Authors' pre-print.

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier XX.XXXX/ACCESS.2021.DOI

The CORSMAL benchmark for the prediction of the properties of containers

ALESSIO XOMPERO[1], SANTIAGO DONAHER[1], VLADIMIR IASHIN[2], FRANCESCA PALERMO[1], GÖKHAN SOLAK[1], CLAUDIO COPPOLA[1], REINA ISHIKAWA[3], YUICHI NAGAO[3], RYO HACHIUMA[3], QI LIU[4], FAN FENG[4], CHUANLIN LAN[4], ROSA H. M. CHAN[4], GUILHERME CHRISTMANN[5], JYUN-TING SONG[5], GONUGUNTLA NEEHARIKA[6], CHINNAKOTLA K. T. REDDY[6], DINESH JAIN[7], BAKHTAWAR UR REHMAN[8], ANDREA CAVALLARO[1]

[1]Centre for Intelligent Sensing, Queen Mary University of London, UK (e-mail: {a.xompero,s.donaher,f.palermo,g.solak,c.coppola,a.cavallaro}@qmul.ac.uk)
[2]Tampere University, Finland (e-mail: vladimir.iashin@tuni.fi)
[3]Keio University, Japan (e-mail: {reina-ishikawa,soccerbass03,ryo-hachiuma}@keio.jp)
[4]City University of Hong Kong (e-mail:{qi.liu,fan.feng,cllan2-c,rosachan}@cityu.edu.hk)
[5]National Taiwan Normal University, Taiwan (e-mail: {guichristmann,ralphydineen}@gmail.com)
[6]IIT Bhubaneswar, India (e-mail: {gn12,ck13}@iitbbs.ac.in)
[7]IIT Hyderabad, India (e-mail: ee17mtech11005@iith.ac.in)
[8]miftah.ai, Pakistan (e-mail: rehmans.285@gmail.com)

Corresponding author: Alessio Xompero (e-mail: a.xompero@qmul.ac.uk).

This work is supported by the CHIST-ERA program through the project CORSMAL, under UK EPSRC grant EP/S031715/1.

ABSTRACT Acoustic and visual sensing can support the contactless estimation of the weight of a container and the amount of its content when a person manipulate them. However, opaqueness and transparencies (both of the container and of the content) and the variability of materials, shapes and sizes make this problem challenging. In this paper, we present an open framework to benchmark methods for the estimation of the capacity of a container, and the type, mass, and amount of its content. The framework includes a dataset, well-defined tasks and performance measures, baselines and state-of-the-art methods, and an in-depth comparative analysis of these methods. Deep learning with neural networks with audio alone or a combination of audio and visual data are used by the methods to classify the type and amount of the content, either independently or jointly. Regression and geometric approaches with visual data are preferred to determine the capacity of the container. Results show that classifying the content type and level with methods that use only audio as input modality achieves a weighted average F1-score up to 81% and 97%, respectively. Estimating the container capacity with vision-only approaches and filling mass with audio-visual, multi-stage algorithms reaches up to 65% weighted average capacity and mass scores.

INDEX TERMS Acoustic signal processing, image and video signal processing, audio-visual classification, object properties recognition

I. INTRODUCTION

PEOPLE interact daily with household containers, such as cups, drinking glasses, mugs, bottles and food boxes. Methods to estimate the physical properties (e.g., weight and shape) of these containers could support human-robot cooperation [1]–[5], video annotation and captioning. Methods should generalize to unknown container instances and operate with only limited prior knowledge, such as generic categories of containers and contents [1], [6], [7]. However, the material, texture, transparency and shape vary considerably across containers and may change with the content. Furthermore, the content may not be visible due to the opaqueness of the container or because of hand occlusions. For these reasons, the predictions of the properties of containers is a very challenging task. The combination of sensing modalities, such as RGB images, depth, and audio, may help to overcome the challenges mentioned above. For example, noisy scenarios, already filled containers with absence of sound, occlusions, or transparent objects whose depth data may be highly inaccurate [8].

Existing methods focus on object recognition, object shape and size reconstruction in 3D, as well as pose estimation of a variety of objects using visual data and objects standing on a surface [9]–[16]. Object properties, such as transparency,

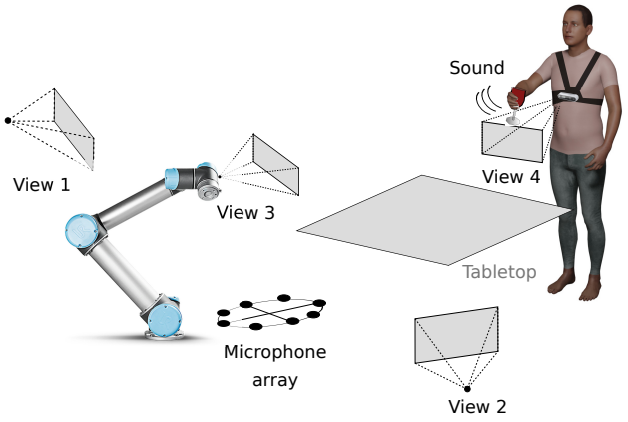

FIGURE 1. The multi-modal, multi-sensor system used to record a person manipulating a container and its content. The system includes two third-person view cameras (at the two sides of the robot), a first-person view camera mounted on the robot, a first-person view from the body-worn camera on the person and a 8-microphone circular array (placed next to the robot arm).

are often tackled independently with ad-hoc designed approaches [8], [17]–[19]. Methods that estimate only the *content level* usually tackle the problem either using a single image [7], [20], exploiting temporal information from sequences of RGB or RGB-D data to track the change in the amount during a mechanical action [21]–[23], or processing the sound signals generated by the contact of the content with a container during a manipulation [24]–[27]. For example, the level of unknown liquids within containers standing on a surface is regressed or classified by using approaches such as Kalman Filter and recurrent neural networks (e.g., with a Long Short-Term Memory unit) with edge features or spectrograms [22], [23], [27].

Recognizing the *content type* within a container is a well-investigated problem for general food recognition using visual information [28]–[30]. However, these approaches are often designed and evaluated on different scenarios, on a single specific physical property, with limited variability in the data. Recognizing different high-level properties, such as the amount and type of multiple materials, the capacity of the container, and the overall weight of the object (i.e., the container with its content) is not yet well-investigated.

In this paper, we present a novel, open framework for the design, evaluation, and comparison of methods that estimate the physical properties of household containers and their content, when a person manipulates the container (see Fig. 1). The framework includes a multi-modal dataset, well-defined tasks and corresponding performance measures, as well as baselines for estimating the type and amount of the content (Sec. II). We also review state-of-the-art methods that used the CORSMAL framework (Sec. III) and we carry out an in-depth comparative analysis of these methods (Sec. IV). Finally, we discuss future directions research directions based on the experience of an international benchmarking challenge[1] organized using the framework (Sec. V).

[1]https://corsmal.eecs.qmul.ac.uk/challenge2020.html

II. BENCHMARKING FRAMEWORK

A. CONTAINERS, FILLINGS, SCENARIOS

The dataset includes audio-visual-inertial recordings of people manipulating a range of containers that vary in shape, size, material, transparency, and deformability, and a set of contents under different scenarios with increasing level of difficulty due to the type of occlusions.

CORSMAL Containers Manipulation [31] is a dataset consisting of 1,140 audio-visual recordings with 12 human subjects manipulating 15 containers, split into 5 cups, 5 drinking glasses, and 5 food boxes. These containers are made of different materials, such as plastic, glass and cardboard. Each container can be empty or filled with water, rice or pasta at two different levels of fullness: 50% and 90% with respect to the capacity of the container. The combination of containers and contents results in a total of 95 configurations acquired for three scenarios with an increasing level of difficulty caused by occlusions or subject motions.

In the first scenario, the subject sits in front of the robot, while a container is on a table. The subject either pours the content into the empty container, while avoiding touching the container, or shakes an already filled food box. Afterwards, the subject initiates the handover of the container to the robot. In the second scenario, the subject sits in front of the robot, while holding a container before starting the manipulation. In the third scenario, a container is held by the subject while standing to the side of the robot, potentially visible only on the third-person camera view. After the manipulation, the subject takes a few steps and initiates the handover of the container in front of the robot. Each scenario is recorded with two different backgrounds and under two different lighting conditions. The first background condition involves a plain tabletop with the subject wearing a texture-less t-shirt, while the second background condition involves the table covered with a graphics-printed tablecloth and the subject wearing a patterned shirt. The first lighting condition is based on artificial illumination as provided by lights mounted on the ceiling of the room. The second lighting condition uses two controlled artificial lights placed at the sides of the robot and illuminating the area where the manipulation is happening. Each subject executed the 95 configurations for each scenario and for each background/illumination condition[2].

B. SENSOR DATA AND ANNOTATION

The dataset was acquired with 4 multi-sensor devices, Intel RealSense D435i, and an 8-element circular microphone array. Each D435i device consists of 3 cameras and provides spatially aligned RGB, narrow-baseline stereo infrared, and depth images at 30 Hz with 1280x720 pixels resolution. One D435i is mounted on a robot arm that does not move during the acquisition and provides a more realistic view of the operating area from the robot perspective. Another D435i

[2]Ethical approval (QMREC2344a) was obtained at Queen Mary University of London, and consent from each person was collected prior to data collection.

is chest mount by the person to provide a first-person view, while the remaining two devices are placed at the sides of the robot arm as third-person views that look at the operating area. The microphone array is placed on a table and consists of 8 Boya BY-M1 omnidirectional Lavelier microphones arranged in a circular shape of radius 15 cm. Audio signals are sampled synchronously at 44.1 kHz with a multi-channel audio recorder. All signals are software-synchronized with a rate of 30 Hz. The calibration information (intrinsic and extrinsic parameters) for each D435i and the inertial measurements of the D435i used as body-worn camera are also provided.

The annotation of the data includes the capacity of the container, the content type, the content level, the mass of the container, the mass of the content, the maximum width and height (and depth for boxes) of each object. Fig. 2 shows the total object mass across containers and their contents.

The dataset is split into training set (684 recordings of 9 containers), public test set (228 recordings of 3 containers), and private test set (228 recordings of 3 containers). The containers for each set are evenly distributed among the three categories. The annotations of the container capacity, content type and level, and the masses of the container and content are provided publicly only for the training set.

C. TASKS AND PERFORMANCE SCORES

We define three tasks for the framework, namely the classification of the amount of content (Task 1), the classification of the content type (Task 2), and the estimation of the capacity of the container (Task 3). We refer to the amount of content as filling level and to the type of content as filling type.

In Task 1, a container is either empty or filled with an unknown content at 50% or 90% of its capacity. There are three classes, $\Lambda = \{empty, half-full, full\}$. For each configuration j, the goal is to classify the filling level ($\lambda^j \in \Lambda$). In Task 2, containers are either empty or filled with an unknown content. There are four filling type classes, $\mathcal{T} = \{none, pasta, rice, water\}$. For each configuration j, the goal is to classify the type of filling, if any ($\tau^j \in \mathcal{T}$). For these two tasks, we compute precision, recall, and F1-score for each class k across all the configurations belonging to class k, J_k. *Precision* is the number of true positives over the total number of true positives and false positives for each class k (P_k). *Recall* is the number of true positives over the total number of true positives and false negatives for each class k (R_k). *F1-score* is the harmonic mean of precision and recall for each class k and defined as

$$F_k = 2\frac{P_k R_k}{P_k + R_k}. \tag{1}$$

We then compute the weighted average F1-score, $\bar{F}_1$, across the K classes,

$$\bar{F}_1 = \sum_{k=1}^{K} \frac{J_k F_k}{J}, \tag{2}$$

where $J = \sum_{k=1}^{K} J_k$ is the total number of configuration and J_k is the subset of configurations belonging to class k. Note that $K = 3$ for filling level classification, whereas $K = 4$ for filling type classification.

In Task 3, containers vary in shape and size. For each configuration j, the goal is to estimate the capacity of the container ($\gamma^j \in \mathbb{R}_{>0}$, in milliliters). For capacity estimation, we compute the relative absolute error between the estimated capacity, $\tilde{\gamma}^j$, and the annotated capacity, γ^j, for each configuration, j,

$$\varepsilon^j = \frac{|\tilde{\gamma}^j - \gamma^j|}{\gamma^j}. \tag{3}$$

We then compute the average capacity score, $\bar{C}$, as

$$\bar{C} = \frac{1}{J}\sum_{j=1}^{J} \mathbb{1} e^{-\varepsilon^j}, \tag{4}$$

The indicator function $\mathbb{1} \in \{0, 1\}$ is 0 only when the capacity (mass) of the container in configuration j is not estimated.

The weight of the object, $\omega \in \mathbb{R}_{>0}$ (in Newtons), is the sum of the mass of the (empty) container, $m_c \in \mathbb{R}_{>0}$ (in grams), and the mass of the (unknown) filling, $m_f \in \mathbb{R}_{>0}$ (in grams), multiplied by the gravitational earth acceleration, $g = 9.81$ m/s^{-2},

$$\omega = (m_c + m_f)g. \tag{5}$$

While we do not require the mass of the empty container to be estimated, we expect methods to estimate the capacity of the container and to determine the type and amount of filling to estimate the mass of the filling. For each configuration j, we then compute the filling mass as

$$m_f^j = \lambda^j \gamma^j D(\tau^j), \tag{6}$$

where $D(\cdot)$ selects a pre-computed density based on the classified filling type. The density of pasta and rice is computed from the annotation of the filling mass, capacity of the container, and filling level for each container. Density of water is 1 g/mL. For filling mass estimation, we compute the relative absolute error between the estimated, $\tilde{m}_f^j$, and the annotated filling mass, m_f^j, for each configuration, j, unless the annotated mass is zero (empty filling level),

$$\epsilon^j = \begin{cases} 0, & \text{if } m_f^j = 0 \wedge \hat{m}_f^j = 0, \\ \tilde{m}_f^j, & \text{if } m_f^j = 0 \wedge \tilde{m}_f^j \neq 0, \\ \frac{|\tilde{m}_f^j - m_f^j|}{m_f^j}, & \text{otherwise.} \end{cases} \tag{7}$$

Similarly to the average capacity score, we compute the average filling mass score, $\bar{M}$.

Note that we will present the scores as percentages when discussing the results in the comparative analysis.

D. BASELINES

CORSMAL provides along with the framework a set of 13 audio and video baselines for the filling level and filling type classification.

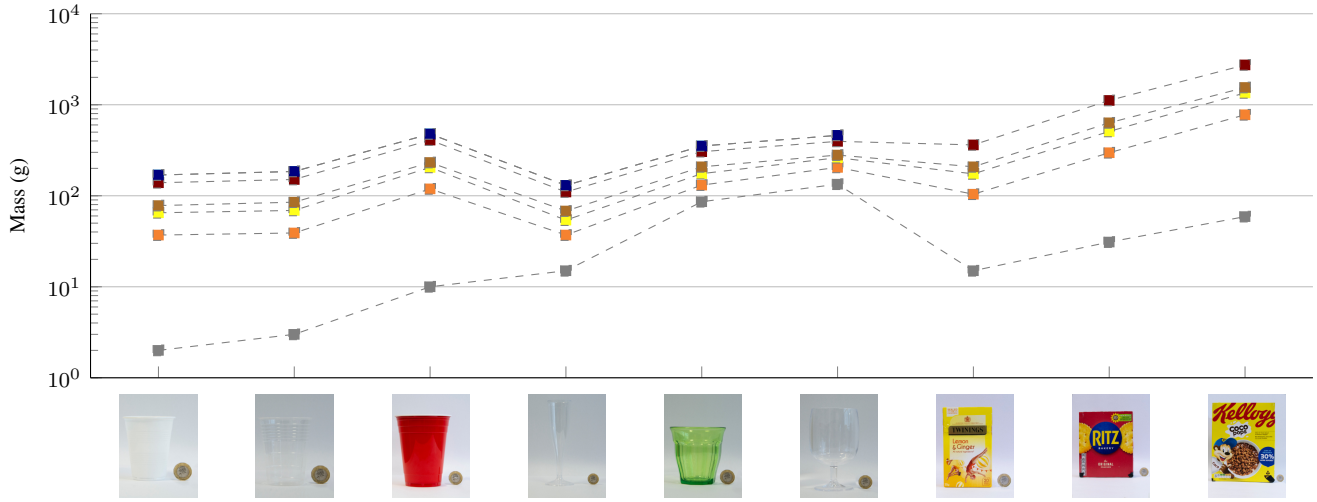

FIGURE 2. The mass of objects (container and content) in the training set of the CORSMAL Containers Manipulation dataset. The class *empty* corresponds to the mass of the container, which is known. Legend: Empty, P5, P9, R5, W5, R9, W9,

As audio-only baselines, we compute traditional acoustic features, such as spectrograms, zero-crossing rate (ZCR), Mel-frequency Cepstrum Coefficients (MFCCs), chromogram, Mel-scaled spectrogram, spectral contrast, and tonal centroid features (tonnetz), and classify filling type and level jointly. For MFCCs, the 1st to 13th coefficients are used, whereas the 0th coefficient is discarded. Similar to the baselines for the environmental sound classification [32], three baselines use k-Nearest Neighbour (kNN) [33], Support Vector Machine (SVM) [34], and Random Forest (RF) [35], respectively, as classifiers whose input is given by the mean and standard deviation of the MFCCs and ZCR features across multiple audio frames. Other 3 baselines extract a feature vector consisting of 193 coefficients from the mean and standard deviation of the MFCCs, chromogram, Mel-scaled spectrogram, spectral contrast, and tonnetz across multiple audio frames, as commonly used in the literature [36]–[39]. For simplicity, we refer to this set of acoustic features as AF193 in the rest of the paper. From the comparison in [40], we select three baselines that use spectrograms as input to the classifiers. The spectrograms are cropped, resized and reshaped into a vector of dimension 9,216. To remove redundant information, three additional baselines perform dimensionality reduction with Principal Component Analysis (PCA) on the reshaped spectrograms, retaining only the first 128 components.

As vision-only baseline, we use two CNNs to perform an independent classification of filling level and filling type from a single image, following previous works [7], [20]. Unlike [20] that used transfer learning and [7] that combined transfer learning with adversarial training, we re-trained ResNet-18 architectures [41] using a subset of frames[3] selected within the video recordings of our training set and cropped to a rectangular area around the container [7]. On the test sets, the baseline is applied to each camera view independently: an image crop is extracted from the last frame using Mask R-CNN [9] and the segmentation mask with the most confident class between *cup* and *wine glass* is selected. The output classes of the two CNNs include an additional class, *opaque*, to handle cases where containers are not transparent and vision alone fails to determine the content type and level [7], [20].

[3]Data available at: https://corsmal.eecs.qmul.ac.uk/filling.html

III. METHODS

We briefly review state-of-the-art methods that used the CORSMAL framework for the estimation of the filling level, filling type, and container capacity [40], [42]–[45]. For simplicity, we refer to the 6 methods as M1, M2 [42], M3 [43], M4 [44], M5 [45] and M6 [40]. These methods address filling type and filling level classification either independently, e.g., when only one of the two properties is necessary for the target application, or jointly, e.g., when both properties are necessary for accurately estimating the total object weight. We discuss the methods based on the modalities used as input, the features extracted, and the type of approach (regression, classification, or geometry-based).

Table 1 summarizes the methods discussed in this section.

A. INDEPENDENT FILLING LEVEL AND FILLING TYPE CLASSIFICATION

For filling type classification, methods preferred audio as input modality and adopted either only CNNs, CNN with RNN, or CNN followed by majority voting as classification approaches [42], [44], [45]. For filling level classification, some methods used also visual data in combination with audio data [43], [45]. Methods computed either traditional, learned, or both traditional and learned acoustic features. Traditional acoustic features, such as MFCCs, spectral characteristics, ZCR, chroma vector and deviation, are computed from short-term windows. Long-term features can be obtained by

TABLE 1. Methods for filling level, filling type, and container capacity estimation. Methods are evaluated on the CORSMAL Container Manipulation dataset.

Ref.	FL	FT	CC	Description	App	JLT	L	Gr	A	R	D	Temp.
M6 [40]	●	●	○	Cropped, resized, reshaped spectrogram + kNN/SVM/RF	C	●	●	●	●	–	–	○
	●	●	○	Cropped and resized spectrogram + CNN	C	●	●	●	●	–	–	○
	●	●	○	Cropped and resized spectrogram + Hierarchy of 3 CNNs	C	●	●	●	●	–	–	○
M1	●	●	○	STFT + FCNN	C	○	●	●	●	–	–	○
M2 [42]	○	●	○	MFCCs + CNN	C	○	●	●	●	–	–	○
	○	○	●	CNN with region of interest and bounding box size	R	○	–	–	○	1	1	○
M3 [43]	●	●	○	Spectrogram + object-specific MLP selected via majority voting of per-frame object detection across multiple views	C	○	●	●	●	4	–	○
	○	○	●	Gaussian processes	R	○	–	–	○	4	–	○
M4 [44]	●	○	○	Multi-channel spectrogram + CNN + LSTM	C	○	●	●	●	–	–	●
	○	●	○	Multi-channel spectrogram + CNN + majority voting	C	○	●	●	●	–	–	●
	○	○	●	Point cloud + 3D cuboid approximation	G	○	–	–	○	1	1	●
M5 [45]	●	○	○	R(2+1)D+GRU (video), CNN+GRU (audio), A34F+RF (audio), Late fusion (averaging)	C	○	●	●	●	1	–	●
	○	●	○	CNN+GRU (audio), A34F+RF (audio), Late fusion (averaging)	C	○	●	●	●	–	–	●
	○	○	●	Energy minimization + 3D cylinder approximation	G	○	–	–	○	2	–	○

KEY – FL: filling level estimation, FT: filling type estimation, CC: container capacity estimation, App: approach, JLT: joint filling type and level classification, A: audio, R: number of used RGB views, D: number of views used with depth data, L: liquids, Gr: granular materials, Temp.: temporal, C: classification, R: regression, G: projective geometry, CNN: convolutional neural network, STFT: short-term Fourier transform, FCNN: fully connected neural network, MLP: multi-layer perceptron, LSTM: Long-Short Term Memory, GRU: Gated Recurrent Unit, kNN: k-Nearest Neighbour, SVM: support vector machine. RF: random forest, A34F: 34 audio features [46] consisting of zero crossing rate, energy, entropy of energy, spectral centroid, spectral spread, spectral entropy, spectral flux, spectral rolloff, Mel-frequency Cepstrum Coefficients (MFFCs), chroma vector and chroma deviation.

summarizing the short-term features from longer windows of the input audio signal and by including additional statistics, such as mean and standard deviation. Learned features are extracted by CNNs from multi-channel or mono-channel audio signals that are post-processed into spectrograms or log-Mel spectrograms [44], [45]. To handle audio signals of different duration in the dataset, long audio signal can be truncated to a pre-defined duration and zero-padding is added to shorter signals [42], [44].

M1 trained a 5-layers fully connected neural network, which takes STFT features as input, with Adam optimizer [47] and dropout [48] on the last hidden layer to reduce overfitting.

M2 [42] addressed only filling type classification and concatenated 40 normalized MFCCs features that are extracted from all audio frames in a window size of 20 ms at 22 kHz, with a maximum length of 30 s. The concatenated features are provided as input to a CNN for directly classifying the filling type (one-to-one sequence classification). The CNN consists of 2 convolutional layers and 1 fully connected layer, (86,876 trainable parameters).

M4 [44] used all the 8 audio signals from the microphone array to compute log Mel-scaled spectrograms with STFT and 64 filter banks for filling type and filling level classification. A sliding window on the cropped spectrogram with 75% overlap forms overlapping audio frames consisting of 3D tensors, where the third dimension is given by the 8 audio channels. Each window is provided as input to a CNN consisting of 5 blocks, each with 2 convolutional and 1 batch normalization layers followed by a max-pooling layer. The CNN is complemented by 3 fully connected layers for the filling type classification of each audio frame and followed by the majority voting. The CNN has a total of 13 layers with 4,472,580 trainable parameters. The same extracted features are also used as input to the three stacked LSTMs for the filling level classification. The three stacked LSTMs are trained with a set of 100 audio frames and contain 256 hidden states, resulting in 2,366,211 trainable parameters.

M3 [43] trained multi-layer perceptrons (MLPs) that are specific only to each object category (*cup*, *drinking glass*, *food box*) and for either filling level or filling type classification. Each MLP has 3 layers with 3,096 nodes in the first hidden layer and 512 in the last hidden layer. The total number of trainable parameters is 20,762,288. The MLPs takes as input a spectrogram computed from a multi-channel sound signal re-sampled at 16,600 Hz and converted into mono-channel by averaging the samples across channels. Only the last 32,000 samples are retained and converted into a spectrogram as audio feature via Discrete Fourier Transform. To select which MLP to use at inference time, regions of interest (ROIs) are detected in all frames of the image sequences of all four views in our dataset by using YOLOv4 [49] pre-trained on MS COCO [50]. The class category (*cup*, *drinking glass*, *food box*) is determined by a majority voting of randomly sampled frames (65% of all frames).

M5 [45] used both traditional and learned acoustic features for filling type classification, whereas visual features are extracted in addition to the acoustic features for filling level classification. Multiple classifiers, each associated with each feature, are used to output the class probabilities and the average across the classifiers is computed to determine the final class for either filling level or filling type. For the acoustic features, the multi-channel input audio signal is converted into a mono-channel by averaging the samples across channels. MFCCs, energy, spectral characteristics, and their statistics from 50 ms windows of the input signal are computed as short-term traditional features. The features are concatenated in a 136-dimensional vector used as input to a

RF classifier. The number of trees of the RF classifier is automatically set during training by selecting the value between $(10, 25, 50, 100, 200, 500)$ that achieves the highest accuracy in validation. For the learned features, the mono-channel signal is re-sampled at 16 kHz and converted into log-Mel spectrograms from 960 ms windows of the re-sampled signal. Each spectrogram is provided as input to a VGG-based model [51] that is pre-trained on a large dataset (e.g., AudioSet [52]) and computes a feature vector of dimension 128. The learned features are then provided as input to a GRU model [53] that has 5 layers and a hidden layer of size 512 to handle the intrinsic temporal relations of the signals. The model has a total of 7,291,395 trainable parameters. Visual features are extracted from the image sequences of all camera views by using R(2+1)D [54], a spatio-temporal CNN that is based on residual connections [41] and 18 (2+1)D convolutional layers that approximate 3D convolution by a 2D convolution (spatial) followed by a 1D convolution (temporal). R(2+1)D is pre-trained for action recognition on Kinetics 400 [55], takes as input a fixed window of 16 RGB frames of 112×112 pixel resolution, and outputs a 512-dimensional feature vector. Long temporal relation between the features of each window are estimated by using a RNN with a GRU model that has 3 layers and a hidden dimension of size 512 (4,729,347 trainable parameters). The GRU models from each camera view are jointly trained and their logits are summed together before applying the final softmax to obtain the class probabilities from the visual input. For filling type classification, the probabilities resulting from the last hidden state of the GRU network and those resulting from the RF are averaged. For filling level classification, the probabilities resulting from the RF classifier and the GRU models for both the audio and visual features are averaged together to compute the final class. The RF classifier and all the GRU models are trained independently for filling type classification and filling level classification by using 3-fold validation strategy.

B. JOINT FILLING TYPE AND LEVEL CLASSIFICATION

Jointly estimating the filling type and level can avoid infeasible cases, such as an *empty water* or *half-full none*. Different traditional classifiers and existing CNNs that use spectrograms as input have been analyzed and compared in [40], especially when different containers are manipulated by a person with different content types, such as both liquids and granular materials.

Because of the different container types and corresponding manipulation, M6 [40] decomposed the problem into two steps, namely action recognition and content classification and devised three independent CNNs. The first CNN (action classifier) identifies the manipulation performed by the human, i.e., shaking or pouring, and the other two CNNs are task-specific and determine the filling type and level. The CNN for action recognition (*pouring*, *shaking*, *unknown*) has 4 convolutional, 2 max-pooling, and 3 fully connected layers; the CNN for the specific action of pouring has 6 convolutional, 3 max-pooling, and 3 fully connected layers; and the CNN for the specific action of shaking has 4 convolutional, 2 max-pooling, and 2 fully connected layers. The choice of which task-specific network should be used is conditioned by the decision of the first CNN. When the action classifier does not distinguish between pouring or shaking, the approach associates the *unknown* case to the class *empty*.

C. CAPACITY ESTIMATION

We categorize the methods in regression [42], [43] and geometric-based approaches [44], [45]. These methods use either RGB, RGB and depth data, or multiple RGB images from our dataset.

Regression approaches use either deep CNNs [42] or distribution fitting via Gaussian processes [43]. M2 [42] trained a CNN architecture consisting of 4 convolutional layers, each followed by batch normalization [56], and 3 fully connected layers (532,175 trainable parameters). The CNN takes as input a ROI and its normalized relative size, and then regresses the capacity of the container limited to 4000 mL, accordingly to the range of capacities in our dataset. The ROI is computed from the contour features of a depth image selected from the frame with the most visible pixels of the frontal, fixed view and assuming a maximum depth of 700 mm. M4 [43] regressed the container capacity using Gaussian processes depending on the container category. To model multiple multi-variate Gaussian functions for each container type, the container type is recognized by detecting multiple ROIs in all frames of all image sequences as done for filling type and level classification.

Geometric-based approaches approximate the container to a primitive shape in 3D, such as cuboid or cylinder [8], [44], [45]. The shape is represented as a point cloud obtained directly from RGB-D data or computed via energy-based minimization to fit the points to the real shape of the object as observed in the RGB images of a wide-baseline stereo camera and constrained by the object masks [8], [45]. The capacity is then computed as a by-product, e.g., by finding the minimum and maximum values for each coordinate in 3D [44] or using volume formulas specific for the primitive shape [45]. The approximated primitives can lead to inaccurate capacities: a cuboid representation could result in an overestimated capacity and hence re-scaling would be necessary [44]; a cylinder representation may not generalize to different shapes than rotationally symmetric objects. To handle occlusions caused by the human hand manipulating a container, [44] selects the RGB-D frame with a single silhouette having the largest number of pixels and post-processes the point cloud to deal with inaccuracies in the segmentation. [45] averages the capacity estimations computed at different frames of the image sequences in the stereo views, as the approach assumes the container to be fully visible.

IV. COMPARATIVE ANALYSIS

We compare and analyze the performance of the 6 state-of-the-art methods and the 13 baselines on the public test set,

the private test set, and their combination of the CORSMAL Containers Manipulation dataset [31].

A. IMPLEMENTATION DETAILS

M2 trained the CNN for filling type classification with SGD optimizer, a fixed learning rate of 2.5×10^{-4} and momentum of 0.9, and a batch size of 16. M4 sets the frame length to 25 ms, the hop-length of 10 ms, and the number of samples for the Fast Fourier Transform to 512 for computing the STFT. During training, M4 crops audio signals based on manual annotations of the starting and ending of the manipulation. The network for filling level classification is trained by using cross-entropy loss and ADAM optimizer [47] with a learning rate of 0.1×10^{-4} and a mini-batch size of 32 for 200 epochs.

B. RESULTS FOR FILLING LEVEL CLASSIFICATION

Table 2 compares the performance of all baselines and methods except M2 that did not address the task. M4, M5 and M6 achieve the highest accuracy with 80.84, 79.65, and 78.65 $\bar{F}_1$ on the combined test set, respectively. This performance is almost twice higher than M1 and M3 and shows that using only audio as input modality is sufficient to achieve an accuracy higher than 75 $\bar{F}_1$. M5 is also using visual information from the RGB sequence of all the four camera views, but the similar performance to M4 and M6 suggests that audio features are dominant in determining the classification decision. M6 is the best performing in the private test set (81.46 $\bar{F}_1$), whereas M4 is the best performing in the public test set (82.63 $\bar{F}_1$). Interestingly, both methods selected a fixed portion of the audio signal, transformed into a spectrogram, where the manipulation of the container by the human subject was more likely to occur (see Fig. 3). However, M6 chose to provide the full trimmed spectrograms as input to the three CNNs, whereas M4 adopted a temporal approach with a sliding window to provide portions of the log-Mel spectrogram to a CNN and an LSTM. Both are shown to be valid methods assuming that the whole audio signal is available and the manipulation is completed.

The confusion matrices in Fig. 4 show that M4 and M6 do not confuse the class *empty*, whereas M5 mis-classifies some *empty* configurations as *half-full*. Not surprisingly, most of the confusions occur between the classes *half-full* and *full* for all methods. M4 and M5 are more accurate than M6 in recognizing the class *half-full*, but M6 is more accurate in recognizing the class *full*. M3 mis-classifies the true class *empty* as *half-full* for 40% of the times and as *full* for 33% of the times, and the class *full* is confused with *half-full* for 75% of the times. M3 recognizes the container categories *cup*, *drinking glass* and *food box* with 92%, 73%, and 88% accuracy, respectively, in the training set. Errors in the category recognition may lead to wrong classifications by the selected category-specific MLP-based classifier, which is also trained with limited and selected data. The CNN of M1 made erroneous predictions across all classes, except for *empty* that was never predicted as *half-full* but only confused with *full*.

TABLE 2. Filling level classification results (Task 1). Baselines and state-of-the-art methods (MX with X ranges from 1 to 6) are ranked by their score in the combined test set.

Method	Input modality					Test set		
	A	R1	R2	R3	R4	Public	Private	Combined
Mask + RN	○	○	○	○	●	25.12	21.99	23.68
Mask + RN	○	○	○	●	○	36.52	25.52	31.46
Spect. + PCA + SVM	●	○	○	○	○	30.08	31.99	31.64
Random	–	–	–	–	–	33.35	41.86	37.62
Spect. + PCA + kNN	●	○	○	○	○	39.03	37.16	38.31
Mask + RN	○	○	●	○	○	48.90	26.73	39.00
M3 [43]	●	●	●	●	●	44.31	42.70	43.53
Spect. + PCA + RF	●	○	○	○	○	46.79	42.46	44.66
Spect. + RF	●	○	○	○	○	45.43	45.59	45.49
Mask + RN	○	●	○	○	○	58.51	32.93	47.00
M1	●	○	○	○	○	50.73	47.08	48.71
Spect. + SVM	●	○	○	○	○	47.66	51.54	49.67
AF193 + kNN	●	○	○	○	○	55.49	53.22	54.47
Spect. + kNN	●	○	○	○	○	59.15	53.47	56.38
ZCR + MFCCs + kNN	●	○	○	○	○	63.63	54.97	59.35
AF193 + SVM	●	○	○	○	○	60.77	58.57	60.09
ZCR + MFCCs + SVM	●	○	○	○	○	66.27	57.19	61.87
AF193 + RF	●	○	○	○	○	64.18	63.94	64.74
ZCR + MFCCs + RF	●	○	○	○	○	70.04	63.11	66.80
M4 [44]	●	○	○	○	○	**82.63**	74.43	78.56
M5 [45]	●	●	●	●	●	78.14	81.16	79.65
M6 [40]	●	○	○	○	○	80.22	**81.46**	**80.84**

Best performing method highlighted in bold.
KEY – A: audio, RX: RGB for view X (1,2,3,4), Mask + RN: Mask R-CNN + ResNet-18, ZCR: zero crossing rate, MFCCs: Mel-frequency cepstrum coefficients, Spect.: spectrogram, RF: random forest, SVM: support vector machine, kNN: k-nearest neighbor, PCA: principal component analysis, AF193: 193 audio features consisting of MFCCs, chromogram, Mel-scaled spectrogram, spectral contrast, and tonal centroid.

The vision-only baseline (using the first camera view, on the left side of the robot arm) confused 81% of the times the class *empty* with *half-full* in addition to mis-classification between *half-full* and *full*, making the performance of the baseline only 10 $\bar{F}_1$ points higher than a random classifier (37.62 $\bar{F}_1$).

C. RESULTS FOR FILLING TYPE CLASSIFICATION

Table 3 shows that M4, M6, and M5 are the best performing with 96.95 $\bar{F}_1$, 94.50 $\bar{F}_1$, 94.26 $\bar{F}_1$ scores on the combined test set (as for filling level classification). Audio is the preferred modality by all the methods except M3 that conditions the selection of the audio-based classifier to the recognition of the container category from visual data. As for filling level classification (43.53 $\bar{F}_1$), selecting which classifier to use is likely to be the main source of error for the classifications of M3 (41.83 $\bar{F}_1$), whereas using only audio is sufficient to achieve performance close to 100 $\bar{F}_1$ score. If the audio modality was not available, both filling level and filling type classifications would be very challenging using only visual data. M1 and M2 achieve 75.24 $\bar{F}_1$ and 86.89 $\bar{F}_1$, respectively, but lower of about 20 and 10 percentage points (pp), respectively, than M4. The table also shows that the performance of the baselines varies from random results to almost the same performance as the best performing M4. Using the spectrogram as an input feature (either after reshaping the spectrogram into a vector or after applying PCA to select the first 128 components) to any of the three classifiers, namely kNN, SVM, or RF, is the worst choice.

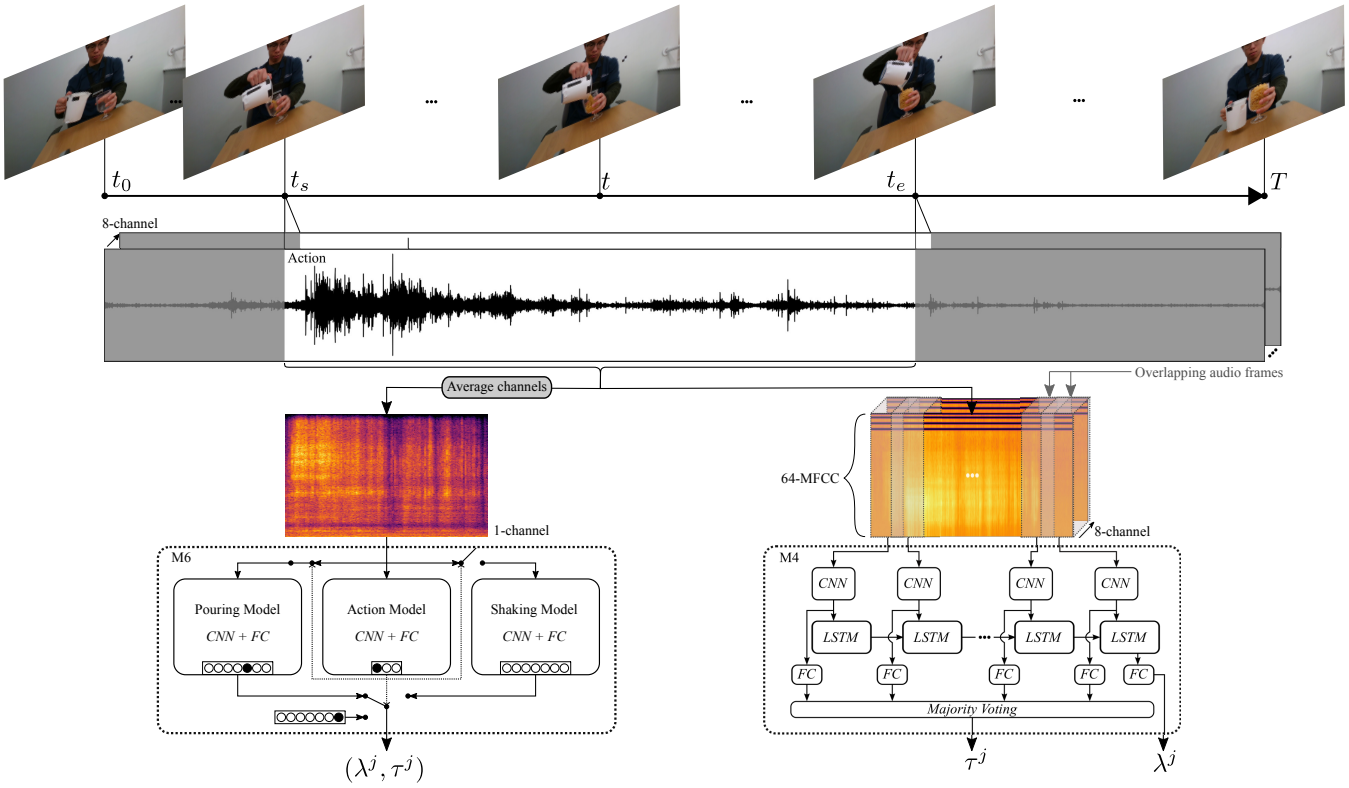

FIGURE 3. Illustrative comparison of M6 (left) and M4 (right) for filling type and level classification.

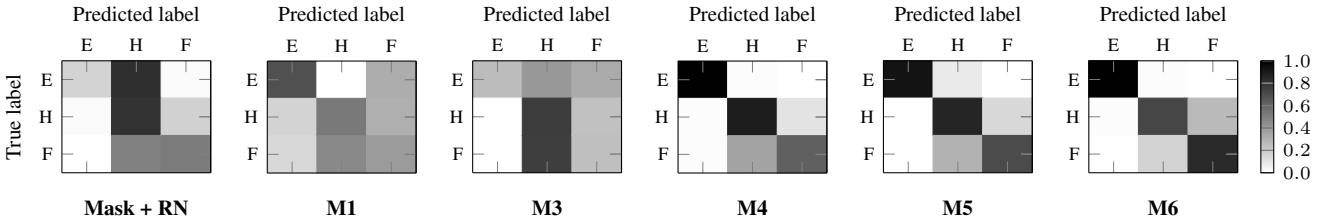

FIGURE 4. Confusion matrices of filling level classification for all methods across all the containers of the public and private testing splits of the CORSMAL Container Manipulation dataset [31]. Note that the counting for each cell is normalized by the total number of true labels for each class (colorbar). KEY –E: empty; H: half-full; F: full, Mask + RN: Mask R-CNN + ResNet-18.

On the combined test set, the lowest performance is obtained by Spectrogram + PCA + SVM with 24.20 $\bar{F}_1$, whereas the highest performance is obtained by Spectrogram + kNN with 64.55 $\bar{F}_1$. Classic audio features, such as MFCCs and ZCR, are more discriminative and sufficient to achieve performance higher than 78 $\bar{F}_1$ for the three classifiers. Simply using ZCR and MFCCs with RF can achieve 91.31 $\bar{F}_1$, which is close to the performance of the three top methods (M5, M6, M4) that are using CNNs and LSTMs. On the contrary, performance decreases when using a larger set of features, such as tonal centroid, spectral contrast, chromogram, Mel-scaled spectrogram, and MFCCs.

Fig. 5 shows the confusion matrices of the methods. M4 made a few mis-classifications for the class *rice* with *none* and *pasta*, and for the class *water* with *none*. M6 confused 4% *pasta* with *rice*, 4% *rice* with *pasta*, 7% *pasta* with *water*, and 2% *water* with *none*. The confusion between *water* and *none* could be expected due to the low volume of the sound produced by the water, whereas the confusion of *water* with *rice* might be caused by the glass material of the container and noise background. The largest confusion for M5 is given by the erroneous prediction of *rice* with *pasta* (13%). As for filling level classification, M1 and M3 have large mis-classifications across different classes, with M3 that could not predict *water* for any audio input.

D. RESULTS FOR CAPACITY ESTIMATION

We compare the results of M2, M3, M4, and M5, in terms of the average capacity score. We also report the results of a pseudo-random generator (Random) that draws the predictions from a uniform distribution in the interval $[50, 4000]$ based on the Mersenne Twister algorithm [57]. We then analyze and discuss the statistics of the absolute error in predicting the container capacity for each testing container as well as for each filling type and level.

Table 4 shows that M2 achieves the best score with

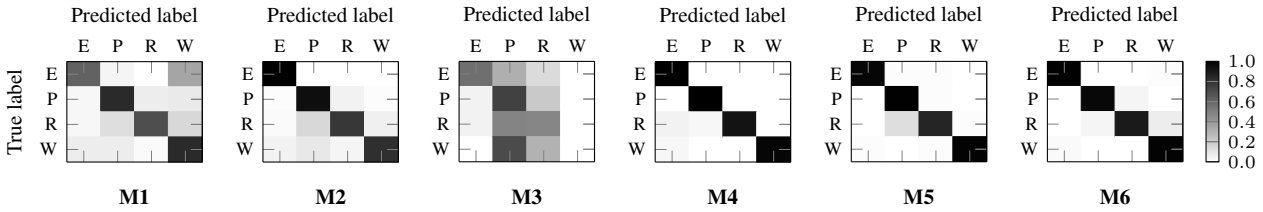

FIGURE 5. Confusion matrices of filling type classification for all methods across all the containers of the public and private testing splits of the CORSMAL Containers Manipulation dataset [31]. Note that each cell is normalized by the total number of true labels for each class (colorbar). KEY – E: empty; P: pasta; R: rice; W: water.

TABLE 3. Filling type classification results (Task 2). Baselines and state-of-the-art methods (MX with X ranges from 1 to 6) are ranked by their score in the combined test set.

Method	Input modality					Test set		
	A	R1	R2	R3	R4	Public	Private	Combined
Mask + RN	○	○	○	○	●	14.12	11.23	12.70
Mask + RN	○	○	○	●	○	21.14	9.04	15.63
Mask + RN	○	○	●	○	○	28.75	15.54	22.90
Mask + RN	○	●	○	○	○	30.85	13.04	23.05
Spect. + PCA + SVM	●	○	○	○	○	20.57	27.60	24.20
Random	–	–	–	–	–	21.24	27.52	24.38
Spect. + PCA + kNN	●	○	○	○	○	24.47	28.34	26.53
Spect. + PCA + RF	●	○	○	○	○	28.75	37.79	33.32
Spect. + SVM	●	○	○	○	○	39.39	41.81	40.61
M3 [43]	●	●	●	●	●	41.77	41.90	41.83
Spect. + RF	●	○	○	○	○	47.98	47.68	47.82
Spect. + kNN	●	○	○	○	○	60.50	68.58	64.55
AF193 + SVM	●	○	○	○	○	64.92	79.72	72.86
M1	●	○	○	○	○	78.58	71.75	75.24
AF193 + kNN	●	○	○	○	○	76.84	75.96	76.41
ZCR + MFCCs + SVM	●	○	○	○	○	84.23	71.96	78.67
ZCR + MFCCs + kNN	●	○	○	○	○	88.19	79.23	83.73
M2 [42]	●	○	○	○	○	81.97	91.67	86.89
AF193 + RF	●	○	○	○	○	88.36	87.46	87.88
ZCR + MFCCs + RF	●	○	○	○	○	92.97	89.74	91.31
M5 [45]	●	○	○	○	○	93.83	94.70	94.26
M6 [40]	●	○	○	○	○	95.12	93.92	94.50
M4 [44]	●	○	○	○	○	**97.83**	**96.08**	**96.95**

Best performing method highlighted in bold.
KEY – A: audio, RX: RGB for view X (1,2,3,4), Mask + RN: Mask R-CNN + ResNet-18, ZCR: zero crossing rate, MFCCs: Mel-frequency cepstrum coefficients, Spect.: spectrogram, RF: random forest, SVM: support vector machine, kNN: k-nearest neighbor, PCA: principal component analysis, AF193: 193 audio features consisting of MFCCs, chromogram, Mel-scaled spectrogram, spectral contrast, and tonal centroid.

66.92 $\bar{C}$, 67.67 $\bar{C}$, and 67.30 $\bar{C}$ for the public test set, private test set, and the combined test set, respectively, when using only depth data from the fixed frontal view. All methods achieve a performance score that is twice higher than the random solution (24.58 $\bar{C}$ for the combined test set): M4 has the lowest score (54.79 $\bar{C}$), whereas M5 and M3 obtain 60.57 $\bar{C}$ and 62.57 $\bar{C}$, respectively. Fig. 6 shows the statistics (minimum, maximum, median, 25$^{\text{th}}$ and 75$^{\text{th}}$ percentiles) of the relative absolute errors for each container in the test sets of the dataset. M2 has the lowest median error for all containers, except for the private containers C14 and C15. The variation of the error across configurations is either smaller than the variation of the other methods or lower than the median value of the other methods. M5 is more consistent in estimating the same container shape and capacity for most of the configurations related to containers C12 and C15. M5 also have the largest variations for C10 and C14; M3 for C12 and C15; and M4 for C11. Interestingly, M3 have a median error lower than M4 and M5 for C13 and achieve the lowest median error with a small variation across configurations for C14. However, we can observe that in general the relative absolute error across containers is around or higher than 0.5.

TABLE 4. Container capacity estimation results (Task 3). Methods ranked by the average capacity score on the combined test set.

Method	Input modality								Test set		
	R1	D1	R2	D2	R3	D3	R4	D4	Public	Private	Combined
Random	–	–	–	–	–	–	–	–	31.63	17.53	24.58
M4 [44]	●	●	○	○	○	○	○	○	57.19	52.38	54.79
M5 [45]	●	○	●	○	○	○	○	○	60.56	60.58	60.57
M3 [43]	●	○	●	○	●	○	●	○	63.00	62.14	62.57
M2 [42]	○	○	○	○	○	●	○	○	**66.92**	**67.67**	**67.30**

Best performing method highlighted in bold.
KEY – RGB (R) or depth (D) modality for view X (1,2,3,4)

In addition to the comparison across containers, Fig. 7 shows the relative absolute errors grouped by filling type and level for each method. We can observe that most of the errors are in the interval [0.3,0.8], and the methods have similar amount of variations between the 25$^{\text{th}}$ and 75$^{\text{th}}$ percentiles, but differences are in the median error and the maximum error (excluding outliers). M2 achieves the lowest median error (always lower than half of the real container capacity) and smaller variations (25$^{\text{th}}$-75$^{\text{th}}$ percentiles), whereas M3 have similar results for *rice full*. M4 has the largest errors for *empty*, *pasta half-full*, *pasta full*, *rice half-full*, and *rice full*. M5 has the largest errors for *water half full* and *water full*.

E. RESULTS PER SCENARIO AND PER CONTAINER

Table 5 analyzes and compares the performance scores of the methods grouped by scenario and containers for all the three tasks. M4, M5, and M6 increase their $\bar{F}_1$ for filling level classification on the testing containers from scenario 1 to scenario 3, showing how audio information is robust despite the increasing difficulty due to the in-hand manipulation (scenario 2 and 3) and larger distance (scenario 3). However, M6 decreases by almost 2 pp from scenario 1 (78.52 $\bar{F}_1$) to scenario 2 (76.92 $\bar{F}_1$). M1 is affected by the in-hand manipulation and distance, decreasing from 52.90 $\bar{F}_1$ in scenario 1 to 45.46 $\bar{F}_1$ in scenario 3. M3 achieve the highest accuracy for scenario 2 (51.34 $\bar{F}_1$), increasing by 11.51 pp compared to scenario 1 (39.83 $\bar{F}_1$), but decreasing to 35.92 $\bar{F}_1$ in scenario 3 (likely caused by the errors in recognizing the container

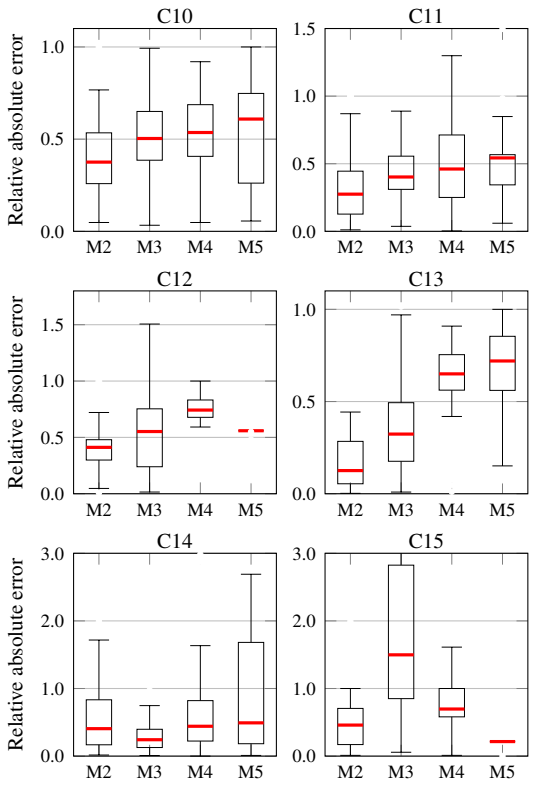

FIGURE 6. Comparison of statistics of the absolute error in estimating the container capacity for each testing container between M2 [42], M3 [43], M4 [44], and M5 [45]. Note that outliers in the data are not shown. Note also the different scale for the y-axis. KEY – CX: container (C) index (X), where X is in the range [10,15].

category). For filling type classification, the performance of M4, M5, and M6 is higher than 90 $\bar{F}_1$ across the scenarios, but the trend is the opposite of filling level classification. M5 and M6 decrease in $\bar{F}_1$ from scenario 1 to scenario 3, whereas M4 achieves the highest accuracy in scenario 2 (98.07 $\bar{F}_1$). M3 and M1 show the same behavior for filling level and type classification with a large decrease in scenario 3 by 15.31 pp and 22.16 pp compared to scenario 1, respectively. For capacity estimation, M3 and M4 are less affected by the variations across the scenarios, whereas M2 is the best performing in scenario 1 (68.81 $\bar{C}$) and scenario 2 (73.70 $\bar{C}$) but decreases by 9.42 pp in scenario 3 compared to scenario 1. M2 is based only on the frontal depth view, where the subject is not visible for most of the time. This challenges the method to detect the object in the pre-defined depth range. M5 is affected by the increasing challenges across scenarios, decreasing from 66.51 $\bar{C}$ in scenario 1 to 55.68 $\bar{C}$ in scenario 3. This shows the limitations of the underline approach [8] that was designed for objects free of occlusions and standing upright on a surface.

The performance across containers varies between the methods. Testing containers 12 and 15 are the most challenging for M3, M4, M5, M6, when classifying the filling level, whereas M1 achieves its best performance on both containers. M4 and M5 have the largest decrease with the score in the interval [40,50] $\bar{F}_1$ compared to the interval [75-93] $\bar{F}_1$ for the other containers. M6 outperforms all the other methods with 64.98 $\bar{F}_1$ and 74.99 $\bar{F}_1$ for containers 12 and 15. For filling type classification, M3 obtains 86.98 $\bar{F}_1$ and 79.45 $\bar{F}_1$ for containers 12 and 15, respectively, and less than 30 $\bar{F}_1$ on the other containers. Because of the dataset structure, M3 can recognize the *box* class and the filling type for that class, but the method cannot easily distinguish filling type and level for drinking glasses and cups. Overall, other methods achieve a score higher than 70 $\bar{F}_1$ across containers. M4 achieves 100 $\bar{F}_1$ on container 13 and M6 on container 15. M4 is the best performing for containers 10 and 11, whereas M5 is the best for container 14. Containers 12 and 15 are the most challenging for M5; container 14 for M6; container 15 for M4; containers 10, 11, and 15 for M2. M1 ranges between 55.60 $\bar{F}_1$ and 87.38 $\bar{F}_1$ across containers, with the drinking glasses being the most challenging and obtaining 69.33 $\bar{F}_1$ for container 11 and 55.60 $\bar{F}_1$ for container 14. For capacity estimation, M2 achieves the best performance on containers 10 (66.02 $\bar{C}$), 11 (69.14 $\bar{C}$), 12 (65.02 $\bar{C}$), and 13 (79.75 $\bar{C}$), M3 on container 14 (76.75 $\bar{C}$), and M5 on container 15 (78.82 $\bar{C}$). M3 achieves higher average capacity score on the private cup and drinking glass than the public containers, but the score drops to 28.02 $\bar{C}$ for the container 15. M4 performs worse on the private testing containers than the public testing containers, with the lowest scores on the boxes (containers 12 and 15). M5 also performs worse for the drinking glass and cups in the private test set than the public test set. Surprisingly, the best score of M5 is on the *box* container 15 (78.82 $\bar{C}$) despite the modeled shape is a 3D cylinder.

TABLE 5. Comparison of the task performance scores between methods for each scenario and for each testing container.

	Method	S1	S2	S3	C10	C11	C12	C13	C14	C15
T1	M1	52.90	47.37	45.46	48.69	46.78	58.27	41.43	38.89	62.58
	M3 [43]	39.83	51.34	35.92	35.24	36.59	22.86	33.74	33.74	26.33
	M4 [44]	75.41	77.70	82.54	92.85	89.25	46.67	86.85	74.16	45.92
	M5 [45]	75.87	80.89	82.03	83.12	90.48	41.26	88.09	90.36	47.68
	M6 [40]	78.52	76.92	86.84	83.12	88.10	64.98	89.16	78.28	74.99
T2	M1	81.22	77.01	66.06	86.42	69.33	79.67	87.38	55.60	72.77
	M2 [42]	90.68	84.57	85.41	77.29	80.60	91.58	94.02	94.09	85.03
	M3 [43]	44.06	51.21	28.75	21.72	26.54	86.98	20.33	34.13	79.45
	M4 [44]	97.35	98.07	95.45	97.63	98.82	96.72	100.00	97.66	87.96
	M5 [45]	96.70	94.76	91.32	96.44	97.62	84.81	97.63	98.81	84.45
	M6 [40]	96.70	95.43	91.27	91.58	96.41	98.33	97.62	85.61	100.00
T3	M2 [42]	68.81	73.70	59.39	66.02	69.14	65.08	79.75	61.12	59.94
	M3 [43]	64.33	60.41	62.96	60.99	66.21	61.30	71.90	76.75	28.02
	M4 [44]	55.45	55.34	53.57	59.62	61.70	47.47	53.77	58.29	42.17
	M5 [45]	66.51	59.51	55.68	60.71	62.43	57.71	53.37	54.75	78.82

KEY – S: scenario, C: container, T: task.

F. RESULTS ON THE OVERALL FILLING MASS

We discuss the overall performance of the methods based on their results on estimating the filling mass. Methods that estimated either of the physical properties in our framework (e.g., M1, M2, and M6) are complemented by the random estimation of the missing physical properties to compute the

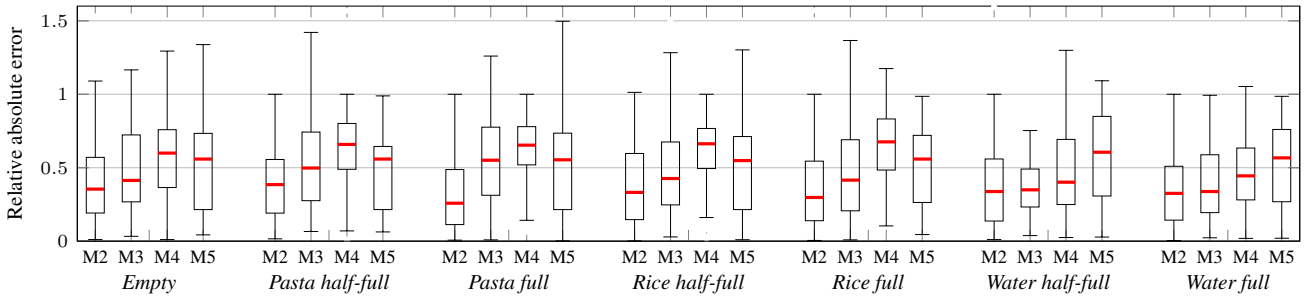

FIGURE 7. Comparison of the absolute error in estimating the container capacity between M2 [42], M3 [43], M4 [44], and M5 [45] for the different combinations of filling type and level in the combined public and private test set of the CORSMAL Containers Manipulation dataset. Statistics of the box plot includes the median (red line), the 25th and 75th percentiles, and the minimum and maximum error..

TABLE 6. Comparison of the filling mass estimation results. Methods are ranked by their score on the combined test sets of the CORSMAL Containers Manipulation dataset. Note that scores are weighed by the number of tasks addressed by the methods.

Method	Task			Test set		
	T1	T2	T3	Public	Private	Combined
M6 [40]	●	●	○	28.25	21.89	25.07
M1	●	●	○	29.25	23.21	26.23
Random	●	●	●	38.47	31.65	35.06
M2 [42]	○	●	●	38.56	39.80	39.18
M3 [43]	●	●	●	52.80	51.14	53.47
M4 [44]	●	●	●	63.32	61.01	62.16
M5 [45]	●	●	●	**64.98**	**65.15**	**65.06**

filling mass[4]. Table 6 shows that methods addressing only filling type and level classification achieve a lower score than a random guess for each task. Given the multiplicative formula of the filling mass estimation (see Eq. 6), even a few errors in these classification tasks can lead to a low score in the filling mass estimation, especially when combined with the random estimation of the container capacity. However, improving the capacity estimation is an important aspect to achieve more accurate results (and higher score) for the filling mass estimation (see M2). M3, M4, and M5 addressed all three tasks and achieved 53.47 $\bar{M}$, 62.16 $\bar{M}$, and 65.06 $\bar{M}$, respectively. Overall, methods perform better on the public test set than the private test set, except for M2 and M5 that achieve similar performance in the two test sets. We can observe that the more accurate predictions in the container capacity help M3 to obtain 53.47 $\bar{M}$ despite the classification errors for filling level and type. The high classification accuracy on filling level and type, combined with a similar score for the capacity estimations with respect to M3, makes M4 and M5 the best performing in filling mass estimation. The similar scores for container capacity and filling mass estimation shows how important it is to accurately predict the capacity in order to correctly estimate the filling mass.

V. CONCLUSION

We presented the open CORSMAL framework to benchmark methods for estimating the physical properties of different containers while they are manipulated by a person with different content types. The framework includes a dataset, a set of tasks and performance measures, and several baselines that use either audio or visual input. The framework supports the contactless estimation of the weight of the container, including its content (if any), despite variations in the physical properties across containers and occlusions caused by the hand manipulation. We performed an in-depth comparative analysis of the baselines and state-of-the-art methods that used the framework. The analysis showed that using only audio as input is sufficient to achieve a weighted average F1-score above 80% for filling type and level classification, but the good performance could be limited to the sensor types and setup of our dataset. Methods that use audio alone are robust to changes in the container type, size, and shape, as well as pose during the manipulation. Moreover, filling type and level estimation can benefit from each other to avoid unfeasible solutions [40]. Container capacity is the most challenging physical property to estimate with all methods affected by large errors and a maximum score of 65%. Performance on this task also affects the successive estimation of the filling mass. The design of a method that can generalize across the different containers and scenarios, especially for container capacity estimation and partially for filling level classification, is still challenging. Future directions involve the exploration of fusion and learning methods with both acoustic and visual modalities to support the contactless estimation of the physical properties of containers and their content.

[4]Note that for the organized challenge, the score is weighted by the number of completed tasks and hence results here are reported in the same manner.

ACKNOWLEDGMENT

We would like to thank Ricardo Sanchez-Matilla and Riccardo Mazzon for their contribution in the design and collection of the data, and the definition of the performance measures.

REFERENCES

[1] 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, Apr. 2020.

[2] J. R. Medina, F. Duvallet, M. Karnam, and A. Billard, "A human-inspired controller for fluid human-robot handovers," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, Cancun, Mexico, 15–17 Nov. 2016.

[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, Jan. 2021.

[4] V. Ortenzi, A. Cosgun, T. Pardi, W. P. Chan, E. Croft, and D. Kulić, "Object handovers: A review for robotics," *IEEE Trans. Robotics*, pp. 1–19, 2021.

[5] W. Yang, C. Paxton, A. Mousavian, Y.-W. Chao, M. Cakmak, and D. Fox, "Reactive human-to-robot handovers of arbitrary objects," in *IEEE Int. Conf. Robotics Autom.*, 2021.

[6] H. Liang, C. Zhou, S. Li, X. Ma, N. Hendrich, T. Gerkmann, F.-C. Sun, and J. Zhang, "Robust robotic pouring using audition and haptics," in *IEEE Int. Conf. Intell. Robot Syst.*, Las Vegas, NV, USA, 24 Oct. 2020–24 Jan. 2021.

[7] 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.*, Anchorage, Alaska, USA, 19–22 Sep. 2021.

[8] 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 Signal Process.*, Barcelona, Spain, 4-8 May 2020.

[9] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *IEEE Int. Conf. Comput. Vis.*, Venice, Italy, 22–29 Oct. 2017.

[10] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized object coordinate space for category-level 6d object pose and size estimation," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Long Beach, CA, USA, 16–20 Jun. 2019.

[11] H. Yang and L. Carlone, "In perfect shape: certifiably optimal 3d shape reconstruction from 2d landmarks," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Seattle, Washington, USA, 16-18 Jun. 2020.

[12] A. Ahmadyan, L. Zhang, J. Wei, A. Ablavatski, and M. Grundmann, "Objectron: A large scale dataset of object-centric videos in the wild with pose annotations," *arXiv preprint arXiv:2012.09988*, 2020.

[13] X. Chen, Z. Dong, J. Song, A. Geiger, and O. Hilliges, "Category level object pose estimation via neural analysis-by-synthesis," *arXiv preprint arXiv:2008.08145*, 2020.

[14] D. Chen, J. Li, Z. Wang, and K. Xu, "Learning canonical shape space for category-level 6D object pose and size estimation," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Virtual, 14–19 Jun. 2020.

[15] T. Hodaň, F. Michel, E. Brachmann, W. Kehl, A. Glent Buch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis, C. Sahin, F. Manhardt, F. Tombari, T.-K. Kim, J. Matas, and C. Rother, "BOP: Benchmark for 6D object pose estimation," in *Eur. Conf. Comput. Vis.*, Munich, Germany, 8–14 Sep. 2018.

[16] R. Kaskman, S. Zakharov, I. Shugurov, and S. Ilic, "Homebreweddb: RGB-D dataset for 6D pose estimation of 3D objects," in *IEEE Int. Conf. Comput. Vis. Workshops*, Seoul, Korea, 27 Oct./2 Nov. 2019.

[17] X. Liu, R. Jonschkowski, A. Angelova, and K. Konolige, "Keypose: Multi-view 3d labeling and keypoint estimation for transparent objects," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Virtual, 14–19 Jun. 2020.

[18] S. Sajjan, M. Moore, M. Pan, G. Nagaraja, J. Lee, A. Zeng, and S. Song, "ClearGrasp: 3D shape estimation of transparent objects for manipulation," in *IEEE Int. Conf. Robotics Autom.*, 2020.

[19] C. J. Philips, M. Lecce, and K. Daniilidis, "Seeing glassware: from edge detection to pose estimation and shape recovery," in *Robotics: Science and Syst.*, Ann Arbor, Michigan, USA, 18-22 Jun. 2016.

[20] R. Mottaghi, C. Schenck, D. Fox, and A. Farhadi, "See the glass half full: Reasoning about liquid containers, their volume and content," in *IEEE Int. Conf. Comput. Vis.*, Venice, Italy, 22–29 Oct. 2017.

[21] C. Schenck and D. Fox, "Reasoning about liquids via closed-loop simulation," in *Robotics: Science and Syst.*, Cambridge, Massachusetts, USA, 12–16 Jul. 2017.

[22] C. Do, T. Schubert, and W. Burgard, "A probabilistic approach to liquid level detection in cups using an RGB-D camera," in *IEEE Int. Conf. Intell. Robot Syst.*, Daejeon, Korea, 9–14 Oct. 2016.

[23] C. Do and W. Burgard, "Accurate pouring with an autonomous robot using an RGB-D camera," in *Int. Conf. Intell. Auton. Syst.*, Baden-Baden, Germany, 12–16 Jul. 2018.

[24] S. Griffith, V. Sukhoy, T. Wegter, and A. Stoytchev, "Object categorization in the sink : Learning behavior – grounded object categories with water," in *IEEE Int. Conf. Robotics Autom.*, Minneapolis, MN, USA, 14–18 May 2012.

[25] S. Ikeno, R. Watanabe, R. Okazaki, T. Hachisu, M. Sato, and H. Kajimoto, "Change in the amount poured as a result of vibration when pouring a liquid," in *Int. AsiaHaptics Conf.*, Tsukuba, Japan, 18–20 Nov. 2014.

[26] S. Clarke, T. Rhodes, C. Atkeson, and O. Kroemer, "Learning audio feedback for estimating amount and flow of granular material," in *Proc. Conf. Robot Learn.*, Zürich, Switzerland, 29–31 Oct. 2018.

[27] H. Liang, S. Li, X. Ma, N. Hendrich, T. Gerkmann, F. Sun, and J. Zhang, "Making sense of audio vibration for liquid height estimation in robotic pouring," in *IEEE Int. Conf. Intell. Robot Syst.*, Macau, China, Nov. 2019.

[28] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101 – mining discriminative components with random forests," in *Eur. Conf. Comput. Vis.*, Zurich, Switzerland, 16–12 Sep. 2014.

[29] Y. Kawano and K. Yanai, "FoodCam-256: A large-scale real-time mobile food recognition system employing high-dimensional features and compression of classifier weights," in *ACM Int. Conf. Multimedia*, Orlando, Florida, USA, 3–7 Nov. 2014.

[30] H. Zhao, K.-H. Yap, and A. C. Kot, "Fusion learning using semantics and graph convolutional network for visual food recognition," in *IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2021.

[31] A. Xompero, R. Sanchez-Matilla, R. Mazzon, and A. Cavallaro, "CORSMAL Containers Manipulation," 2020, (1.0) [Data set]. Queen Mary University of London. [Online]. Available: http://corsmal.eecs.qmul.ac.uk/containers_manip.html

[32] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *Proc. Int. Workshop Mach. Learning Signal Process.*, 2015.

[33] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, pp. 21–27, 1967.

[34] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[35] L. Breiman, "Random forests," *Machine Learning*, vol. 45, pp. 5–32, 2004.

[36] V. Vivek, S. Vidhya, and P. MadhanMohan, "Acoustic scene classification in hearing aid using deep learning," in *Int. Conf. Comm.Signal Proc.*, India, 28–30 Jul. 2020.

[37] S. Yang and W. Deliang, "Robust speaker identification using auditory features and computational auditory scene analysis," in *IEEE Int. Conf. Acoustics, Speech Signal Process.*, Las Vegas, Nevada, USA, 30 Mar./4 Apr. 2008.

[38] G. Deepanway and K. Maheshkumar, "Music genre recognition using deep neural networks and transfer learning," in *INTERSPEECH*, Hyderabad, India, 2–6 Sep. 2018.

[39] R. Shashidhar, P. Sudarshan, and S.-B. Puneeth, "Audio visual speech recognition using feed forward neural network architecture," in *IEEE Int. Conf. for Innovation in Technology*, Bengaluru, India, 6–8 Nov. 2020.

[40] S. Donaher, A. Xompero, and A. Cavallaro, "Audio classification of the content of food containers and drinking glasses," in *Europ. Signal Proc. Conf.*, Virtual, 23–27 Aug. 2021.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 27–30 Jun. 2016.

[42] G. Christmann and J.-T. Song, "2020 CORSMAL Challenge - Team NTNU-ERCReport," 2020. [Online]. Available: https://github.com/guichristmann/CORSMAL-Challenge-2020-Submission/blob/master/PaperReport.pdf

[43] Q. Liu, F. Feng, C. Lan, and R. H. M. Chan, "VA2Mass: Towards the fluid filling mass estimation via integration of vision & audio learning," in *IEEE Conf. Pattern Recognit. Workshops and Challenges*, Virtual, 10–15 Jan. 2021.

[44] 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*, Virtual, 10–15 Jan. 2021.

[45] 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*, Virtual, 10–15 Jan. 2021.

[46] T. Giannakopoulos, "pyaudioanalysis: An open-source python library for audio signal analysis," *PLoS ONE*, vol. 10, no. 12, p. 1–17, Jan. 2015.

[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Int. Conf. Learning Represent.*, San Diego, CA, USA, 7–9 May 2015.

[48] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[49] A. Bochkovskiy, C. Wang, and H. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," arXiv:2004.10934 [cs.CV], 2020.

[50] 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.*, Munich, Germany, 8–14 Sep. 2018.

[51] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, and B. Seybold, "CNN architectures for large-scale audio classification," in *IEEE Int. Conf. Acoustics, Speech Signal Process.*, 2017.
[52] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *IEEE Int. Conf. Acoustics, Speech Signal Process.*, New Orleans, LA, 2017.
[53] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Adv. Neural Inf. Process. Syst. Workshop Deep Learning*, 2014.
[54] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, Utah, USA, 18–22 Jun. 2018.
[55] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The Kinetics human action video dataset," arXiv:1705.06950 [cs.CV], 2017.
[56] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Int. Conf. Mach. Learning.*, Jul. 2015.
[57] M. Matsumoto and T. Nishimura, "Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, p. 3–30, Jan. 1998.