CONTAINER LOCALISATION AND MASS ESTIMATION WITH AN RGB-D CAMERA

Tommaso Apicella, Giulia Slavic, Edoardo Ragusa, Paolo Gastaldo and Lucio Marcenaro

DITEN, University of Genoa, Italy

ABSTRACT

In the research area of human-robot interactions, the automatic estimation of the mass of a container manipulated by a person leveraging only visual information is a challenging task. The main challenges consist of occlusions, different filling materials and lighting conditions. The mass of an object constitutes key information for the robot to correctly regulate the force required to grasp the container. We propose a single RGB-D camera-based method to locate a manipulated container and estimate its empty mass i.e., independently of the presence of the content. The method first automatically selects a number of candidate containers based on the distance with the fixed frontal view, then averages the mass predictions of a lightweight model to provide the final estimation. Results on the CORSMAL Containers Manipulation dataset show that the proposed method estimates empty container mass obtaining a score of 71.08% under different lighting or filling conditions.

***Index Terms*—** Convolutional Neural Networks, Object detection, Mass estimation

1. INTRODUCTION

Estimating the physical properties of objects is fundamental for the success and safety of human-robot collaboration [1]. A robot is a completely autonomous system and is supposed to extract information of the target object in order to assist the human correctly. Robots could be employed, for example, to help people perform housework, lift heavy loads or even bring medicines to the elderly. An incorrect prediction of the object properties could harm the human, e.g., dropping the object or spilling dangerous substances [2].

In a typical human-to-robot handover scenario, the person exchanges a container with a robot, which takes it from human hand(s) [1, 3]. The estimation of container properties such as its width, height and mass represents a crucial stage, since the robot regulates the force to hold the object during the handover and the maneuvering [2]. Moreover, it is not a trivial task since the object could be unknown [3, 4] or the physical properties of the container could change based on the interaction, e.g., deformation due to the grasp, or different stiffness and filling amounts [5].

The container mass can be indirectly retrieved by combining two properties: filling mass and empty container mass. The filling mass can be seen as the result of three contributions: filling type classification, filling level and container capacity estimation [6]. Recent exisiting solutions use the CORSMAL Containers Manipulation (CCM) dataset [7], which includes annotated audio-visual recordings of people interacting with containers [8]. To perform filling type classification, audio is one of the most used modality, either processing spectrograms [9] or classical audio features [10]. Solutions for filling level estimation can exploit only audio modality [11] or the combination with visual data [12, 13]. Visual clues represent the main modality used to estimate the container capacity. Approaches can consider the task as a regression problem, employing either simple CNNs on single fixed frontal view depth data [9] or distribution fitting via Gaussian processes using object category as a prior across multiple views [12]. Otherwise, the segmented container can be approximated to a primitive shape in 3D, computing capacity as a by-product [11], or using volume formulas [13].

Unlike previous works that focused on estimating the filling mass, we propose an approach to localise the container manipulated by a person and then estimate its empty mass (regardless of the content) from an RGB-D camera with fixed frontal view. Adopting human-to-robot handover as a use case, the strategy consists in automatically selecting a number of containers candidates based on the average distance with respect to the frontal view and then estimating the empty container mass using a lightweight CNN[1]. A similar CNN was previously devised to regress the container capacity using only depth image crops as input [9]. However, depth images can be noisy, and can contain missing values even after applying basic morphological operations (e.g., closing). Using both geometrical information (distance from depth map and aspect ratio with respect to the original resolution) and the colour information from the RGB frame increases the probability of localising the object and estimating the empty container mass.

2. LOCALISATION AND MASS ESTIMATION

The proposed method uses RGB-D data and is divided into three steps (see Fig. 1). The first step localises the container every n frames using Mask R-CNN [14] pre-trained on COCO [15], selecting only the classes *cup*, *book*, *wine glass* and *bottle* [13]. Assuming that the manipulated container is the nearest with respect to a fixed frontal view, the method automatically selects the nearest detected containers based on the average distance of the depth values belonging to the container mask (*K-nearest patches selection*). The last step is the *final prediction* in which the model performs one prediction per each patch and then empty container mass ($\hat{m}$) is computed as the average of the predictions.

2.1. Mass estimation model

The model, designed to be lightweight, takes as input RGB patches of the container, the width and height aspect ratios with respect to the original image resolution (a and b) and the average distance of the object from the camera (d). The rationale behind this last features (f) is to preserve some geometrical information of the distance between the container and camera, along with the aspect ratios. Similarly to [9], the model has four convolutional layers, two Fully Connected (FC) layers, concatenation of the output of the second FC layer with f, and one FC layer after the concatenation. Batch normalization and ReLU activations are used after each layer. Max Pooling is used after each convolutional layer. The size of convolutional kernels is

[1] Code available at: `https://github.com/CORSMAL/Visual`

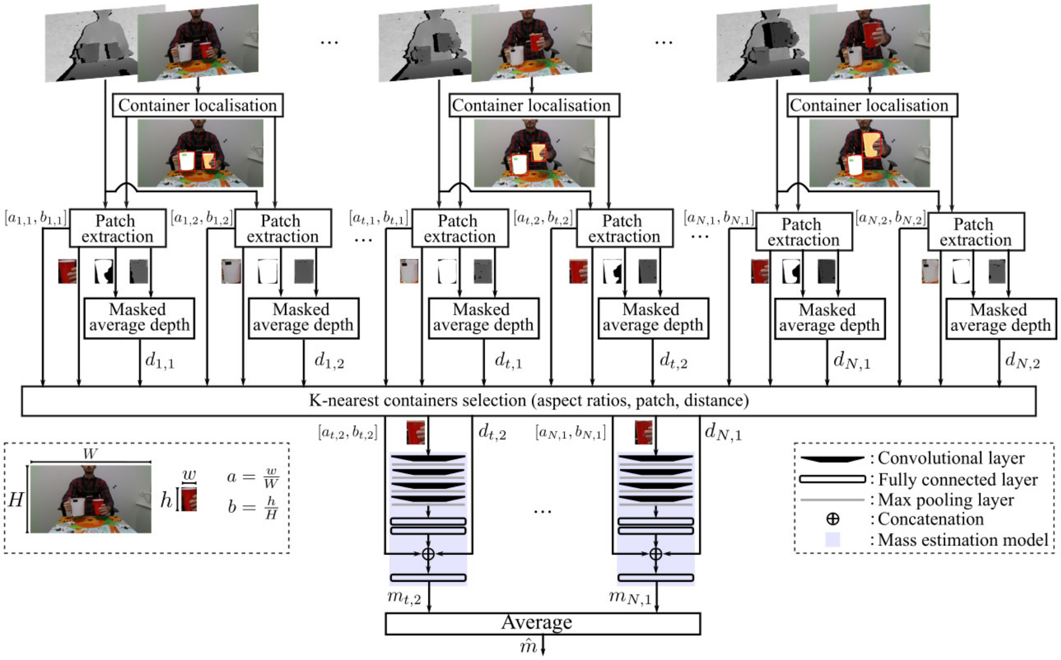

Fig. 1: Block diagram of the proposed approach. For each frame, containers are detected, then K nearest patches are selected leveraging the raw depth maps considered in the segmentation mask coordinates. The empty mass of each patch (m) is predicted by the model which takes as input the RGB patch and triplets of values: aspect ratio width (a) and height (b), and average distance (d). The final empty mass estimation ($\hat{m}$) is the average of K mass predictions.

$(3, 3)$, paddings and strides are $(1, 1)$, and channel dimensions are $(32, 64, 64, 128)$. The size of Max pooling kernel is $(2, 2)$. The output of the first two FC layers has dimension 64 and 6, respectively. Unlike [9] that uses depth patches and their aspect ratios $[a, b]$, our model takes as input RGB patches and the features $f = [a, b, d]$. The number of model parameters is 533,000.

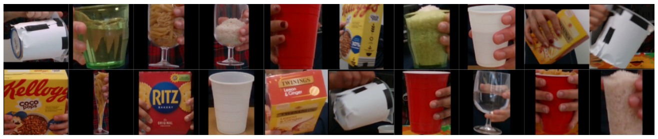

Fig. 2: Sample patches of the extracted dataset. Black padding is applied before resizing to keep the same aspect ratio.

2.2. Model training

The model is trained using the containers patches extracted using the first two steps of the described procedure on the CCM dataset. Mask R-CNN is applied to the entire dataset, similarly to Crop-CCM [16]. The main differences are that we use a single view of the scene, we do not restrict the model solely to cup and glass cases, and we do not perform a manual check of the results. The selection of the containers patches is indeed performed automatically during *K-nearest patches selection* phase. Images resolution is 1280 × 720, the detection threshold is set to 0.4 and every frame of the video is analysed ($n = 1$). In each frame, Mask R-CNN could find in general more than one object e.g. the pitcher used to fill the cup and the cup itself. The maximum number of considered containers candidates (K) is set to 5 (note: the model could also detect less patches in a recording). The rationale behind the parameter value choice is to have a trade-off to obtain enough patches to train the model and provide a more robust mass prediction, by averaging a number of candidates, as well as minimizing computational overhead. Our extracted dataset consists of 3,408 patches, some of them are shown in Fig. 2. The proposed automatic retrieval leads to almost 8% of the patches containing the pitcher used to fill containers, which are not annotated in CCM dataset. The empty mass annotation of each video is applied to each extracted patch.

The model is trained on the regression task. The patches are resized to 112 × 112 resolution, using zero padding on the shorter dimension in order to maintain the proportions, and are normalized to [0, 1] range. The following transformations are employed to augment the patches dataset: horizontal flip with probability 50%, vertical flip with probability 50%, random rotation between 0 and 180 degrees without cropping the patch, color jitter which consists in randomly changing the brightness into $[0.8, 1.2]$ range, contrast into $[0.8, 1.2]$ range, saturation into $[0.8, 1.2]$ range and hue into $[-0.2, 0.2]$ range. The aspect ratios, average distance and empty mass labels are normalized using the minimum and maximum values retrieved from the training set. The following setup is common to the experiments: mean square error loss, batch size 32, learning rate 0.0015 with an exponential decay rate of 0.9985 and with decay steps equal to 20; weight decay is set to 0.001.

3. EXPERIMENTAL SETUP

We evaluate our proposed model on the CCM dataset [7] which consists of 1140 audio-visual recordings. During each recording, a person interacts with a container (e.g., filling a cup with rice contained in a pitcher) and then prepares for the handover. Videos differ in conditions such as lighting, person's clothing, hand occlusions. The total number of containers in the dataset is 15: 9 (684 recordings)

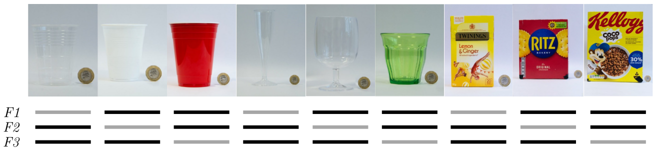

Fig. 3: 3-fold cross-validation setups ($F1$, $F2$, $F3$) of the CCM training set. Each fold selects videos from one instance of each container type as test set (▬), while videos belonging to the other instances are used as training set (▬).

constitute the training set, the other 6 are evenly split into a public test set (228 recordings) and private test set (228 recordings).

To assess the generalization performances of the method, we leave one instance per category (*box*, *glass*, *cup*) out, creating three folds [13], see Fig. 3. For each testing fold, the training set includes containers belonging to the other two folds, which are split into training and validation sets using 80% and 20% as respective percentages of data. The training set is augmented of 3 times using the described transformations. The model is trained using 100 epochs. The validation set is used for model selection and the best model, the one with the lowest mean loss, is kept for testing.

In addition to this 3-fold cross-validation, we also randomly split the whole training set in training and validation with the 80:20 ratio as before to include all available containers in the training phase. In this case, the method is validated through the CORSMAL Challenge which provides the results for the public test set and privates test set, since the annotations are private. Every patch in the training generates other 4 images using the described transformations. The model is trained for 300 epochs, as the number of training images increases with respect to 3-fold cross-validation. The same rationale of model selection is applied for this experiment.

As performance measures, we compute for each recording j the relative absolute error between the estimated measure $\hat{m}^j$, and the true measure m^j, as:

$$\epsilon(\hat{m}^j, m^j) = \frac{|\hat{m}^j - m^j|}{m^j} \tag{1}$$

The score $s \in [0, 1]$ (where 1 is best), across all J recordings for each measure is[2]:

$$s = \frac{1}{J} \sum_{j=1}^{J} \mathbb{1}_j \, e^{-\epsilon(\hat{m}^j, m^j)} \tag{2}$$

The value of the indicator function $\mathbb{1}_j \in \{0, 1\}$ is 0 only when $\hat{m}$ in recording j is not estimated.

4. RESULTS

Fig. 4 (top) analyses the per-class scores (colored based on *cup*, *glass* and *box*) and the *total* score for the three splits (*F1*, *F2*, *F3*) and the validation set (*VAL*). The whole score can be obtained using Eq. 2 or by multiplying per-class score by the number of instances, summing and then dividing by the total number of instances. Fig. 4 (bottom) provides a complementary analysis through the per-class mean of relative absolute error ϵ. Overall, Fig. 4 shows that the model does not generalize to testing containers significantly different from the training ones. The model achieves the highest score and the lowest mean error for the class *box*, probably due to the fact that they feature different colors and have a larger shape with respect to other classes. The mean relative absolute error in the *cup* cases suggests that the empty mass for this class is not properly learned. A possible explanation for the low performance on the first test fold is that the training images contain colored and opaque *cups*, whereas in the test set *cups* are transparent, and the only transparent containers in the used training folds are *glasses*. Other folds statistics point out that the presence of similar containers between training and test sets helps reducing the error and improving the score. The values of mean relative absolute error for *glass* and *box* classes fall in the $[0, 1]$ range, while the *cup* class drops with respect to fold *F1*, yet it remains higher than 1. The last group of bars in Fig. 4 shows the results for the validation set predictions. The score value underlines that the chosen model is able to learn from recordings having the same containers, yet in different configurations, e.g., lighting conditions or filling. Also in this case, the class that impacts the most on the score is *cup*.

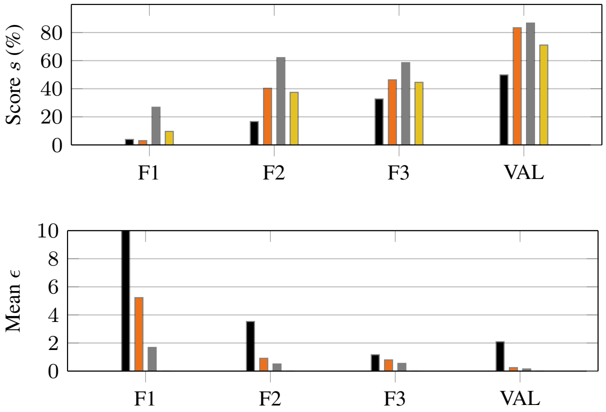

Fig. 4: Analysis per container type of 3-fold cross validation and random cross-validation of our proposed model for container empty mass estimation. Top: testing score s in percentage. Bottom: mean of relative absolute error ϵ. The maximum y-axis value is set to 10 for visualization purpose, the actual value for *cup* in *F1* is 22.952. Legend: ▬ *cup*, ▬ *glass*, ▬ *box* ▬ *total*

The performance score of CORSMAL challenge methods evaluated on private and public test set are shown in Table 1. Method 1 (*M1*) exploits RGB-D data from the fixed frontal view, extracts object patches using an object detector (YoloV5[3]), then predicts the empty container mass using an efficient model (MobileNetV2 [17]) enhanced with attention mechanism and pre-trained on container dimension estimation [18]. The eventual mass prediction is obtained by averaging the mass predictions. Method 2 (*M2*) regresses the empty container mass using a custom CNN which combines: the patch of the container extracted using a formula to select the most visible view (across fontal, left and right views of the scene), the symmetrically restored object mask from left side fixed view, and information about container dimensions (height, width at the bottom, width at the top) [19]. The container detection is performed by Localisation and object Dimensions Estimator (LoDE) [6]. Our model ($M3$) is the one providing *VAL* results of Fig. 4. As baselines for comparison, we consider also a pseudo-random generator (*M4*) that draws the predictions from a uniform distribution in the interval $[1, 351]$ based on the Mersenne Twister algorithm [20], and average (*M5*) computed on mass labels. In general, some similar

[2] https://corsmal.eecs.qmul.ac.uk/challenge.html

[3] https://github.com/ultralytics/yolov5

Table 1: Public and private test scores of container mass estimation solutions in percentage.

Set	M1	M2	M3 (Ours)	M4	M5
Public test	55.25	43.61	53.14	30.59	21.88
Private test	62.32	36.77	46.14	28.25	22.24
Combination	58.78	40.19	49.64	29.42	22.06

features across the methods can be highlighted e.g. the employment of two-stage approach (first detection then mass estimation) to tackle the problem and the use of lightweight models to perform the mass prediction. Our method achieves a higher score with respect to *M2*, *M4* and *M5*. Contrary to *M2*, our method does not use LoDE during the container detection phase and exploits only the fixed frontal view. The generalization properties on private test set show that *M1* performs better than other models, probably due to the employment attention mechanisms. Compared to this solution, our model is not pre-trained on other container properties estimation and features less parameters than MobileNetV2 model.

5. CONCLUSION

This paper provides a method to analyze a video, extract the container subject to manipulation based on its distance with respect to the fixed frontal view, and estimate its mass regardless of the content using a lightweight model. The low percentage of pitcher patches suggests that the proposed method is able to locate the manipulated container in training recordings. In the experiments, the model learns from similar containers, but poorly generalizes to unseen containers, especially *cups*. As future work, we will investigate multi-task learning to also classify the container type and improve generalization, as well as audio-visual perception to exploit the complementarity of the audio modality. Moreover, the case of camera mounted on the robot will be considered to analyse how partial views of the container due to movement affect model predictions.

6. REFERENCES

[1] V. Ortenzi, A. Cosgun, T. Pardi, W. P. Chan, E. Croft, and D. Kulić, "Object Handovers: a Review for Robotics," *IEEE Trans. Robotics*, vol. 37, no. 6, pp. 1855–1873, 2021.

[2] Y. L. Pang, A. Xompero, C. Oh, and A. Cavallaro, "Towards safe human-to-robot handovers of unknown containers," in *IEEE Int. Conf. Robot and Human Interactive Comm.*, 2021.

[3] P. Rosenberger, A. Cosgun, R. Newbury, J. Kwan, V. Ortenzi, P. Corke, and M. Grafinger, "Object-Independent Human-to-Robot Handovers using Real Time Robotic Vision," *IEEE Robotics Autom. Lett.*, vol. 6, no. 1, pp. 17–23, 2021.

[4] W. Yang, C. Paxton, A. Mousavian, Y. Chao, M. Cakmak, and D. Fox, "Reactive Human-to-Robot Handovers of Arbitrary Objects," in *IEEE Int. Conf. Robotics Autom.*, 2021.

[5] R. Sanchez-Matilla, K. Chatzilygeroudis, A. Modas, N. Ferreira Duarte, A. Xompero, P. Frossard, A. Billard, and A. Cavallaro, "Benchmark for Human-to-Robot Handovers of Unseen Containers with Unknown Filling," *IEEE Robotics Autom. Lett.*, vol. 5, no. 2, 2020.

[6] A. Xompero, R. Sanchez-Matilla, A. Modas, P. Frossard, and A. Cavallaro, "Multi-View Shape Estimation of Transparent Containers," in *IEEE Int. Conf. Acoustics, Speech and Signal Process.*, 2020.

[7] A. Xompero, R. Sanchez-Matilla, R. Mazzon, and A. Cavallaro, "CORSMAL Containers Manipulation," 2020, (1.0) [Data set]. Queen Mary University of London. https://doi.org/10.17636/101CORSMAL1.

[8] A. Xompero, S. Donaher, V. Iashin, F. Palermo, G. Solak, C. Coppola, R. Ishikawa, Y. Nagao, R. Hachiuma, Q. Liu, F. Feng, C. Lan, R. H. M. Chan, G. Christmann, J.-T. Song, G. Neeharika, C. K. T. Reddy, D. Jain, B. U. Rehman, and A. Cavallaro, "The CORSMAL benchmark for the prediction of the properties of containers," arXiv:2107.12719v2, 2021.

[9] G. Christmann and J. Song, "2020 CORSMAL Challenge - Team NTNU-ERCReport," 2020, https://corsmal.eecs.qmul.ac.uk/resources/challenge/2020.11.30_CORSMAL_NTNU-ERC_Report.pdf.

[10] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the Properties of Neural Machine Translation: Encoder-Decoder Approaches," arXiv:1409.1259v2, 2014.

[11] R. Ishikawa, Y. Nagao, R. Hachiuma, and H. Saito, "Audio-Visual Hybrid Approach for Filling Mass Estimation," in *IEEE Conf. Pattern Recognit. Workshops and Challenges*, 2021.

[12] Q. Liu, F. Feng, C. Lan, and R. H. M. Chan, "VA2Mass: Towards the Fluid Filling Mass Estimation via Integration of Vision and Audio Learning," in *IEEE Conf. Pattern Recognit. Workshops and Challenges*, 2021.

[13] V. Iashin, F. Palermo, G. Solak, and C. Coppola, "Top-1 CORSMAL Challenge 2020 Submission: Filling Mass Estimation Using Multi-Modal Observations of Human-Robot Handovers," in *IEEE Conf. Pattern Recognit. Workshops and Challenges*, 2021.

[14] K. He, G. Gkioxari, P. Doll'ar, and R. B. Girshick, "Mask R-CNN," in *IEEE Int. Conf. Comput. Vis.*, 2017.

[15] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," in *Eur. Conf. Comput. Vis.*, 2014.

[16] A. Modas, A. Xompero, R. Sanchez-Matilla, P. Frossard, and A. Cavallaro, "Improving Filling Level Classification with Adversarial training," in *IEEE Int. Conf. Image Process.*, 2021.

[17] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018.

[18] H. Wang, C. Zhu, Z. Ma, and C. Oh, "Improving Generalization of Deep Networks for Estimating Physical Properties of Containers and Fillings," in *IEEE Int. Conf. Acoustics, Speech and Signal Process., Grand Challenges: Audio-Visual Object Classification for Human-Robot Collaboration*, 2022.

[19] T. Matsubara, S. Otsuki, Y. Wada, H. Matsuo, T. Komatsu, Y. Iioka, K. Sugiura, and H. Saito, "Shared Transformer Encoder with Mask-Based 3D Model Estimation for Container Mass Estimation," in *IEEE Int. Conf. Acoustics, Speech and Signal Process., Grand Challenges: Audio-Visual Object Classification for Human-Robot Collaboration*, 2022.

[20] M. Matsumoto and T. Nishimura, "Mersenne Twister: a 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator," *ACM Trans. Modeling Comput. and Simulation*, vol. 8, no. 1, pp. 3–30, 1998.