video frames that are needed for our approach, we have recorded our own data set. To enable further algorithm development in this direction, we have made our data publicly available. In summary, our contributions are:

- Competitive volume estimation accuracy based on short video sequences recorded with a smartphone only relying on monocular camera data along with inertial data and not imposing additional constraints in terms of background and viewing angles.
- Novel scheme for unsupervised pre-training of Long Short-Term Memory network to predict 6 DoF relative pose change between subsequent video frames.
- Minimized data annotation efforts due to unsupervised training of IMU feature extractor and use of 2D image segmentation network to extract features from individual video frames.
- Design of modular architecture, fusing variable rate video data and variable length IMU data for volume estimation that can be extended to other objects of interest with minimal effort.
- Accurate estimation of the model's uncertainty for every video sequence, through ensembling and the use of Gaussian negative loglikelihood (GNLL) loss.
- Comprehensive dataset of videos along with inertial data recorded of 11 classes of fruits and vegetables consisting of 5 different individual objects each with ground-truth volume information and several recorded trajectories with variations in different parameters (speed, pitch angle, distance, complexity of background).

2. Related work

Classical methods typically rely on well-controlled data recording setups including uncluttered backgrounds for object segmentation and subsequent volume estimation. In Ref. [8], the food objects are segmented from homogeneous backgrounds and then compared to a projected light spot of known size by fitting a model based on projective geometry. This work has been extended to other types of reference objects [9,10]. Another approach consists in segmenting the object of interest from images and then matching the detected objects to 3D shape primitives. In Ref. [11], a template shape based on the predicted food class is matched to the segmented object and scaled to estimate volume based on detected 3D feature points from single images. This is extended to extract shapes from silhouettes in multi-view acquisitions by Ref. [12]. Similar approaches based on classical segmentation and subsequent template matching with or without reference objects for sizing have been proposed by various groups [13–18]. Other approaches attempt to directly reconstruct a 3D model based on two or multiple views to determine volume [19–21]. Experimental setups involving stereo-cameras [22] and structured light illumination [23] were also successfully used to determine 3D shape and volume. While achieving good performance on the types of objects and scenarios considered, these classical approaches suffer from poor generalizability to heterogeneous, cluttered backgrounds where typically the segmentation algorithms fail.

Recently, learning-based methods, deep learning in particular, have excelled at image recognition tasks, including segmentation [24]. A variety of approaches combine deep learning methods for segmentation with nonlearning-based methods to estimate the volume of the segmented objects (see Table 1). In Ref. [25], the authors estimate the volume of food items singularly placed on a dish. By recording two images, the top and side views, that include a reference object with known size, and segmenting the object with Faster-RCNN and GrabCut, they were able to estimate the volume of 20 different classes with a Mean Absolute Percentage Error

Table 1
Summary of features of the cited approaches for volume estimation where CNN are used for image segmentation.

Image proc. Model	Data	Food type	MAPE
Mask R–CNN [29]	Single image	16 classes	13%–108%
Modify. MobileNetV2 [31]	Single image + reference	50 food dishes	11% - 20%
Faster R–CNN [25]	Multiple images + reference	20 classes	0.7%–33.5%
Faster R–CNN [26]	Multiple images + reference	5 classes	3.1%–4.3%
FCN model [27]	Multiple images + sound echo	3 food dishes	0.3%–12.7%
Mask-RCNN [30]	RGB-D image	8 items	3.3%–9.4%
CNN-based model [28]	RGB-D image	3 food dishes	0.4%–14%
Inception-v2 [32]	RGB-D images, calories est.	10 food dishes	5%–35%

(MAPE) ranging up to 33.5% depending on the class. A very similar approach has been proposed by Ref. [26] where a convolutional neural network (CNN) is used in addition, to regress the refined bounding polygons to a volume.

More recently, in Ref. [27] the authors attempt to estimate the volume of complex food items in bowls and plates using a smartphone's camera and microphone, with the latter being used to obtain a depth map of the food.

In its container. To separate the food from its container, two images per item, a top and a side view, are recorded and segmented using various CNN architectures. The reported MAPE is in the worst case above 12% but the algorithm is tailored to a small set of specific food containers.

A very similar approach is presented in Ref. [28] where, instead, the depth is estimated by taking advantage of the stereo information from the pair of cameras in modern smartphones. A CNN is again used for image segmentation into 1000 food categories, however only 3 food items are studied in the paper. This makes the method very tailored to this particular scenario, and probably overfitted. The reported MAPE is however still above 10% in the worst case. Another approach [29], uses a CNN architecture to predict a dense depth map from 2D images and combines this information with a segmentation network to generate a 3D point cloud of the objects of interest from which volume can be estimated. Even with a homogeneous, empty background, the resulting MAPE is rather large, exceeding 80% for some classes. Yet another approach [30] relies on RGB-D image data: first, a segmentation mask is created for the object of interest using a Mask R–CNN network, then, the masked depth image is used to predict a second depth image from the opposing viewing angle with a dedicated neural network. Finally, the two depth images are synthesized into a 3D model using the iterative closest point algorithm. While the computed error for 8 items considered is 10% or less, it is unclear if this performance would extend to unseen items of the same class but with different volumes as the algorithm was only evaluated on the items that it was trained with. A more comprehensive approach is instead pursued in Ref. [32], with the primary goal of the authors being the creation of an opensource dataset of prepared food items together with detailed ground-truth data. The images were taken in a fixed structure with 4 RGB cameras on the sides of the objects and, for 3.5 k dishes, also the depth information was recorded with an overhead RGB-D camera. The authors have also developed an algorithm for nutrients prediction based on a CNN using as inputs either only the 4 cameras or also the depth information with a total MAPE of 13% (exceeding 30% in some cases) for calories estimation in the best case, i.e., including the depth information. According to the authors, the greatest part of this error is related to the implicit volume estimation.

Finally, a different approach is pursued in Ref. [31] by employing only a single 2D image per item and relying on the presence of objects of known size in the image, i.e., a plate of known radius. By comparing the features of such reference objects in a modified MobileNetV2 with the features from the object of interest, the volume can be estimated. The reported relative errors on the test dataset are on average 11%–20%. Another approach that directly predicts class and nutritional information from 2D images using a CNN architecture has been presented in Ref. [33]. Here, the authors focus on directly predicting caloric content based on 2D images alone. While the achieved results show good accuracy at least for the food objects contained in their dataset, the method requires a special fiducial marker for size and color calibration to be present in each image that is being analyzed.

Table 2 summarizes the key features and results of the above-mentioned state-of-the-art works that use deep learning methods, like Mask R–CNN [24], for image segmentation. Most of the above approaches have in common that they put restrictions on the experimental setup that are difficult to realize for real-world, end-user applications in that they require either a certain reference object to be always present in the images or that they require certain hardware (stereo, multiple cameras, depth cameras) that is not likely to be available. The only approaches considering only imaging data and easily available sensor data suffer from poor accuracy [29] or only work with a very restricted set of well-defined food containers [27].

3. Methods

An overview of the proposed modular pipeline for volume estimation is shown in Fig. 1. A sequence of N video frames is first passed through an instance of a Mask R–CNN [24] network trained to detect objects of interest on each frame and estimate their segmentation masks and class labels. Likewise, N sequences of IMU data are first passed through an LSTM-based network.

that has been trained in an unsupervised way to predict 6 DoF pose changes along with IMU noise parameters and the initial velocity. The predictions of both networks are then concatenated and passed through another LSTM network that is trained for volume

Table 2
Description of the distribution of the volumes of the food items recorded in the dataset per class. The values, expressed in dm^3, are truncated to 3 decimal points.

Class	Range	Mean	Std
apple	0.143–0.324	0.187	0.077
avocado	0.21–0.243	0.222	0.013
banana	0.185–0.24	0.21	0.027
blackberry	0.049–0.16	0.109	0.042
blueberry	0.185–0.295	0.253	0.042
carrot	0.029–0.082	0.049	0.020
cucumber	0.26–0.543	0.391	0.131
grape	0.034–0.293	0.21	0.107
peach	0.117–0.131	0.124	0.006
pear	0.145–0.191	0.157	0.019
strawberry	0.201–0.296	0.249	0.035

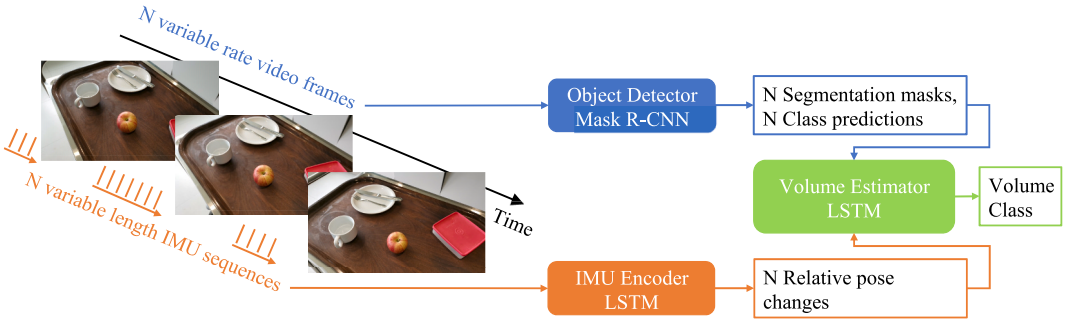

Fig. 1. Proposed volume estimation pipeline: The class and volume of the object of interest (here an apple) are predicted by fusing segmentation masks and class predictions obtained from each video frame with predicted relative pose changes between subsequent camera positions based on the collected inertial data.

estimation and classification. For better modularity of the resulting network architecture, we rely here on a late fusion approach where the predictions of networks are fused together rather than intermediate features. This also follows the intuition, that the predictions combined contain the information that is needed to perform volume estimation: the measured size of the object in the images given by its segmentation mask can be converted to a metric volume through the information on the camera trajectory obtained from the IMU encoder network and information about typical geometric properties (such as aspect ratio) inferred from the class label prediction.

3.1. Dataset

The data was collected with a mobile phone Huawei Mate 20 Pro using a custom application implemented in Android Studio to record videos and sequences of inertial data from its internal IMU. Each video consists of a sequence of video frames obtained with a variable sampling rate of around 2 Hz and IMU data recorded at a fixed sampling rate of 100 Hz. Overall, 1320 videos have been recorded, each containing the full 360-degree view of an object of interest, which is recorded by going around the object with the mobile phone pointing at the object. The objects of interest are items from one of the selected set of classes of fruits and vegetables: apple, avocado, banana, blackberry, blueberry, carrot, cucumber, grape, peach, pear, and strawberry. Five different food items, labeled with IDs from 1 to 5, are recorded for each class. Each food item is recorded 24 times by varying the setup along the following parameters: i) background (simple, medium, or complex), ii) motion speed (fast or slow), iii) pitch angle (low or high), and iv) distance (close or far). These setups represent different real-world scenarios to record an object. Examples of recorded image frames under various setups are given in Fig. 2. Note that each recording is made by free movement and the background varies between the recordings. The size of the recorded video frames is 1280×720 and each IMU reading consists of six values corresponding to linear accelerations with gravity filtered out (in m/s^2, TYPE LINEAR ACCELERATION from Android Studio API) and angular velocities (in rad/s) for 3 axes each.

From these raw videos, a dataset of shorter video segments was created for volume estimation. Each video segment consisted of $N = 9$ video frames and associated IMU sequences starting with the IMU sequence taken before the first video frame. Several filters were applied to obtain the video segments used for our investigations. The following statements are therefore true for the final dataset:

- All video segments have a video frame rate of at least 1 Hz at any given time.
- Video segments from the same raw video overlap by at most two frames.
- All images in the video segments are landscape.
- All values in the IMU sequences are numerical.

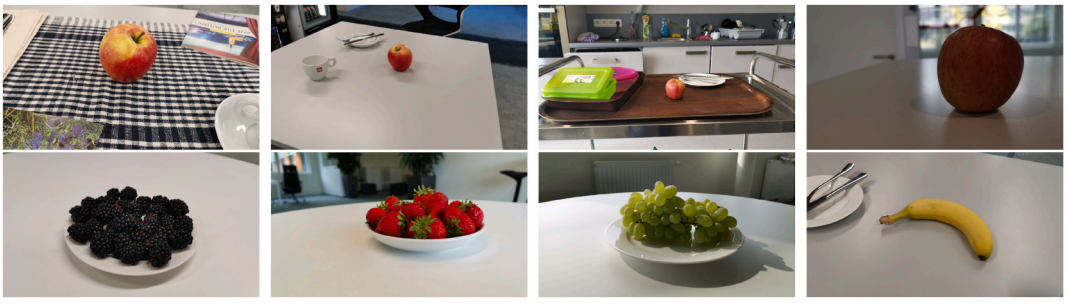

Fig. 2. Examples of images from the collected dataset. The top row shows apples in different scenes with varying background, the bottom row shows from left to right: a bowl of blackberries, a bowl of strawberries, grapes, and a banana.

This resulted in a total of 6909 video segments (examples of the image frames of one video segment are shown in Fig. 3). This dataset was then split into a training and a test set, used to create the model, and a validation set that was used to validate the model. Different data splits were exploited in the analysis, but one constraint is that the validation set has only instances of food items unseen in the training and test sets. In this way, the ability of the model to generalize to unseen objects can be investigated.

The dataset is annotated with the ground truth classes and volumes, measured either with a 3D scanner or with a traditional water displacement method. The volumes of the recorded food items vary from 0.034 dm^3 to 0.543 dm^3, and the volume range per class is given in Table 2. The volumes of food items from the classes avocado and peach have the narrowest range, while the items from the classes cucumber, grape, and apple have the greatest volume variety. Note that items from the classes apple, avocado, banana, carrot, cucumber, peach, and pear were considered individually, while for the classes blackberry, blueberry, grapes, and strawberry only clusters consisting of several items were considered.

The raw dataset with 24 different recordings of the 360-degree view of 55 different food items of 11 different classes of fruits and vegetables, together with their measured volume, is available for download at https://sst.aau.at/cns/datasets. For each recording, the sequences of video frames and corresponding IMU data will be given along with information about the characteristics of the trajectory performed (e.g., speed, pitch angle, distance). The combination of video frames with inertial data makes our data set a unique asset to develop and evaluate novel types of algorithms for food volume estimation using common handheld smart devices not available in other public data sets such as [32].

3.2. Network architectures and training details

3.2.1. Object detection network

Object detection is performed on each individual video frame using a pre-trained Mask R–CNN network with Resnet-18 backbone for feature extraction. To train this network, an MS Coco pre-trained version was finetuned using 851 hand-labeled images of the 11 different types of fruits and vegetables considered for volume estimation. The majority of these images were obtained from the web with varying backgrounds. About 330 images were taken from the dataset recorded for volume estimation described above. These images were then excluded from the data for later training of the volume estimation network. The network was finetuned with this data for 40 epochs using the stochastic gradient descent (SGD) optimizer with a learning rate of 0.005, momentum of 0.9, and a weight decay of 0.0005. The learning rate also decayed every 20 epochs with a gamma of 0.1. The batch size was 8. As a loss function, the multitask loss described in Ref. [24] was used. The output of the object detection network contains object bounding boxes, objectness scores, segmentation masks, and class label scores. Of these, only the segmentation masks and the class label scores were used in the volume estimation network as described below.

3.2.2. IMU encoder network

The IMU encoder network takes sequences of variable lengths of IMU samples and predicts the change in position from the time point immediately before the recorded IMU sequence to the time point at the end of the IMU sequence. As the IMU sequence always corresponds to the IMU samples recorded between two video frames, this basically predicts the change in camera pose between the two video frames but expressed as a change in the pose of the IMU in the world. The network (Fig. 4) consists of a bi-directional, bi-layer LSTM network with the size of the hidden layer of 128. The final forward and reverse hidden states for the two layers are concatenated and passed through a Rectified Linear Unit (ReLU) activation function. This is followed by one linear layer. As input, the network takes the sequences of IMU samples with each sample being comprised of 6 values corresponding to 3 axes each for measured linear accelerations (a_{mx}, a_{my}, a_{mz} in $\frac{m}{s}$) and angular velocities ($\omega_{mx}, \omega_{my}, \omega_{mz}$ in $\frac{rad}{s}$). Since the video frames were recorded at a variable rate, the length of IMU sequences between any two frames varies. For training, the network was always provided with 100 IMU samples corresponding to 1 s worth of data. This corresponds to a maximum length of 1 s worth of IMU samples and matches the cutoff value defined for the video frame rate for selecting segments from the recorded dataset (see above). Note that for inference, always the actual number of IMU samples recorded in between two video frames is passed to the IMU encoder network. The linear layers reduce the output of the LSTM network from 128 to 15. The 15 values that are returned predict i) the rotational change expressed as axis-angle $\boldsymbol{r}_\Delta$, ii) the relative translation $\boldsymbol{p}_\Delta$, iii) the bias of the IMU accelerometer $\boldsymbol{b}_a$ and gyro $\boldsymbol{b}_\omega$, and iv) the initial velocity $\boldsymbol{v}_0$ at the start of the IMU sequence. The network is trained in an unsupervised way by comparing the predicted change in the pose with the change in pose obtained from the classical integration of the state dynamics given as

$$\dot{\boldsymbol{p}}_\Delta = \boldsymbol{v}_\Delta$$

$$\dot{\boldsymbol{v}}_\Delta = R_{q_\Delta}(\boldsymbol{a}_m - \boldsymbol{b}_a)$$

Fig. 3. A sample video segment with 9 video frames obtained from the recording of a banana with a volume of 0.24 dm^3 (The shown recording was done with a simple background, slow recording motion, low pitch angle, and close distance.). The camera position and yaw angle with respect to the object changes from left to right.

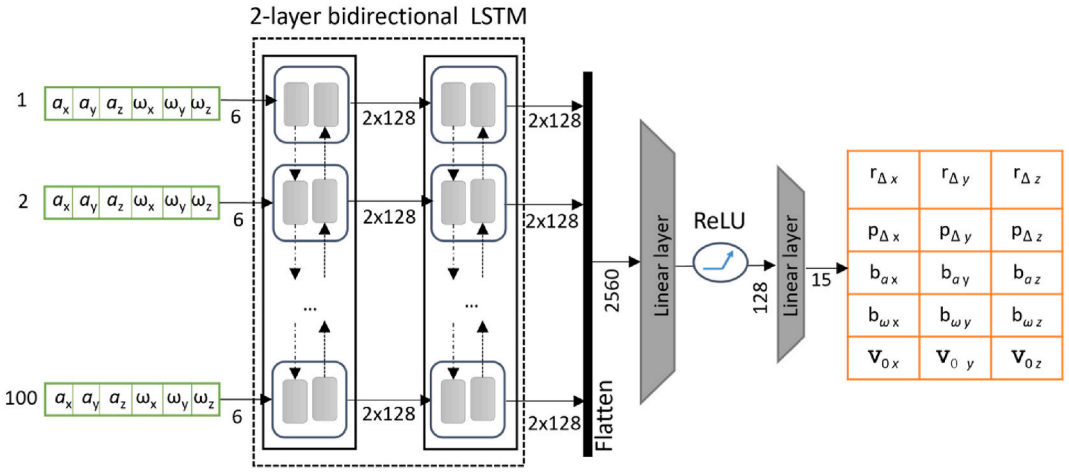

Fig. 4. IMU encoder network architecture with detailed information about the input and output data.

$$\dot{\boldsymbol{q}}_\Delta = \frac{1}{2}\Omega(\boldsymbol{\omega}_m - \boldsymbol{b}_\omega)\boldsymbol{q}_\Delta$$

where $\boldsymbol{p}_\Delta$ is the relative translation of the IMU between the start and the end of the sequence expressed in the world frame, $\boldsymbol{v}_\Delta$ is the corresponding velocity, $\boldsymbol{q}_\Delta$ is the change in orientation of the IMU in the world frame, $\boldsymbol{a}_m$ is the measured acceleration in the IMU frame, $\boldsymbol{b}_a$ and $\boldsymbol{b}_\omega$ are the accelerometer and gyro biases predicted by the network, $\boldsymbol{\omega}_m$ is the measured angular velocity in the IMU frame, $\Omega(\omega)$ is the quaternion multiplication matrix of ω, and R_q is the rotation matrix of the corresponding quaternion. As the gravity vector was already removed from our data (see above), the subtraction of the gravity vector in the equation for $\dot{\boldsymbol{v}}_\Delta$ can be omitted here. The loss function is then given as

$$loss = l_{pos} + l_{rot}$$

where l_{pos} is the mean-squared-error loss between the predicted relative translation and the translation calculated according to the state dynamics above and l_{rot} is the geodesic distance between the predicted rotation and the one calculated according to the state dynamics. The network was trained with 39,000 IMU sequences for 30 epochs using the Adam optimizer with a learning rate of 0.0015 and batch size of 32.

We would like to point out that the classical equations of motion cannot be used to calculate the change in pose directly, as important parameters such as the noise terms of the IMU are not known and are in fact predicted by the network. Instead, in our self-supervised training scheme, the classical equations of motion may be considered as a kind of regularization during training that ensures that the solutions predicted by the network conform to the underlying dynamics of the system.

3.2.3. *Volume estimation network*

The volume estimation network, shown in Fig. 5, is a LSTM-based model taking in a concatenated sequence of $N = 9$ predictions from the object detection network and the IMU encoder network. While the latter is taken as is, the output of the object detector for each video frame is consolidated to one class prediction vector given by the average scores obtained for each predicted class label and

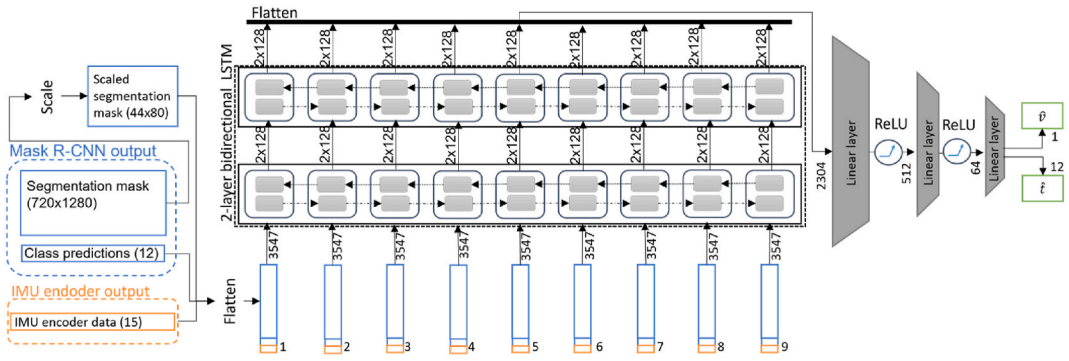

Fig. 5. Volume estimation network architecture with detailed information about the input data (extracted from the object detection network and IMU encoder output) and output data (class predictions ($\hat{t}$) and volume prediction ($\hat{v}$)).

one segmentation mask obtained through the scaled union of all predicted segmentation masks with objectness score higher than a selected threshold of 0.3. The masks are scaled down to the dimension 80 × 44. Therefore, the dimension of the input vector for the model is 15 + 12+80*44 = 3547. The output of the network is a 13-dimensional vector, where 12 values represent the predicted class label scores $\widehat{t}_i, i \in 0..11$, including a background class ($i = 0$), and one value represents the predicted volume on a log-scale ($\widehat{v}$), see Table 3.

The network itself is a bidirectional 2-layered LSTM with a hidden size of 128. Its output, which includes the forward and reverse hidden states at each of the nine time steps, is connected to a pipeline of 3 linear layers, that have inputs of sizes 2304, 512 and 64, accordingly. The first two layers have in addition a ReLU activation function, whereas the third layer gives the output of the model. The loss of the network is calculated as a sum of the classification loss, which is the cross-entropy loss l_{CE}, and the volume loss, which is the mean squared error l_{MSE} between the natural logarithm of the true volume V and the predicted log-value $\widehat{v}$. The formula is given in Eq. (1).

The loss of a batch is calculated as a mean value

$$loss = l_{CE}(\widehat{t}, C) + l_{MSE}(\log(\widehat{V}), \log(V)) \tag{1}$$

.

The training was performed using a data split where all data from one selected item of each class is used in the validation set. All other data were divided into training and test data so that 30% of the data from each class is used for test, while the rest is used for training. Preliminary investigations on different optimizers, parameters, and batch sizes resulted in choosing the Adamax optimizer with a learning rate of 0.001 and batch size of 6.

Experiments with ensemble models for volume estimation were also performed. Five models form an ensemble, and bootstrapping was used to form the training data for the multiple models. More precisely, the training data in each model was taken by sampling with replacement a certain amount of the training data (25%, 33% and 50%). The class prediction of the ensemble model is obtained using majority voting. In cases where no class has more than two of the votes, the model predicts an undefined label. The volume prediction of the ensemble is obtained by taking the average predicted volume of all models, while the variance of the ensemble is estimated at the variance of the predicted volume. Furthermore, when using ensemble models, the GNLL loss was also used instead of the mean squared error (MSE) loss. The training data in each model used 20%, 33%, and 50% of the initial training data with replacement. To be able to use the GNLL loss, the volume estimation model predicts an additional parameter $\widehat{s}$, that is related to the model's variance, alongside $\widehat{v}$, that is connected to the volume prediction. The loss of the i-th model in an ensemble was obtained using Eq. (2). The GNLL loss is calculated based on the estimated and true volume, and not based on their log value used in Eq. (1).

$$loss = l_{CE}(\widehat{t}, C) + l_{GNLL}(\widehat{V}_i, V, \widehat{\sigma}_i), \text{where } \widehat{\sigma}_i = \log(1 + e^{\widehat{s}_i}) \text{ and } \widehat{V}_i = \log(1 + e^{\widehat{v}_i}) \tag{2}$$

The class predictions for the ensemble models that use the GNLL loss were again obtained with majority voting. While the volume prediction $\widehat{V}$ is taken as the average of the estimated volumes $\widehat{V}_i$ calculated for each model i (Eq. (3)), the estimated variance $\widehat{\sigma}$ is calculated using Eq. (4).

$$\widehat{V} = \frac{1}{N}\sum_{i=1}^{N}\widehat{V}_i \tag{3}$$

$$\widehat{\sigma} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\widehat{V}_i^2 + \widehat{\sigma}_i^2\right) - \widehat{V}^2} \tag{4}$$

Table 3
Output specification of the volume estimation network.

Output index	Type	Details
0	Class	Background
1	Class	Apple
2	Class	Avocado
3	Class	Banana
4	Class	Blackberry
5	Class	Blueberry
6	Class	Carrot
7	Class	Cucumber
8	Class	Grapes
9	Class	Peach
10	Class	Pear
11	Class	Strawberry
12	Volume	log (V)

4. Results

4.1. Object detection network

An evaluation of the object detection networks was obtained using Voxel51 [34]. By considering detections with confidence above 0.5, the fine-tuned Mask R–CNN model achieved a weighted average F1-score of 0.84 on its validation dataset. Furthermore, the weighted average recall was 0.92 and the precision was 0.78. The performance metrics per class are given in Table 4. These results were obtained using a mask threshold of 0.2 and an Intersection over Union (IoU) threshold of 0.5. The ability of the model to correctly distinguish the instances within the detected masks is, however, best assessed using the mean average precision (mAP) metrics, since it is calculated using multiple IoU thresholds. For the Mask R–CNN model, the mAP is 0.64, and the average precision (AP for each class is shown in Fig. 6.

Two examples of the detected objects and extracted segmentation masks for the classes carrot and strawberries, taken from the dataset used for volume estimation training, are shown in Fig. 7.

4.2. IMU encoder network

The performance of the IMU encoder network was evaluated on the validation dataset which consisted of all trajectories recorded on the validation food items of all classes. During training, the change in the pose was predicted for each IMU sequence between two consecutive video frames. This time though, we predicted each trajectory separately and sequentially from start to end. This is achieved by setting an arbitrary starting position at (0,0,0) and sequentially applying the predicted transformations. A plot of all predicted trajectories in the validation dataset is shown on the right in Fig. 8. As can be seen, the predicted trajectories are semi-circular to ellipsoidal in shape. Absent any ground-truth information on the actual trajectories, we cannot evaluate the performance quantitatively. We do note, however, that the qualitative behaviour of the trajectories matches the experimental design as described in Sec. 3.1: All datasets were recorded along circular or ellipsoid trajectories at different radii but covering about 360° around the object. In addition, the diameters of the predicted trajectories range from about 0.8 to about 1.5 m which seems reasonable given how the experiment was carried out. As can be seen, all trajectories exhibit significant drift. This is somewhat expected, as we have trained against a dead-reckoning algorithm that is prone to drift. In the end, the purpose of this network is to provide information of the metric scaling factor that is not observable from images alone (because only the ratio of size over distance is observed in images). The results of the ablation study presented in Sec. 4.4 illustrate that this is indeed the case.

4.3. Volume estimation network

The performance of the volume estimation network has been investigated in two different scenarios, representing different choices of the validation set:

- a naive validation set, in which one fruit item per class is chosen for validation, specifically the one with id 2, without any special considerations;
- a balanced validation set, in which the fruit items in the validation set are chosen so that the training set covers in a balanced way the whole distribution of volumes per class. In this case, 2 to 3 items per class are chosen for validation, representing a more challenging validation set but with a more appropriate training set.

4.3.1. Naive validation set

The class accuracy, which is the percentage of instances that are correctly classified with respect to the total instances, is 99% for the training set. The results show a good generalization to the unseen data instances in the test set where an accuracy of 97% is obtained. The accuracy of the validation set, but also the average recall and precision, is 95% and that shows a good classification of the

Table 4
Precision, recall and F1-score of the fine-tuned Mask R–CNN per class.

Class	Precision	Recall	F1-score
Apple	0.81	0.92	0.86
Avocado	0.84	0.98	0.91
Banana	0.72	0.85	0.78
Blackberry	0.82	0.90	0.86
Blueberry	0.66	0.93	0.77
Carrot	0.75	0.92	0.83
Cucumber	0.72	0.86	0.79
Grapes	0.71	0.90	0.79
Peach	0.73	0.95	0.82
Pear	0.74	0.93	0.82
Strawberry	0.84	0.95	0.89

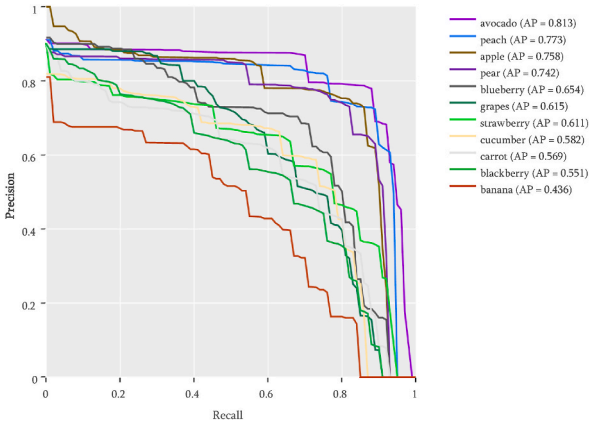

Fig. 6. Precision-recall (PR) curves of the fine-tuned Mask R–CNN.

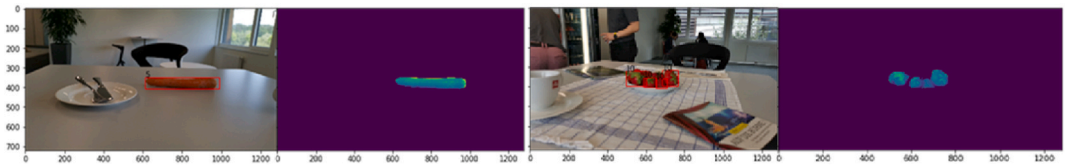

Fig. 7. Two examples of the Mask R–CNN detection and segmentation masks on the data used for volume estimation. Left: carrot with bounding box and segmentation mask, right: strawberries with bounding boxes and segmentation mask.

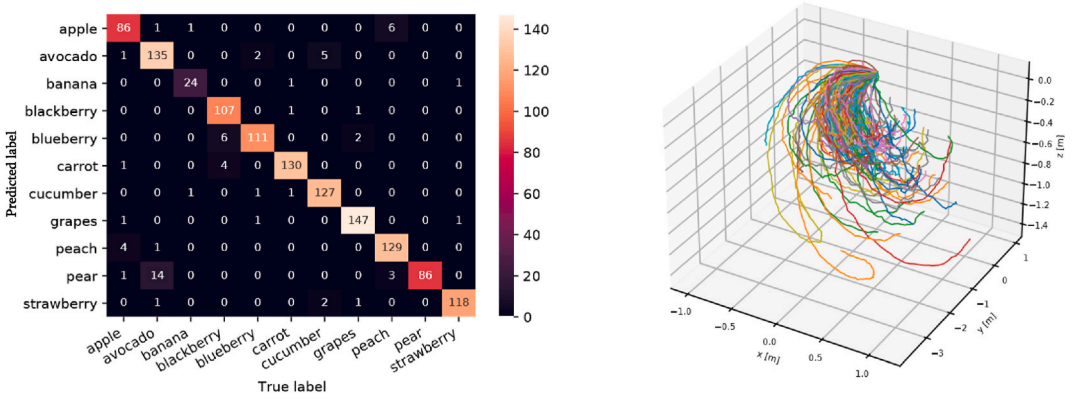

Fig. 8. Left: Validation data set confusion matrix. Right: Trajectory predictions of the IMU network on the validation data.

unseen food items. A slight confusion is mainly seen between visually similar classes such as pear and avocado, apple and peach, and blueberry and blackberry (Fig. 8).

The MAPE obtained for the three different datasets in the case of the naive validation set selection is given in Table 5. It is important to inspect if the estimated volume of the different food items per class correlates to their actual volumes. The volume estimation distribution per food item for all classes is shown in Fig. 9. The x marks on the graphs show the true.

volume of the items with ids 1 to 5. Note that the results from the validation set are those with id 2, while the other results belong to the training and test sets. The volumes of the items in the classes avocado, carrot, and peach have a tight range, and therefore unsurprisingly the MAPE on the volume is low. On the other hand, a greater variety of volumes is seen for the classes banana, blackberry,

Table 5
MAPE on the volume for each data set using the two different choices for the validation set.

	Training	Test	Validation
Naive val.	5.0%	16.7%	20.7%
Balanced val.	3.8%	16.4%	16.0%

and blueberry. Here one can observe that the estimations made for different items are in correlation with the true volumes of the items. For classes apple and grapes, instead, there is one item with a volume quite different than the other items. For this reason, the estimation on the test sequences of the 5-th item for the class apple underestimates the true volume, and also there is a wide range of estimated values for its volume (cf. Empty box in Fig. 9). On the other hand, the 1st item in class grapes, which has a much smaller volume than the others, is predicted well. The items in the class cucumber have the highest range and variety, and it can be observed that the range of the estimations per item is wider. Still, the estimations made for the bigger items are bigger than the estimations for the smaller items.

Overall, the estimations for unseen items (id 2) are relatively good in cases when the estimated volume is inside the volume range of the other items, as expected. The items from the classes grapes and strawberry have higher volumes than the other items, and here an underestimation of the volume is observed. We thus conclude that the performance of the volume estimation depends heavily on the data and especially on the volume distribution of the items in each class. As Table 5 shows, the overall accuracy of the model in the test set is significantly lower than the one on the training set, but still higher than the one on the validation set, signalling a difficulty in generalization to unseen objects.

4.3.2. *Balanced validation set*

To overcome the limitations of the data choice presented in the previous section, the balanced validation set has been designed on the basis of the.

food items absolute values (the crosses in Fig. 9). Since the classification accuracy is already excellent, our investigation has concentrated on the volume estimation performance. A summary of the MAPEs for the naive and the balanced approaches is shown in Table 5. As can be seen, the training set error decreases in the balanced case, while the distance between the test and validation set is generally reduced. This shows that the model is able to generalize to unseen objects.

Fig. 10 shows the detailed distribution of the errors per class, both in relative and absolute terms, only on the validation set. The relative error on its own is not enough to fully judge the model performance in some cases, for instance for the class *carrot* which only has objects with small volumes, thereby showing a large relative error but low in absolute terms. The balanced validation set has increased the error spread, as expected since it has many more data points than the naive validation set, but it has almost always decreased the median error. This shows once again the importance of the data choice for proper model generalizability. The results also indicate that, for the classes in which the distribution of volumes is wider (e.g., grapes, cucumber), more data points are needed to achieve accuracy on par with the ones of classes mapped better, such as avocado and strawberry.

4.4. *Comparisons*

In order to test the importance of all components of the proposed model, an ablation study has been conducted. Specifically, four configurations have been tested, where either the IMU data has been excluded, the LSTM for Volume Estimation has been replaced by a linear layer or no class information has been used. The results are reported in Table 6, and clearly show that all components of the model are necessary for achieving the best performance.

Due to the differences in the data (i.e., no IMU and video data in the literature, no RGB-D, reference, or echo in our dataset), a direct comparison with state-of-the-art approaches is not possible. A qualitative comparison between our best results from Table 6 and the

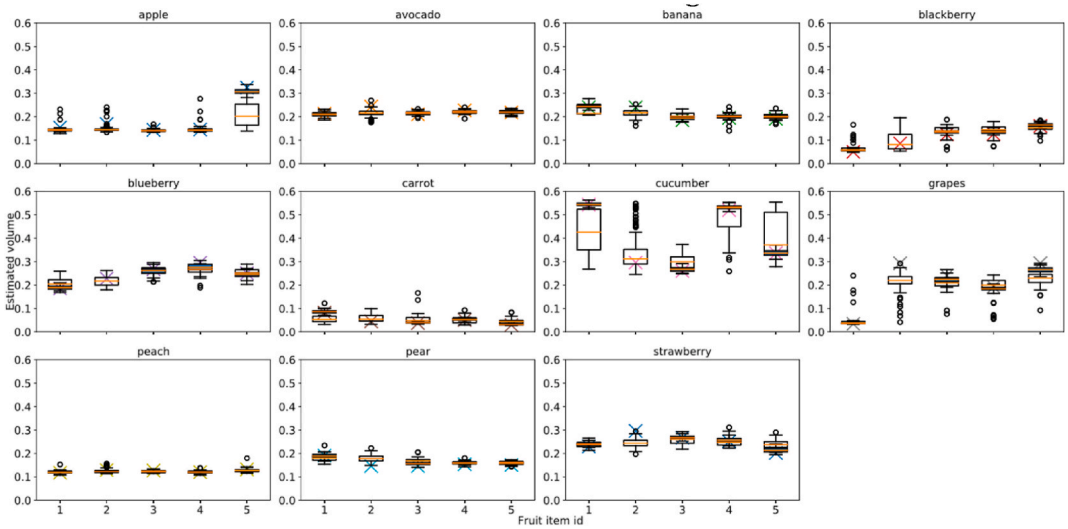

Fig. 9. Estimated volume distribution per item together with the true volume (the crosses) for each class using the naive validation set. For item id #2, the filled boxes are derived from the training data and the empty ones on the test data. First row, left to right: apple, avocado, banana, blackberry. Middle row, left to right: blueberry, carrot, cucumber, grapes. Last row, left to right: peach, pear, strawberry.

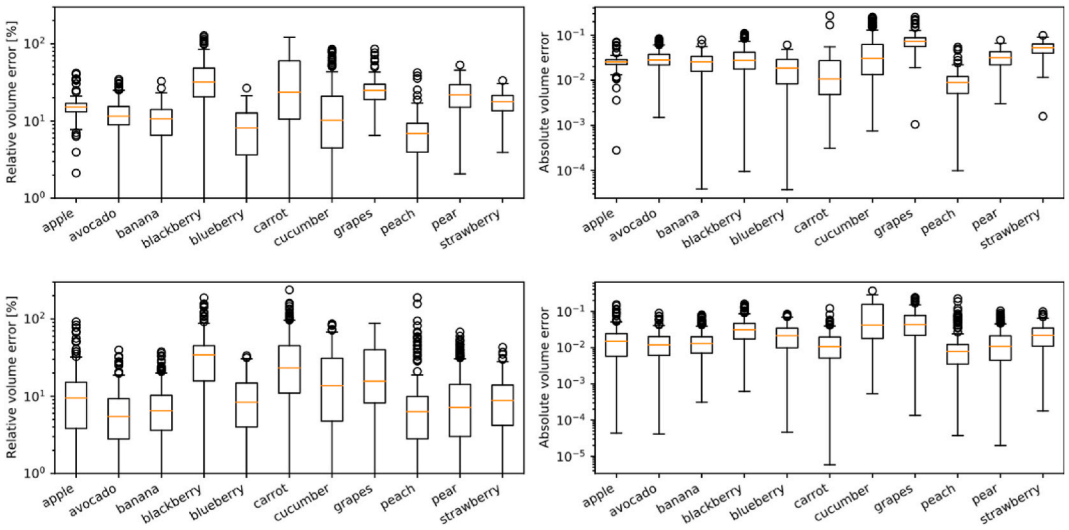

Fig. 10. Relative (left) and absolute (right) volume error distributions per class on the validation set using the different choices for the validation set: Naive validation (top) and balanced validation (bottom).

ones from Table 2 shows that our approach is very competitive, despite demanding a significantly simpler data acquisition. Only methods with reference objects and additional imaging modalities perform better than our proposed approach. Of those, only methods considering very few food items perform significantly better. In addition, the large MAPEs for some classes for our model, above 20%, are related to those classes that have objects with very small volumes, and therefore in these cases, the absolute error is low. This shows that basing the comparison between models solely on MAPE can be misleading in real-life scenarios.

4.5. *Uncertainty estimation*

As described in Sec. 3.2.3, six different ensemble models were trained. Within these models, MSE and GNLL loss are considered, while also three different training data split proportions are used to create the training set for each model in the ensemble. Fig. 11 shows a small difference between the distributions of the absolute values of the relative volume error of the trained ensemble models, obtained on the validation set. The small effect of the size of the training data on the model's performance in terms of volume prediction suggests that the model is simply overfitted with more data and even fewer data will suffice. An explanation can be found in the data set itself, where many samples are obtained for the same fruit/vegetable item with the same volume. We assume that a data set with more different volume values and fewer samples with the same volume would help to improve the volume estimation model. This hypothesis should be further evaluated on an extended dataset, which is left for future work.

Comparing the ensemble models with the different losses, Fig. 11 shows that the models trained with GNLL loss perform slightly better than those using MSE loss. We have further compared the effect of the loss function on the volume prediction uncertainty. For this purpose, we have estimated the variance of the volume as stated in Sec. 3.2.3. Assuming the predicted volume has a normal distribution, 95% of the predicted volume values should be below the 2σ threshold. Therefore, the percentage of samples that are above the 2σ threshold was inspected. The results, shown in Fig. 12, suggest a superior performance of the models that use GNLL loss. The models obtained with the GNLL loss not only have a lower volume error distribution, but they are also less uncertain and can be trusted better. The overall performance of the ensemble models with GNLL loss, involving the classification precision and the MAPE of the volume prediction, is shown in Table 7. Balancing between the volume prediction error and uncertainty, but also classification precision, a training set ratio of 33% obtains the preferred performance. Although it.

has a slightly higher volume error compared to the single model explained before, the ensemble model offers variance estimation that contributed to lower uncertainty in volume prediction.

5. Conclusions

We have presented a novel approach to estimate the volume of different types of fruits and vegetables from short video sequences consisting of monocular RGB video frames and associated inertial data capturing the motion of the smartphone. By balancing training and validation datasets to match.

the distribution of volumes in each class, we achieve a MAPE of 16.0% on the validation dataset consisting of only unseen items with unique volumes. This compares well to the state-of-the-art that has been obtained with more elaborate data recording setups. Our approach does not require reference objects to be present in the images, makes no assumptions about the 3D shape of the different food classes, and works with common sensors available in every smartphone. Our modular network architecture only requires ground-truth volume labels provided that a pretrained 2D object detector is available as the IMU encoder network is trained in an unsupervised way.

Table 6
Results of the ablation study, MAPE average, and ranges between classes.

Model	Training (%)	Test (%)	Validation (%)
Full model	3.8 (2.6–6.0)	16.4 (6.8–36.1)	16.0 (7.4–31.1)
No IMU data	3.6 (1.6–6.2)	18.3 (5.7–54.1)	16.4 (6.4–37.5)
No Vol. Estimation LSTM	5.8 (3.3–12.7)	24.3 (9.8–54.3)	19.3 (8.0–41.0)
No classes	6.5 (4.1–10.9)	43.3 (15.4–109.8)	39.2 (14.6–129.3)

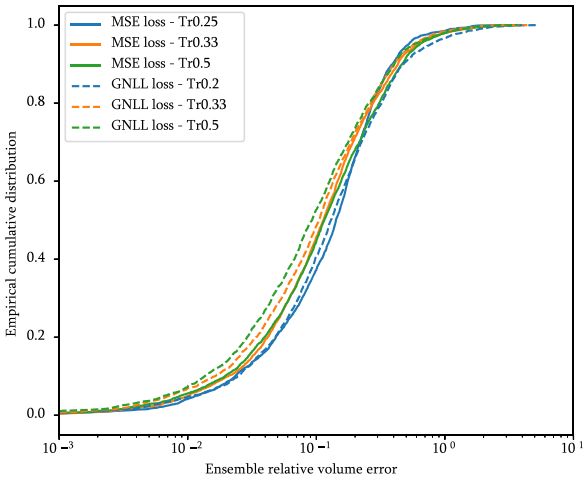

Fig. 11. Empirical cumulative distribution of the absolute relative volume error on the validation set using ensemble models with different choices for the loss function and training set ratio.

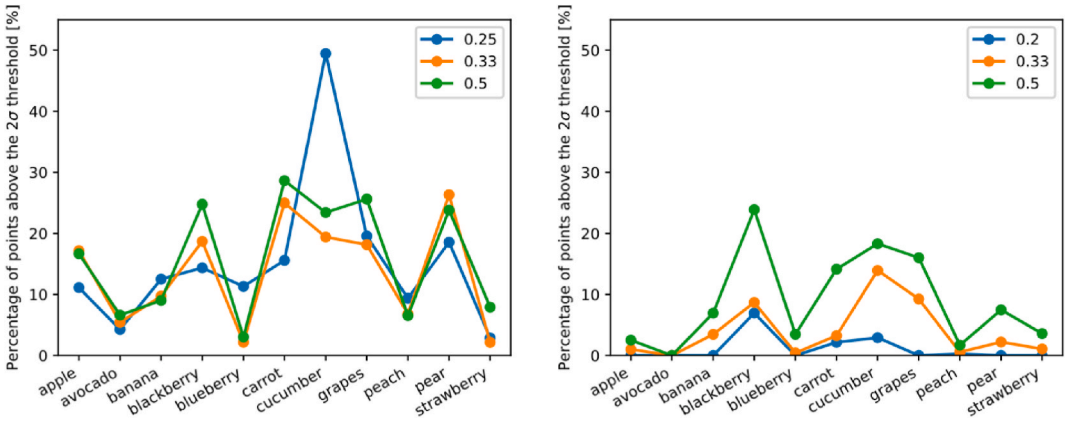

Fig. 12. Percentage of samples in the validation set with an absolute value of the relative volume error above the 2σ threshold using ensemble models with different choices for the training set ratio obtained with MSE loss (left) and GNLL loss (right).

Table 7
MAPE and class precision on the ensemble models that use the GNLL loss for three choices of the training set ratio.

	Training		Test		Validation	
	MAPE	Precision	MAPE	Precision	MAPE	Precision
GNLL, 0.2	18.6%	99.8%	28.6%	91.6%	23.2%	85.5%
GNLL, 0.33	9.5%	99.7%	22.1%	92.0%	17.9%	86.0%
GNLL, 0.5	5.0%	99.9%	20.5%	92.7%	17.6%	88.3%

Moreover, in order to address the increasingly important requirement of Explainable AI models, an accurate uncertainty estimate has been obtained, which allows to automatically disregard the cases of high model errors. In addition, the dataset has been constructed to mirror real-world use cases incorporating varying backgrounds, different angles, and distances from the object of interest, making it a challenging dataset. To enable further algorithm development in volume estimation of fruits and vegetables, the dataset along with the measured ground-truth volumes of the different items is made publicly available at https://sst.aau.at/cns/datasets. Future work will focus on expanding the current dataset to include more food classes and more food items per class with a larger variation in volume distribution. We will also evaluate the possibility to deploy our models directly on the smartphone for online inference.

Author contribution statement

Jan Steinbrener, Ph.D.; Vesna Dimitrievska; Federico Pittino: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Wrote the paper.

Frans Starmans; Roland Waldner; Jürgen Holzbauer; Thomas Arnold: Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data.

Funding statement

This work was performed within the COMET Centre ASSIC Austrian Smart Systems Integration Research Center, which is funded by BMK, BMDW, and the Austrian provinces of Carinthia and Styria, within the framework of COMET - Competence Centers for Excellent Technologies.

.

The COMET programme is run by FFG.

Data availability statement

Data associated with this study has been deposited at https://sst.aau.at/cns/datasets.

Declaration of interest's statement

The authors declare no competing interests.

References

[1] F.P.M. Hoevenaars, C.M.M. Berendsen, W.J. Pasman, T.J. van den Broek, E. Barrat, I.M. de Hoogh, S. Wopereis, Evaluation of food intake behavior in a healthy population: personalized vs. one-size-fits all, Nutrients 12 (9) (2020) 2819, https://doi.org/10.3390/nu12092819.

[2] S.C. Woods, M.W. Schwartz, D.G. Baskin, R.J. Seeley, Food intake and the regulation of body weight, Annu. Rev. Psychol. 51 (1) (2000) 255–277, https://doi.org/10.1146/annurev.psych.51.1.255.

[3] W.C. Knowler, S.E. Fowler, R.F. Hamman, C.A. Christophi, H.J. Hoffman, A.T. Brenneman, J.O. Brown-Friday, R. Goldberg, E. Venditti, D.M. Nathan, 10-year follow-up of diabetes incidence and weight loss in the diabetes prevention program outcomes study, Lancet 374 (9702) (2009) 1677–1686, https://doi.org/10.1016/s0140-6736(09)61457-4.

[4] F.E. Thompson, A.F. Subar, Dietary Assessment Methodology, Elsevier, 2017, pp. 5–48, https://doi.org/10.1016/b978-0-12-802928-2.00001-1.

[5] G.C. Marks, M.C. Hughes, J.C. van der Pols, Relative validity of food intake estimates using a food frequency questionnaire is associated with sex, age, and other personal characteristics, J. Nutr. 136 (2) (2006) 459–465, https://doi.org/10.1093/jn/136.2.459.

[6] D.F. Barbin, N.A. Valous, D.-W. Sun, Tenderness prediction in porcine longissimus dorsi muscles using instrumental measurements along with NIR hyperspectral and computer vision imagery, Innovat. Food Sci. Emerg. Technol. 20 (2013) 335–342, https://doi.org/10.1016/j.ifset.2013.07.005.

[7] A. Ulrici, G. Foca, M.C. Ielo, L.A. Volpelli, D.P.L. Fiego, Automated identification and visualization of food defects using RGB imaging: application to the detection of red skin defect of raw hams, Innovat. Food Sci. Emerg. Technol. 16 (2012) 417–426, https://doi.org/10.1016/j.ifset.2012.09.008.

[8] W. Jia, Y. Yue, J.D. Fernstrom, N. Yao, R.J. Sclabassi, M.H. Fernstrom, M. Sun, Imaged based estimation of food volume using circular referents in dietary assessment, J. Food Eng. 109 (1) (2012) 76–86, https://doi.org/10.1016/j.jfoodeng.2011.09.031.

[9] W. Jia, Y. Yue, J.D. Fernstrom, Z. Zhang, Y. Yang, M. Sun, 3d localization of circular feature in 2d image and application to food volume estimation, IEEE (2012), https://doi.org/10.1109/embc.2012.6346978.

[10] Z. Zhang, Y. Yang, Y. Yue, J.D. Fernstrom, W. Jia, M. Sun, Food volume estimation from a single image using virtual reality technology, IEEE (2011), https://doi.org/10.1109/nebc.2011.5778625.

[11] J. Chae, I. Woo, S. Kim, R. Maciejewski, F. Zhu, E.J. Delp, C.J. Boushey, D.S. Ebert, Volume estimation using food specific shape templates in mobile image-based dietary assessment, SPIE (2011), https://doi.org/10.1117/12.876669.

[12] C. Xu, Y. He, N. Khannan, A. Parra, C. Boushey, E. Delp, Image-based Food Volume Estimation, ACM Press, 2013, https://doi.org/10.1145/2506023.2506037.

[13] G.V. Venkatesh, S.M. Iqbal, A. Gopal, D. Ganesan, Estimation of volume and mass of axi-symmetric fruits using image processing technique, Int. J. Food Prop. 18 (3) (2014) 608–626, https://doi.org/10.1080/10942912.2013.831444.

[14] Y. He, C. Xu, N. Khanna, C.J. Boushey, E.J. Delp, Food image analysis: segmentation, identification and weight estimation, IEEE (2013), https://doi.org/10.1109/icme.2013.6607548.

[15] T. Suzuki, K. Futatsuishi, K. Kobayashi, Food volume estimation using 3d shape approximation for medication management support, IEEE (2018), https://doi.org/10.1109/acirs.2018.8467253.

[16] C. Xu, Y. He, N. Khanna, C.J. Boushey, E.J. Delp, Model-based food volume estimation using 3d pose, IEEE (2013), https://doi.org/10.1109/icip.2013.6738522.

[17] S.M. Iqbal, A. Gopal, A.S.V. Sarma, Volume estimation of apple fruits using image processing, IEEE (2011), https://doi.org/10.1109/iciip.2011.6108909.

[18] S. Fang, C. Liu, F. Zhu, E.J. Delp, C.J. Boushey, Single-view food portion estimation based on geometric models, IEEE (2015), https://doi.org/10.1109/ism.2015.67.

[19] M. Puri, Z. Zhu, Q. Yu, A. Divakaran, H. Sawhney, Recognition and volume estimation of food intake using a mobile device, IEEE (2009), https://doi.org/10.1109/wacv.2009.5403087.

[20] J. Dehais, M. Anthimopoulos, S. Shevchik, S. Mougiakakou, Two-view 3d reconstruction for food volume estimation, IEEE Trans. Multimed. 19 (5) (2017) 1090–1099, https://doi.org/10.1109/tmm.2016.2642792.
[21] A. Gao, F.P.-W. Lo, B. Lo, Food volume estimation for quantifying dietary intake with a wearable camera, IEEE (2018), https://doi.org/10.1109/bsn.2018.8329671.
[22] M.A. Subhi, S.H.M. Ali, A.G. Ismail, M. Othman, Food volume estimation based on stereo image analysis, IEEE Instrum. Meas. Mag. 21 (6) (2018) 36–43, https://doi.org/10.1109/mim.2018.8573592.
[23] J. Shang, M. Duong, E. Pepin, X. Zhang, K. Sandara-Rajan, A. Mamishev, A. Kristal, A mobile structured light system for food volume estimation, IEEE (2011), https://doi.org/10.1109/iccvw.2011.6130229.
[24] K. He, G. Gkioxari, P. Dollar, R. Girshick, Mask r-CNN, IEEE Trans. Pattern Anal. Mach. Intell. 42 (2) (2020) 386–397, https://doi.org/10.1109/tpami.2018.2844175.
[25] Y. Liang, J. Li, Computer Vision-Based Food Calorie Estimation: Dataset, Method, and Experiment, 2017 arXiv preprint arXiv:1705.07632.
[26] Y. Liu, J. Lai, W. Sun, Z. Wei, A. Liu, W. Gong, Y. Yang, Food volume estimation based on reference, ACM (2020), https://doi.org/10.1145/3390557.3394123.
[27] J. Gao, W. Tan, L. Ma, Y. Wang, W. Tang, Musefood: multi-sensorbased food volume estimation on smartphones, in: IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), IEEE, 2019, pp. 899–906.
[28] Y. Ando, T. Ege, J. Cho, K. Yanai, Depthcaloriecam: a mobile application for volume-based food calorie estimation using depth cameras, in: Proceedings of the 5th International Workshop on Multimedia Assisted Dietary Management, 2019, pp. 76–81.
[29] A. Graikos, V. Charisis, D. Iakovakis, S. Hadjidimitriou, L. Hadjileontiadis, Single Image-Based Food Volume Estimation Using Monocular Depth-Prediction Networks, 2020, pp. 532–543, https://doi.org/10.1007/978-3-030-49108-6_38. Springer International Publishing.
[30] F. Lo, Y. Sun, J. Qiu, B. Lo, Food volume estimation based on deep learning view synthesis from a single depth map, Nutrients 10 (12) (2005), https://doi.org/10.3390/nu10122005, 2018.
[31] Z. Yang, H. Yu, S. Cao, Q. Xu, D. Yuan, H. Zhang, W. Jia, Z.-H. Mao, M. Sun, Human-mimetic estimation of food volume from a single-view rgb image using an ai system, Electronics 10 (13) (2021) 1556.
[32] Q. Thames, A. Karpur, W. Norris, F. Xia, L. Panait, T. Weyand, J. Sim, Nutrition5k, Towards automatic nutritional understanding of generic food, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8903–8911.
[33] J. He, Z. Shao, J. Wright, D. Kerr, C. Boushey, F. Zhu, Multi-task image-based dietary assessment for food recognition and portion size estimation, in: 2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), IEEE, 2020, https://doi.org/10.1109/mipr49039.2020.00018.
[34] B.E. Moore, J.J. Corso, Fiftyone, GitHub, 2020. https://github.com/voxel51/fiftyone.