Detecting Counterfeit Liquid Food Products in a Sealed Bottle Using a Smartphone Camera

Bangjie Sun
National University of Singapore
bangjie@comp.nus.edu.sg

Sean Rui Xiang Tan
National University of Singapore
seantanr@comp.nus.edu.sg

Zhiwei Ren
National University of Singapore
renzhiwei@comp.nus.edu.sg

Mun Choon Chan
National University of Singapore
chanmc@comp.nus.edu.sg

Jun Han*
Yonsei University
junhan@cyphy-lab.org

ABSTRACT

We are witnessing a surge in the reported cases of counterfeit liquid products in the market including olive oil, honey, and alcohol. Counterfeiters often *adulterate* the liquid products by replacing a large portion of the authentic content with cheaper substitutes (e.g., mixing vodka with cheaper alcohol or potentially toxic methanol). Exacerbating the problem, the counterfeits are packaged and sealed to factory standards, rendering it extremely difficult for an average consumer to identify them. While solutions exist, they are often impractical for the general public as they require specialized and costly equipment. To overcome these limitations, we propose *LiquidHash*, a novel counterfeit liquid food product detection system. *LiquidHash* is a practical solution that only requires the use of a *commodity smartphone* to *detect* adulterated liquid products *without opening the bottles*. *LiquidHash* works by detecting and tracking the shape and movement of air bubbles that form inside the bottles. We implement *LiquidHash* and evaluate its feasibility with real-world experiments under varying conditions with a total of more than 500 minutes of video recording and observe an overall detection accuracy of up to 95%.

CCS CONCEPTS

• **Human-centered computing** → **Smartphones**; **Mobile phones**; • **Computing methodologies** → **Computer vision**.

KEYWORDS

Counterfeit, Liquid Testing, Smartphone Camera

ACM Reference Format:
Bangjie Sun, Sean Rui Xiang Tan, Zhiwei Ren, Mun Choon Chan, and Jun Han. 2022. Detecting Counterfeit Liquid Food Products in a Sealed Bottle Using a Smartphone Camera. In *The 20th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '22), June 25–July 1, 2022, Portland, OR, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3498361.3539776

*Corresponding author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MobiSys '22, June 25–July 1, 2022, Portland, OR, USA
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9185-6/22/06…$15.00
https://doi.org/10.1145/3498361.3539776

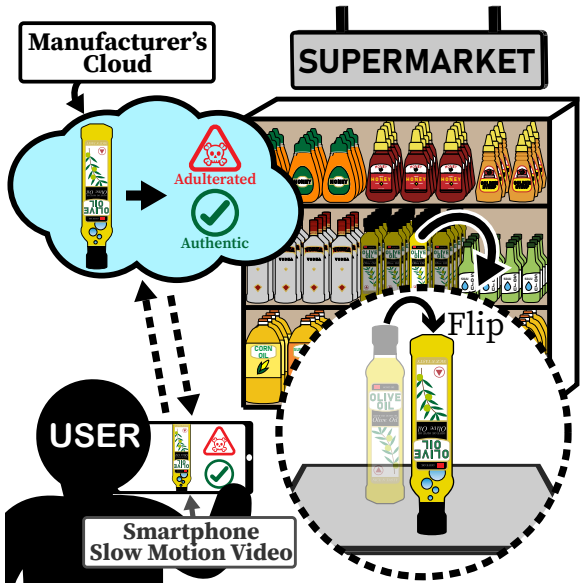

Figure 1: Figure depicts a use case scenario of *LiquidHash*. As Alice enters a supermarket wishing to purchase an olive oil, she wishes to verify if the bottle contains authentic content. She inverts the bottle and records it in motion using her smartphone camera. Ultimately, *LiquidHash* utilizes the captured air bubble movements to verify the liquid authenticity.

1 INTRODUCTION

The cases of counterfeit liquid food products are increasing and it is now a major societal concern. The World Health Organization (WHO) estimates that 25% of the alcohol consumed worldwide is counterfeit [57]. There are also many reports of counterfeit liquid food products involving olive oil [7, 12, 14, 30, 46, 52, 60, 72], honey [2, 20, 62, 69], and alcohol [10, 36, 54, 65–67, 78].

The surge in these cases can be attributed to the counterfeiters' economic benefits as they *adulterate*, or replace a large amount of authentic liquid contents with cheaper alternatives (e.g., replacing a significant portion of authentic vodka with cheaper alcohol or even potentially toxic methanol). Such adulteration can often cause detrimental health problems leading to fatalities [31, 36, 46, 57, 65, 66]. In addition, the liquid product manufacturers also suffer from significant monetary loss adding up to a total loss of $40 billion globally every year [11, 61, 63].

It is extremely difficult, however, for an average consumer to detect the adulterated liquid contents because the counterfeits are often packaged in authentic bottles which are easily accessible [78], and are sealed to factory standards [19, 73]. Hence, the general public is left vulnerable and relies on the authorities to test the liquid authenticity [57].

There are state-of-the-art solutions that attempt to analyze the liquid contents without opening the bottles. These include Near-Field (NIR) and Raman spectroscopy that utilize absorption of electromagnetic radiation to obtain chemical information of the liquid contents [15, 43, 48]. However, these solutions are not available to the general public as they use specialized and costly equipment. There are also recent research techniques that utilize wireless signals to identify the liquid types. Specifically, they leverage the coupling effect between the liquid content and the radio-frequency (RF) signals to measure electric permittivity [13, 26, 74, 79]. However, these solutions also require specialized equipment and setup. The wireless measurement is also susceptible to environmental effects such as multi-path interference.

To overcome the limitations of the state-of-the-art solutions, we ask the following question – can we achieve a *practical* solution that only leverages an average consumer's **commodity smartphone** to **detect** adulterated liquid products **without opening the bottles**? As an answer to this question, we propose *LiquidHash*, a liquid counterfeit detection system that utilizes a commodity smartphone camera to capture information of the liquid contents sealed inside the bottles. The core idea of *LiquidHash* is that we can infer liquid properties from the air bubble shapes and movements inside the bottle. This is possible because liquid properties, in particular, *density*, *viscosity*, and *surface tension* influence the *radius*, *aspect ratio*, and *terminal velocity* of bubbles as they rise to the top (See § 2.1). Hence, by quantifying these features from the observed bubbles, we are able to differentiate among different liquid products.

Figure 1 illustrates the process of using *LiquidHash* to detect adulteration in a liquid product. The user rotates the sealed bottle upside down while recording the bubble shape and movement in slow-motion. *LiquidHash* then analyzes the recording and processes the images to determine if the liquid product is *authentic* or *adulterated*. As an illustration, consider the following use case scenario. Alice enters a shop and wishes to purchase a bottle of olive oil. As a precaution, she wants to check the authenticity of the bottle of olive oil before purchasing. To do this, Alice uses the *LiquidHash* app installed on her phone to capture the images of the bubble shape and movement in the olive oil after she rotates the bottle. The "signature" extracted from these images are then compared against the "signature" provided by the manufacturer. This process is analogous to a *hash checksum verification* (hence the name *LiquidHash*).

Designing *LiquidHash*, however, comes with the following challenges. First, as the camera placement with respect to the bottles can vary significantly across different rotations, it is difficult to compare across different image captures consistently. To overcome this challenge, we utilize computer vision techniques to normalize the images to be robust against differences in camera distance and angle to the bottle.

Second, *LiquidHash* requires fine-grained segmentation and tracking of the individual bubbles detected. This is challenging because the shape of a bubble can change as it moves and the outer boundaries may not always be clearly defined. As traditional image processing techniques are insufficient for our purpose, we utilize a fully convolutional neural network (FCN) for bubble segmentation.

Third, based on the extracted features such as *radius*, *aspect ratio*, and *terminal velocity*, we need to infer the authenticity of the liquid. *LiquidHash* utilizes machine learning classification to train and predict the authenticity of the liquid contents by matching these extracted features to the features of authentic liquids that have been already learned.

We implement *LiquidHash* and evaluate its feasibility by conducting real-world experiments under varying conditions utilizing three authentic liquid products – i.e., extra virgin olive oil, pure raw honey, and vodka – and eight different adulterants. We invite multiple participants to rotate the bottles containing different liquid contents while recording them with a smartphone camera, and collect more than 500 minutes of recording. Our evaluations show that *LiquidHash* yields detection accuracy of up to 95%. In summary, we make the following contributions:

- We propose, *LiquidHash*, a counterfeit liquid detection system that utilizes *commodity smartphone camera* to detect adulterated liquid contents *without opening* a sealed bottle.
- We present the design and implementation of *LiquidHash* that overcomes the challenges of extracting air bubble features in a noisy environment with commodity smartphone camera and utilizing these features to ultimately detect adulterated liquid contents using computer vision and machine learning techniques.
- We demonstrate through a set of comprehensive evaluation that *LiquidHash* is able to yield high accuracy in real-world settings across different liquid products and adulterants.

2 BACKGROUND AND FEASIBILITY STUDY

We first present the relevant background information on the physics model of rising air bubbles and capture of slow-motion video using smartphone cameras. We then conduct a preliminary study to demonstrate *LiquidHash*'s feasibility.

2.1 Physics Model of Rising Air Bubbles

Different liquid contents exhibit unique physical properties such as *density*, *viscosity* and *surface tension* [37, 49, 55, 71, 77]. However, these properties cannot be directly measured from sealed bottles. Hence, *LiquidHash* utilizes the observations of the air bubbles arising inside the bottles to determine the authenticity of the liquid food products. Specifically, we find that the characteristics of the bubbles, namely size and shape (i.e., *radius* and *aspect ratio*) and rising speed (i.e., *terminal velocity*), are correlated with the liquid physical properties [4, 8, 25, 34, 35, 39, 44, 68, 80]. Stokes' Law [6] and Young-Laplace Law [42, 82] describes the relationship between the characteristics of the bubbles and liquid properties. Specifically, for a **spherical** bubble, Stokes' Law describes the relation of terminal velocity (V) on the liquid *density* (ρ), *viscosity* (μ) and *bubble radius* (R) as: $V = \frac{2}{9}\frac{(\rho_{air}-\rho_{liquid})}{\mu}gR^2$. Furthermore, Young-Laplace Law describes the relation of the pressure difference (Δp) between the bubble and the liquid, the radius (R) of a **spherical** bubble and

the surface tension of the liquid (σ) as $\Delta p = 2\frac{\sigma}{R}$. Intuitively, surface tension describes the force which resists an increase in surface area between the liquid and the bubble. When surface tension decreases, resistance is lowered and this allows the bubble to expand to keep the pressure difference (Δp) constant.

The above equations provide a procedure to use bubbles to distinguish between different types of liquid. Specifically, we can use a camera to capture the rising bubble's *radius* and *terminal velocity*. By re-ordering the Stokes' Law equation, we can compute V/R^2, a **constant value** encapsulating liquid *density* and *viscosity*. We name it as *V-R ratio*. Furthermore, using Young-Laplace Law equation, we know that bubble's *radius* is proportional to *surface tension*. If a liquid food product is adulterated with another type of liquid of different physical properties, the physical properties of the mixture deviate from the original ones, leading to **different values** of V/R^2 and R. However, both of the laws assume that bubbles are spherical and do not interfere with each other, but these assumptions often do not apply in practice, where bubbles may collide with each other and take on non-spherical shapes (e.g., spherical-cap, ellipsoidal and irregular shapes). To address these issues, in addition to the features computed based on R and V/R^2, *LiquidHash* incorporates bubble shape (estimated using aspect ratio (E) [18, 38, 44]) as an additional feature (See § 4.3.1).

2.2 Utilizing Smartphone Cameras

Most commodity smartphones [5, 58] are equipped with cameras that support video recording in slow-motion mode with high frame rates (i.e., 120fps and 240fps). *LiquidHash* uses these slow-motion cameras to capture fast-moving bubbles in liquid food products as a high frame rate helps *LiquidHash* accurately capture bubble trajectories and measure bubble characteristics. However, the characteristics of bubbles captured by a camera depend significantly on the camera's distance and viewing angle to the bottle. When a spherical bubble is projected to the image plane, we only observe the cross-section of the bubble. The distance between the camera and the bottle determines the resolution of the bubble (i.e., the number of pixels constituting the bubble in the image). Finally, when light passes through both the liquid and the bottle, it bends due to refraction. As a result, bubbles on the image are distorted and appear with different shapes and sizes depending on the camera's viewing angle, increasing the challenges of *LiquidHash*. To address this issue, *LiquidHash* extracts distance and orientation information of markers on bottles to mitigate the impact of the distance and to take into account the viewing angle from the camera to the bottle (See § 4.2).

2.3 Feasibility Study

We conduct a feasibility study to test our hypothesis that we can utilize bubble's characteristics (i.e., *radius*, *aspect ratio*, and *terminal velocity*) to distinguish between different types of liquid. Specifically, we prepare one authentic pure olive oil (Oil_{auth}), and two adulterated olive oil: Oil_{fake_1} (70% Oil_{auth} + 30% sunflower oil) and Oil_{fake_2} (50% Oil_{auth} + 50% sunflower oil). We place each of them in separate bottles as depicted in Figure 2(a). To generate consistent bubbles without adding other sources of noise, we then insert a thin tube into each bottle, and attach a syringe to the other end of the tube. We generate bubbles by gently pressing the syringe, and capture bubbles using a smartphone camera placed on a tripod at a fixed distance from the bottle.

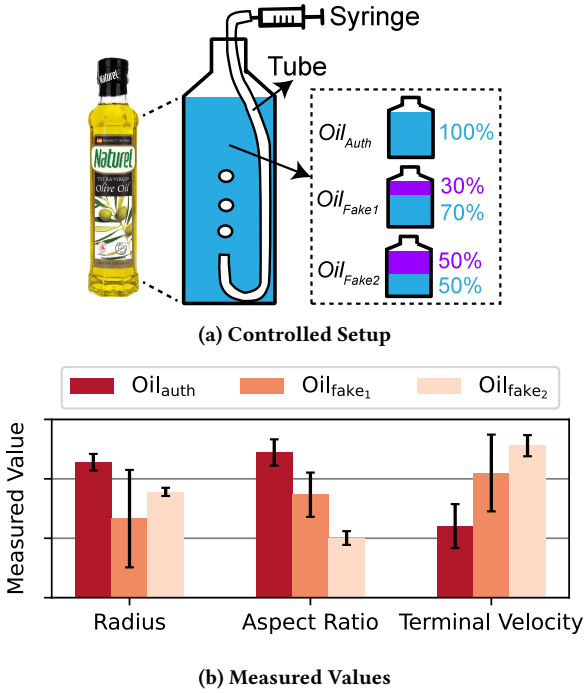

Figure 2: (a) depicts the controlled setup of feasibility study. We insert a thin tube into the bottle, and use a syringe to generate bubbles. (b) depicts the average values of measured radius, aspect ratio, and terminal velocity from bubbles in each type of oil. Figure indicates the possibility of using the measured values to differentiate authentic oil from the adulterated ones.

Figure 2(b) depicts the measured average radius, aspect ratio, and terminal velocity. The measured radius of Oil_{auth} is higher than that of Oil_{fake_1} and Oil_{fake_2}, despite significant noise in measuring Oil_{fake_1}. There is a general trend of decreasing aspect ratio as the concentration of sunflower oil increases from 0% to 50%. These observations on radius and aspect ratio correlate with Young-Laplace Law equation because olive oil has a higher surface tension than sunflower oil [55]. Furthermore, the terminal velocity increases as the concentration of sunflower oil increases from 0% to 50%. This observation correlates with Stokes' Law equation as sunflower oil has a higher ratio of density over viscosity (i.e., $\frac{\rho_{sunflower}}{\mu_{sunflower}} > \frac{\rho_{olive}}{\mu_{olive}}$) [55]. From this preliminary study, we demonstrate the feasibility of utilizing bubbles to distinguish different liquid contents.

3 SYSTEM MODEL

We introduce our system model, describing the goal, requirements and assumptions of *LiquidHash*. The main goal of *LiquidHash* is to verify the authenticity of a liquid food product in a sealed bottle leveraging observations of air bubbles. We design *LiquidHash* to fulfill the following requirements: (1) be accurate in classifying

authentic and fake liquid food products (accuracy), (2) work without opening bottles (usability), (3) work by leveraging commodity devices and real-world sealed bottles (deployability).

To achieve the aforementioned goal while satisfying the requirements, we make the following assumptions: (1) counterfeiters strive for profit gain by replacing a large amount (i.e., at least 30%) of authentic liquid content with cheap alternatives; (2) authentic and adulterated liquids produced by counterfeiters are distinguishable by their properties, namely *density*, *viscosity*, and *surface tension*; and (3) bottles and liquid contents are semi- or fully transparent to allow observations of the bubbles.

4 SYSTEM DESIGN

We present *LiquidHash*'s design and implementation details.

4.1 System Overview

LiquidHash's goal is to verify a liquid food product in a sealed bottle by leveraging characteristics of rising air bubbles, namely *radius*, *aspect ratio*, and *terminal velocity* utilizing computer vision and supervised learning techniques. *LiquidHash* is divided into two phases - *Bootstrapping* and *Verification Phases*. *Bootstrapping Phase* occurs offline, where the manufacturers train their liquid content models by collecting a large set of video data, including both authentic as well as adulterated liquid contents, along with ground truth labels. *LiquidHash* then utilizes the trained model in its *Verification Phase*, which occurs online as the consumers utilize *LiquidHash* app on their phone to record videos of unknown liquid food products to verify the authenticity.

Figure 3 illustrates *LiquidHash*'s design overview. *LiquidHash* takes the recorded video as input to the *Pre-processing* module (§ 4.2), which selects and processes frames that contain useful bubble information. Next, the processed frames are input to the *Bubble Feature Extraction* module (§ 4.3), which segments and tracks the bubbles to extract features. The *Prediction* module (§ 4.4) takes the features as input to train a machine learning model in the *Bootstrapping Phase*, and predict the authenticity of an unverified liquid food product in the *Verification Phase*. We now present the details of *LiquidHash*'s design.

4.2 Pre-processing

The goal of this module is to process the frames such that (1) the bubbles are comparable despite the differences in camera distance and angle; and (2) it reduces redundant information from input video. It takes as input the *video recordings* and outputs the *processed frames* and a *reference ratio* (i.e., a ratio inversely proportional to the distance to the camera), as depicted in Figure 4. We first select the frames that contain bubble information in the *Frame Selection* stage (§ 4.2.1). We then ensure the viewing angle from the camera to the bottle is correct in the *Pre-screening Test* stage (§ 4.2.2). If ensured, we proceed to the following stages in parallel. In the *Frame Processing* stage (§ 4.2.3), we process the selected frames to output the processed frames. Simultaneously, in the *Distance Estimation* stage (§ 4.2.4), we extract the distance to the camera as a *reference ratio*.

4.2.1 Frame Selection. In this stage, *LiquidHash* aims to select frames that contain bubble information. It takes raw input frames and outputs bubbles' moving direction (i.e., *bubble vector*), positions of pre-determined markers (i.e., *bottle markers*) and *steady frames*. We define *steady frames* as frames in which the bottle is inverted and the impact of bottle rotation to the bubbles is minimal. We select *steady frames* by using an *object detector*, a computer vision technique, to recognize pre-determined markers on bottle and bubbles' moving direction. We train the Faster-RCNN [50] *object detector* with annotated bottle and bubble images. Faster-RCNN is a real-time object detection neural network. Using transfer learning, we augment a pre-trained Faster-RCNN network with around 200 annotated images of bottles and bubbles to implement our *object detector*. *LiquidHash* samples one in every 60 frames of input video recorded at 240fps, and process sampled frames to obtain ① bubble vector, ② bottle markers and ③ steady frames.

① Bubble vector. We use the *object detector* to obtain a set of bubble positions and compute the best-fit line of the bubble positions to represent bubble's moving direction, $l_{bubbles}$. We compute the average of square distances, d, between each bubble position and $l_{bubbles}$.

② Bottle markers. We use the *object detector* to obtain a set of marker positions, $P_{markers}$. As a proof-of-concept, we utilize a black square on the bottle, and define the four vertices as markers. In practice, *LiquidHash* can also be extended to utilize different logo and the product description texts on commodity liquid bottles.

③ Steady frames. For each sampled frame, we use ① bubble vector to check bubbles' moving direction and ② bottle markers to check bottle orientation to select *steady frames*. First, we ensure that bubbles move in a straight line which indicates that bubbles are minimally affected by liquid perturbation. When the liquid is recently perturbed by bottle rotation, bubbles move erratically in curved trajectories. Such bubbles are undesirable as they exhibit noisy characteristics. As the liquid settles, bubbles tend to move in a straight line. We select the frame, if $d < d_\theta$, where d_θ is an empirically set threshold, to ensure trajectories of bubbles are straight. Second, a subset of bottle markers is always positioned in the lower half of the frame, or below other markers when the bottle is inverted. We compare the positions of these markers to determine if the bottle is inverted and discard frames where the bottle is not. We output bottle markers ($P_{markers}$), bubble vector ($l_{bubbles}$) and *steady frames* to the next stage.

4.2.2 Pre-screening Test. Subsequently, *LiquidHash* validates each trial to ensure camera angle is correct. It takes as input bottle markers ($P_{markers}$), bubble vector ($l_{bubbles}$) and *steady frames* from the *Frame Selection* stage, and outputs them to the next stage if a trial is valid. Recall from § 2.2 that the viewing angle from the camera to the bottle may affect the sizes and shapes of the captured bubbles due to refraction through the bottle. *LiquidHash* rejects a trial when a subset of markers is occluded and cannot be detected by the *object detector*. We also utilize the pixel distances between markers to further examine the viewing angle. In addition, we require the number of selected frames to be large to contain sufficient bubble information. *LiquidHash* requires the user to retry if the trial fails the pre-screening test.

4.2.3 Frame Processing. *LiquidHash* then removes redundant information. It takes as input bottle markers ($P_{markers}$), bubble vector ($l_{bubbles}$) and *steady frames* from the *Pre-screening Test* stage, and

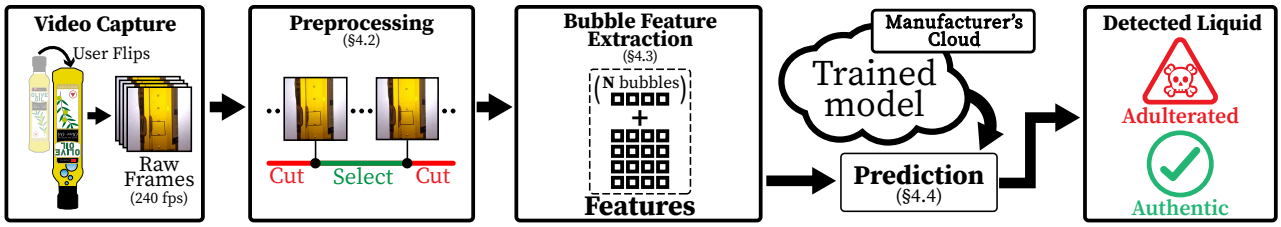

Figure 3: Figure depicts the flowchart of *LiquidHash*'s design. During the *Bootstrapping Phase*, liquid manufacturers train models with video data from authentic products and different types of adulteration, and save the model in the cloud. During the *Verification Phase*, the user records a slow-motion video of rotating the bottle. *LiquidHash* analyzes the recorded video by detecting the bottle and rising air bubbles. *LiquidHash* processes the bubbles to determine if the liquid food product is authentic or adulterated.

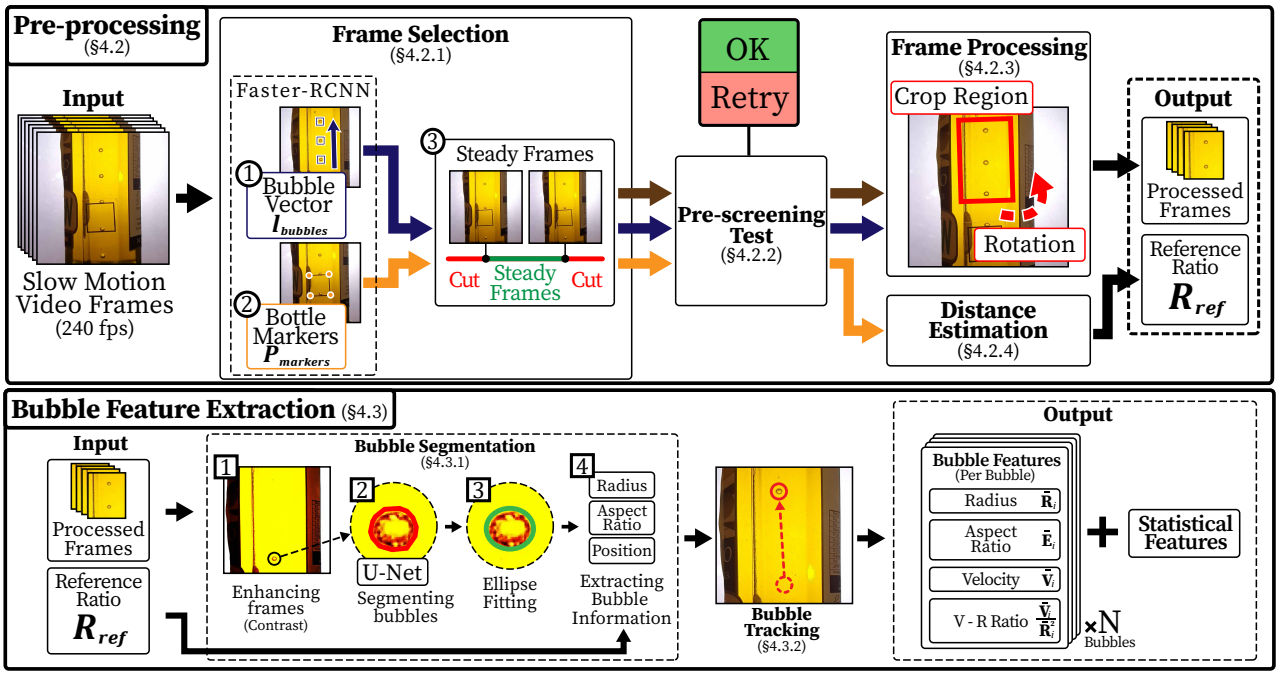

Figure 4: Figure depicts *LiquidHash*'s *Pre-processing* (§ 4.2) and *Bubble Feature Extraction* (§ 4.3) module pipelines. *Pre-processing* module processes raw input video frames, and extracts a reference ratio containing distance information. *Bubble Feature Extraction* module extracts features from processed frames, and normalizes them using the reference ratio.

outputs the processed frames. We locate a small region in *steady frames* to contain only bubble information, by adding offsets to $P_{markers}$. In this region, bubbles are mostly visible to the camera because regions between bottle labels generally do not occlude the liquid content. We crop the region and rotate it to align $l_{bubbles}$ and the vertical axis of the image. We output the processed frames to the *Bubble Feature Extraction* module.

4.2.4 Distance Estimation. *LiquidHash* mitigates the impact of the distance of the camera to the bottle. It takes as input bubble markers ($P_{markers}$) from *Pre-screening Test* stage, and outputs a *reference ratio*, R_{ref}, the ratio of an object's size measured in the real-world to that in the image. We compute R_{ref} as the ratio of a real-world distance between markers, L_{actual}, to detected pixel distance, L_{pixel}, by $R_{ref} = L_{pixel}/L_{actual}$. This ratio is inversely proportional to the distance. We output R_{ref} to *Bubble Feature Extraction* module.

4.3 Bubble Feature Extraction

Utilizing the *processed frames* and the *reference ratio* (R_{ref}), this module extracts features corresponding to the bubble characteristics, including *radius*, *aspect ratio*, and *terminal velocity*, in addition to *V-R ratio* (See § 2.1), to perform machine learning training and prediction through the following stages as depicted in Figure 4. During the *Bubble Segmentation* stage (§ 4.3.1), we utilize the instance segmentation technique to accurately segment bubbles in each frame, and extracts each bubble's radius and aspect ratio. During the *Bubble Tracking* stage (§ 4.3.2), we devise a tracking algorithm to

obtain bubbles' unique trajectories across frames to obtain bubble's terminal velocity and reduce measurement errors in segmentation. We then take average values of radius, aspect ratio and terminal velocity as *bubble features*. In addition, we also extract *statistical features* computed based on the *bubble features*, namely, the standard deviation, harmonic mean, minimum, and maximum, to reveal the impact of rotation to bubble characteristics.

4.3.1 Bubble Segmentation. In this stage, *LiquidHash* segments pixels containing the bubbles from the background in each frame, and estimates the radius, and aspect ratio of the bubbles by proceeding with the following steps.

1 **Enhancing frames.** We first enhance the images to increase the contrast between bubble and background pixels to aid the segmentation. For each pixel with value c_{pixel}, we scale the value using $c_{pixel} = \alpha \cdot c_{pixel} - \beta$, where α and β are empirically set values of 2 and 200, respectively.

2 **Segmenting bubbles.** We segment the bubble from the background pixels using U-Net, a neural network frequently used in biomedical image segmentation [53]. We utilize U-Net for its performance even with relatively small training dataset. We train the U-Net bubble segmentation with only 200 annotated bubble images.

3 **Fitting an ellipse.** To mitigate segmentation errors due to intermittently bright pixels within the bubble, we approximate an ellipse that spans the bubble region. We choose an ellipse because only spherical and ellipsoidal bubbles obey the aforementioned physics model (See § 2.1) and their cross-sections are in the form of ellipse.

4 **Extracting bubble information.** Based on the correctly segmented bubble pixels, we extract bubble radius, aspect ratio and position in the following steps. First, we estimate the bubble radius using the *oval equivalent radius* [70] that captures the pressure and resistance experienced by ellipse shaped objects in the liquid. Specifically, we use width (w) and height (h) of the fitted ellipse normalized by the *reference ratio* (R_{ref}, See § 4.2.4), respectively, to compute the radius. Second, we approximate the bubble aspect ratio (E) to capture the overall shape of the bubble [44]. We compute $E = \min(w,h)/\max(w,h)$. E is always between 0 and 1, and decreases as the circularity of bubble decreases (i.e., bubble becomes flatter). Third, we calculate bubble position using the **topmost** pixel position of the segmented pixels because the shape change at the top is almost negligible whereas the tail of the bubble experiences significant shape changes [44]. Finally, we output bubble radius, aspect ratio and position of all bubbles in each frame.

4.3.2 Bubble Tracking. In this stage, we aim to track bubbles across frames to identify unique bubble trajectories, using bubble radius, aspect ratio and position from the previous stage. This is to obtain bubble's terminal velocity and reduce measurement errors in radius and aspect ratio arising from bubble segmentation. However, it is challenging as bubbles may overlap with each other and may not be visible or detected in some frames. To overcome this challenge, we devise a tracking algorithm based on the observation that the same bubble appearing in consecutive frames has a small travelled distance and small changes in the radius and aspect ratio. The tracking algorithm consists of two parts, namely (1) tracking a single bubble from a given starting position, and (2) tracking all unique bubbles.

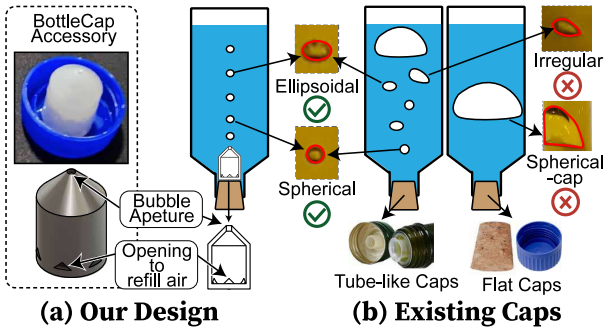

Figure 5: (a) depicts our proposed design of the bottle cap *accessory*. (b) depicts examples of undesirable bubbles induced by different existing bottle caps.

(1) Tracking a single bubble. The tracking algorithm takes as input an initial bubble position and searches for similar bubbles in subsequent frames to output a bubble trajectory. For an initial bubble position P_t at frame t, we search the bubbles in the next k frames such that the distance between two bubbles are sufficiently small, i.e., $||P_{t+k} - P_t|| < dk$ where d is an empirically set threshold representing the maximal distance per frame. We find the smallest k and continue searching from bubble position P_{t+k}.

(2) Tracking all unique bubbles. We identify all unique bubble trajectories by repeating part (1) on unique starting bubbles. If a bubble is already in the trajectory of another bubble, it is no longer a unique bubble. Thus, we use a set, $S_{visited}$, to keep track of all previously tracked bubbles to avoid repetitive tracking.

4.4 Prediction

In this module, *LiquidHash* performs training and prediction using features extracted from the *Bubble Feature Extraction* module. *LiquidHash* takes video data of authentic liquid food products and target types of adulteration, with their ground truth labels (i.e., authentic and fake), to train a machine learning classifier in the *Bootstrapping Phase*. In the *Verification Phase*, *LiquidHash* tests newly collected video data to predict the authenticity of an unverified liquid. However, it is challenging to rely on a traditional classifier due to uneven number and weightage of *bubble features* (i.e., radius, aspect ratio, terminal velocity and *V-R ratio*) extracted from each bubble and *statistical features* extracted by combining information across all bubbles. To overcome this challenge, we **ensemble** two classical machine learning classifiers, Clf_{bubble} and Clf_{stats}, for training and prediction. For prediction, each classifier predicts the a probability (i.e., P_{bubble} and P_{stats}) for each class (i.e., authentic and fake). We add the probabilities to obtain the final probability, P_{final}, and make a decision by taking the class label of the highest probability (i.e., output is authentic when $P_{final}(authentic) > P_{final}(fake)$, and fake otherwise). In our implementation, we use Adaptive Boosting (AdaBoost) [17, 28], a statistical classification meta-algorithm, as our classifier. An AdaBoost classifier adjusts weights of incorrectly classified instances to focus on difficult cases where extracted features of different types of liquid are numerically similar to each other.

Use Case	Authentic Product	Adulterant	%
Olive Oil	O_A:Naturel Extra Virgin	-	0
	O_A:Naturel Extra Virgin	O_{F1}:Knife Groundnut Oil	30
	O_A:Naturel Extra Virgin	O_{F2}:Golden Circle Sunflower Oil	30
	O_A:Naturel Extra Virgin	O_{F3}:OKI Premium Soybean Oil	30
	O_A:Naturel Extra Virgin	O_{F4}:Golden Circle Corn Oil	30
Honey	H_A:Balparmak Pine	-	0
	H_A:Balparmak Pine	H_{F1}:Little Bee Golden Syrup	30
	H_A:Balparmak Pine	H_{F2}:Wescobee Australian Honey	30
Vodka	V_A:Smirnoff Red	-	0
	V_A:Smirnoff Red	V_{F1}:Chamisul Soju Fresh	30
	V_A:Smirnoff Red	V_{F2}:Water	30

Table 1: Table depicts the list of liquid food products and adulterants in our experiments.

4.5 Enhancing Bubble Generation

Recall that *LiquidHash* relies on features extracted from bubble characteristics to determine the authenticity of liquid food products. To further improve *LiquidHash*'s usability and capability of differentiating authentic and adulterated liquid food products, we devise a bubble generation mechanism, as an **accessory** to a bottle cap, to assist users to generate a large number of consistent bubbles. Figure 5(a) depicts the design of our proposed bottle cap accessory (See § 6.1 for practicality concerns). Specifically, the accessory has a small *bubble aperture* (i.e., an opening to allow air to flow out) to constrain the size of bubbles to a small range, and slow down the rate of bubble generation to ensure that most bubbles are generated after liquid stabilizes. When a bottle is inverted, the air is released slowly via the small *bubble aperture* in a steady stream of regularly shaped spherical and ellipsoidal bubbles. The proposed accessory helps *LiquidHash* to address two practical issues. First, large bubbles arise due to insufficient collision forces to break the air gap in the bottle. Large bubbles has non-spherical shapes (e.g., spherical-cap and irregular shapes) and significant shape changes during the rising process. *LiquidHash* discards these bubbles which do not obey the physics model (See § 2.1). Second, bubbles are generated before the liquid stabilizes, and are significantly affected by the liquid perturbation due to the bottle rotation. *LiquidHash* discards video frames containing these bubbles. Figure 5(b) illustrates examples of undesirable bubbles discarded by *LiquidHash*.

5 EVALUATION

We now present *LiquidHash*'s comprehensive evaluation.

5.1 Experiment Setup

Apparatus. Figure 6 illustrates our experimental procedures. We evaluate *LiquidHash* in three use cases, namely olive oil, honey and vodka, with authentic liquid food products and corresponding common off-the-shelf adulterants [2, 7, 10, 46, 60, 65, 69, 72]. In total, we have 11 instances of liquid content enumerated in Table 1. Note that the three chosen types of liquid are representative of a wide range of common liquid food products with viscosity in the range from less than one centipoise (i.e., vodka) [45] to 10,000 centipoises (i.e., honey) [51]. In addition to the three authentic liquid food products (i.e., O_A, H_A, V_A), we prepare four adulterated instances of olive oil (i.e., O_{F1}, O_{F2}, O_{F3}, O_{F4}), two adulterated instances of honey (i.e., H_{F1}, H_{F2}), and two adulterated instances of vodka (i.e., V_{F1}, V_{F2}), all packaged in their corresponding authentic bottles. We use the original packaging for these bottles *without* removing any brand covers, printed labels, and stickers. We prepare all adulterated liquid food products by replacing 30% of the authentic liquid content with adulterants, to represent majority of reported real-world incidents [46, 65, 69]. In this experiment setup, we demonstrate the sensitivity of *LiquidHash* to various amounts of differences in liquid properties. Despite the 30% adulterant added, the difference in liquid properties between authentic and adulterated liquid contents is in a wide range. For example, the difference in viscosity is in the granularity of millicentipoise, centipoise and several hundred centipoises in the use cases of vodka, olive oil, and honey, respectively. Furthermore, we also apply the proposed bottle cap accessory when conducting the experiments (See § 4.5).

Data Collection. We recruit five participants (two female and three male) to perform two tasks under a room lighting condition and a constant temperature of 25°C. We limit the participant numbers due to COVID-19 restrictions. We ask each participant to rotate each of the 11 liquid contents up to eight times. The total number of trials is over 400 in our experiment, exclusive of the controlled experiments. It is important to note that the rotation pattern is different across trials (even performed by the same participant). Participants record slow-motion videos of each rotation with a smartphone placed on a tripod with an additional light source (i.e., a table lamp) to illuminate the liquid content. Overall, we have a total of more than 500 minutes of slow-motion video recording. Figure 6 illustrates this experimental setup. To ensure that each collected video data passes *LiquidHash*'s pre-screening test (See § 4.2.2), each participant undergoes a five-minute training session. We conduct this study upon the approval of our institution's Institutional Review Board.

Evaluation Metrics. We define the metrics to evaluate *LiquidHash*'s performance. We define A_t and A_f as correctly and falsely classified authentic liquid food products, respectively, and F_t and F_f as correctly and falsely classified fake liquid food products, respectively. *Accuracy* refers to the percentage of correctly classified instances in all tested authentic and fake liquid food products (i.e., $Accuracy = (A_t+F_t)/(A_t+F_t+A_f+F_f)$). *Precision* is the percentage of correctly classified fake instances in all tested fake liquid food products (i.e., $Precision = F_t/(F_t + F_f)$). *Recall* is the percentage of correctly classified authentic instances in all tested authentic liquid food products (i.e., $Recall = A_t/(A_t + A_f)$). When we evaluate *LiquidHash*'s performance on a specific class (i.e., authentic or fake), we use $Accuracy_{auth}$ and $Accuracy_{fake}$ to represent the percentage of correctly classified authentic and fake instances in the tested liquid food products of the same class, respectively (i.e., $Accuracy_{auth} = A_t/(A_t + A_f)$ and $Accuracy_{fake} = F_t/(F_t + F_f)$).

5.2 Overall Performance

Data Preparation. To evaluate the overall performance of *LiquidHash*, we collect 125, 120 and 120 video clips for the olive oil, honey, and vodka along with their corresponding adulterants, all packaged in their corresponding authentic bottles. We utilize the data for cross validation across the five participants – i.e., training with four participants' data and testing on the remaining participant's data at a time.

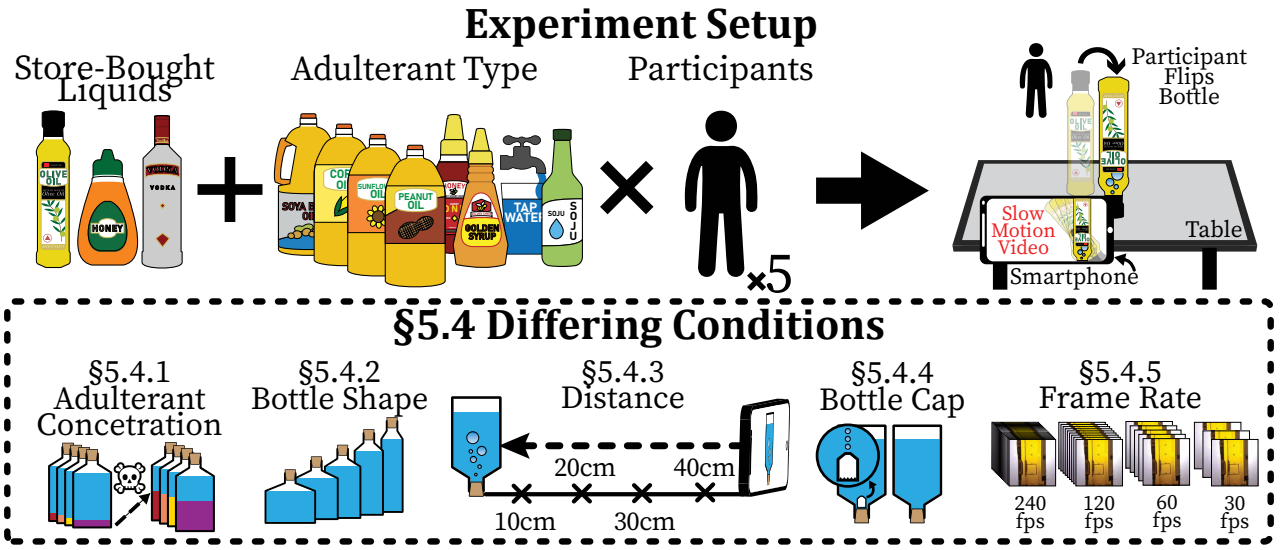

Figure 6: Figure depicts the setup of our experiments. We evaluate the accuracy of *LiquidHash* across different liquids and adulterants when used by different participants. We also vary experimental and environmental conditions to comprehensively evaluate *LiquidHash*.

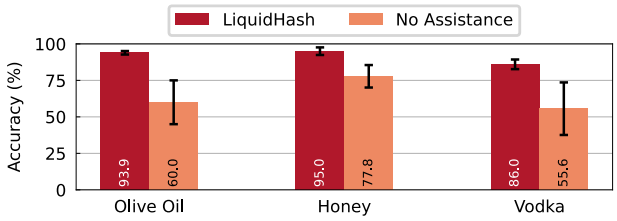

Figure 7: Figure depicts the accuracy of *LiquidHash* in three use cases, namely *olive oil, honey,* and *vodka,* compared to the accuracy of *No Assistance* method.

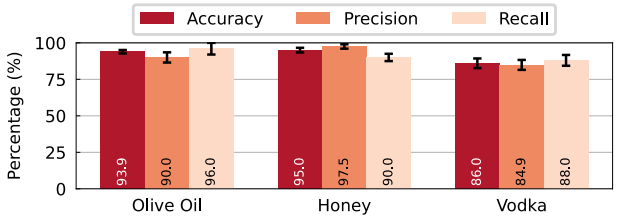

Figure 8: Figure depicts the accuracy, precision and recall of *LiquidHash* in three use cases, namely *olive oil, honey,* and *vodka.*

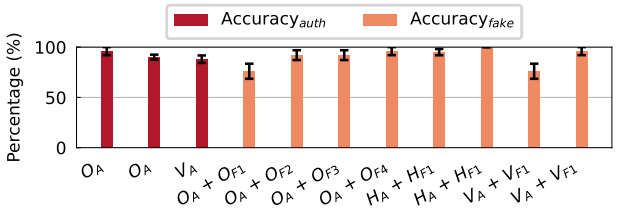

Figure 9: Figure depicts the accuracy breakdown in different types of authentic and adulterated liquids.

Baseline. We resort to *No Assistance* method as our baseline by simply asking the participants to identify authentic liquid products without any assistance, resembling consumers point-of-view in the real-world – i.e., they interact with the liquid food products freely without opening the bottles. We allow participants to use their smartphones to obtain additional information of the products, such as referencing online images to compare the appearances and using the flashlight to compare the liquid colors.

Results. We illustrate the overall results in Figure 7, where we observe that *LiquidHash* achieves an accuracy of 93.9%, 95.0% and 86.0% for the use cases of olive oil, honey and vodka, respectively, significantly outperforming the *No Assistance* method by 33.9%, 17.2%, and 30.4%, respectively. On the contrary, the *No Assistance* method only yields an accuracy of 60.0%, 77.8% and 58.3%, respectively. On average, 60%, 40% and 70% of fake olive oil, fake honey and fake vodka, respectively, are misclassified as authentic while only 20% of authentic vodka is misclassified as fake and none of authentic olive oil and honey is misclassified. These results indicate that participants are more biased toward authentic bottles (i.e., classifying as authentic when they are unsure), leading to higher false-positive rates (i.e., fake samples misclassified as authentic). Furthermore, as we simply mix authentic liquids and adulterants without adding any colorant, visible difference in the color of the adulterated liquid is unavoidable, especially for honey adulterated with syrup (i.e., $H_A + H_{F1}$) which all participants can observe the color difference. This explains the higher accuracy of the *No Assistance* method than that of a random guess (i.e., 50%) in all use

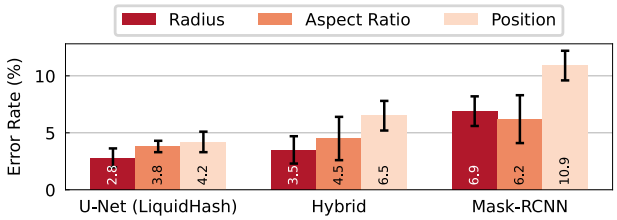

Figure 10: Figure depicts the error rates in measuring radius, aspect ratio, and position respectively by different deep learning models. U-Net outperforms the others by yielding the lowest error rates.

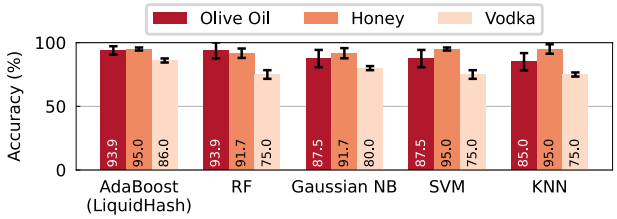

Figure 11: Figure depicts the accuracy of various machine learning classifiers in the final decision making.

cases, and the unexpectedly high accuracy of 77.8% in the honey use case. Nevertheless, we also observe that human judgement in the *No Assistance* method is overly sensitive to color differences between liquids, which can lead to misclassification.

We further illustrate a more detailed results of *LiquidHash* in Figure 8. We observe that *LiquidHash* achieves high accuracy, precision, and recall in the use cases of olive oil, honey, and vodka. However, vodka yields the lowest accuracy of 86%. We attribute this to the significant shape and size changes of bubbles during the rising process in water-like liquids with low viscosity. Furthermore, *LiquidHash* yields precision of 90.0%, lower than accuracy by 3.9%, in the use case of olive oil, and precision of 84.9%, lower than accuracy by 1.1%, in the use case of vodka because specific types of adulterants are less likely to be distinguished due to their liquid properties being similar to the authentic liquids. Recall from § 2.1 that if liquid properties are similar, the bubble characteristics may not be sufficiently different for *LiquidHash* to distinguish. We can understand this effect by focusing on the accuracy of specific types of adulterants, which we illustrate in Figure 9. *LiquidHash* achieves the lowest accuracy when detecting vodka adulterated with soju (i.e., $V_A + V_{F1}$), and olive oil adulterated with peanut oil (i.e., $O_A + O_{F1}$). We attribute this to the fact that liquid properties of these liquids are similar [37, 49, 55, 71]. Overall, *LiquidHash* achieves high accuracy in both verifying authentic liquid food products, and detecting different types of adulteration.

5.3 Performance of System Modules

We evaluate the internal modules of *LiquidHash* to compare between alternative designs.

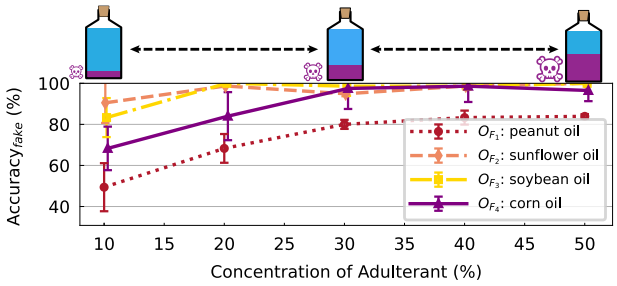

Figure 12: Figure depicts the $Accuracy_{fake}$ with respect to varying concentration of four types of adulterants.

5.3.1 Deep Learning Models for Bubble Segmentation. Recall that we utilize U-Net for bubble segmentation in § 4.3.1. We now evaluate different deep learning models and compare the performance. Specifically, we evaluate the effect of the models on the measurement errors of bubble radius, aspect ratio and position in ten representative bubble images, compared to ground truth values from manual annotations. From Figure 10, we observe that U-Net has the lowest error rates compared to Mask-RCNN [29] and a hybrid approach (i.e., Faster-RCNN for object detection and U-Net for segmentation). Note that high error rates in the measurement of bubble characteristics are detrimental to *LiquidHash*'s performance, because the numerical difference in these characteristics may not be large enough between authentic and adulterated liquids, and large measurement errors can easily result in misclassifications.

5.3.2 Classifiers for Decision Making. Recall that we utilize Adative Boosting (AdaBoost) for prediction in § 4.4. We also evaluate other widely adopted classifiers including Random Forest (RF), Gaussian Naive Bayes (Gaussian NB), Support Vector Machine (SVM) and K-Nearest-Neighbours (KNN). Figure 11 demonstrates that AdaBoost outperforms other classical machine learning classifiers. This is attributed to the fact that AdaBoost is more likely to distinguish between numerical features than other classifiers as it adjusts weights of incorrectly classified instances to tackle boundary cases.

5.4 Impacts on Experimental Conditions

We now evaluate *LiquidHash* under varying experimental and environmental conditions. We conduct the experiments using olive oil and its adulterants, namely corn, soybean, sunflower and peanut oil as depicted in Table 1.

5.4.1 Varying Concentration of Adulterants. Adulterated liquid food products are made by replacing a percentage of authentic liquid content with a fake liquid. As the amount of the fake liquid added per unit volume increases (i.e., concentration of adulterant), the difference in liquid properties between the mixture and authentic liquid also increases. We evaluate *LiquidHash*'s performance against varying concentration of common adulterants of olive oil, namely corn, soybean, sunflower and peanut oil, from 10% to 50%, all packaged in the authentic olive oil bottles [1]. In Figure 12, we

[1]We note that the majority of reported real-world incidents witness adulteration of at least 30% concentration [46, 65, 69].

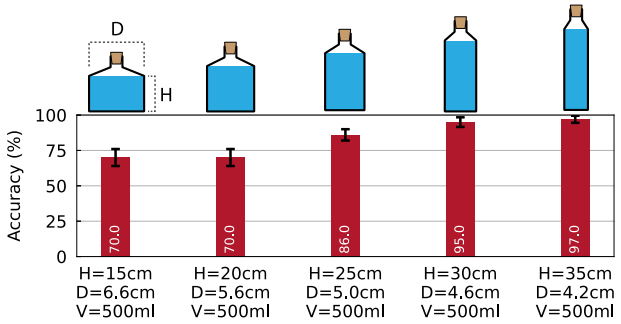

Figure 13: Figure depicts the accuracy with respect to varying bottle dimensions.

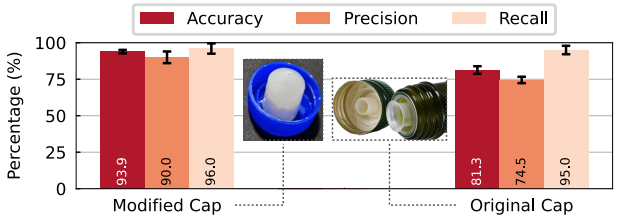

Figure 14: Figure depicts the performance of *LiquidHash* with and without our cap accessory.

observe a general trend of increasing accuracy in detecting adulteration as the concentration of adulterant increases. This is intuitively expected due to the increasing difference of liquid properties and hence difference in bubble characteristics (See § 2.1). *LiquidHash* achieves a decent accuracy of above 80% when the concentration of soybean oil and sunflower oil is above 10%, and the concentration of corn oil is above 20%. However, we also observe that *LiquidHash* has the lowest accuracy in detecting the adulteration with peanut oil. The accuracy remains below 90% even if the concentration of peanut oil increases to 50%. We attribute this to the liquid properties of peanut oil being relatively similar to those of olive oil, as explained previously in § 5.2.

5.4.2 Varying Bottle Height and Shape. Liquid food products have standard volumes (e.g., 350ml, 500ml, 750ml) but often packaged in bottles of various dimensions. To simulate this scenario, we evaluate *LiquidHash*'s performance in five 3D-printed bottles of different heights (H) and diameters (D) but a constant volume (V) of 500ml. We prepare two types of liquid (i.e., O_A and $O_A + O_{F2}$), and conduct ten trials per liquid per bottle. For each bottle, we perform cross validation using the leave-one-out approach (i.e., train on nine trials and test on the remaining trial). Figure 13 illustrates the accuracy to classify the two types of liquid. We observe a general trend of increasing accuracy as the height increases, and the diameter decreases. We propose two reasons to explain this trend. First, as the height of the bottle increases, a bubble travels a longer distance after reaching its terminal velocity. *LiquidHash* captures the bubble in a larger number of frames. Second, as the diameter of the bottle decreases, the liquid turbulence propagates through

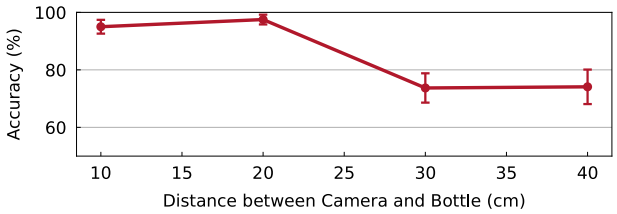

Figure 15: Figure depicts the accuracy with respect to varying the camera-to-bottle distance.

a smaller distance in the horizontal direction, and is further constrained by the bottle wall, leading to faster liquid stabilization after the rotation. *LiquidHash* selects a larger number of frames containing consistent bubbles after the pre-processing (See § 4.2.1). A larger number of *bubble features* contribute to the final decision, leading to the increase in accuracy. It is important to note that the height of the bottle needs to be large for non-viscous liquid, such as vodka, for *LiquidHash* to capture a sufficient number of frames *after* the bubbles reach the terminal velocity and the liquid stabilizes. *LiquidHash* does not work when the bottle is small for these types of liquid.

5.4.3 Varying Bottle Caps. Recall from § 4.5 that the physical design of bottle caps affects bubble generation, and regularly shaped spherical and ellipsoidal bubbles are desirable to *LiquidHash*. To demonstrate the practicality of *LiquidHash*, we evaluate the performance with the original olive oil bottle cap. Figure 14 illustrates the accuracy to classify authentic olive oil and adulteration, respectively. With the original cap, *LiquidHash* is capable of achieving an accuracy of 81.3%, a precision of 74.3% and a recall of 95.0%. However, this is still much lower than the accuracy using the modified cap with the accessory proposed in § 4.5. The original cap has an inner tube with a diameter of approximately 50mm, which is significantly larger than the diameter of 2mm in *LiquidHash*'s design. This causes the number of valid bubbles extracted per trial to be extremely small (i.e., only one to two bubbles per trial), and the velocities of bubbles are significantly affected by the turbulence in the liquid as the bubbles are most likely to rise in curved trajectories.

5.4.4 Varying Camera-to-Bottle Distance. We evaluate *LiquidHash* against changing the distance of the bottle to the camera. The camera-to-bottle distance affects the quality of bubbles captured because as distance increases, the number of pixels per bubble decreases proportionally. Figure 15 illustrates the effect of distance on the accuracy to classify authentic and fake olive oil products. *LiquidHash* achieves the highest accuracy within a distance of 20cm, but the accuracy drops below 80% when the distance is above 30cm. We attribute this performance degradation to the significant reduction in the number of pixels of both the bottle and bubbles. Specifically, as the distance increases from 20cm to 30cm, the measured width of a bottle and a bubble decreases from around 300 pixels and 25 pixels to around 200 pixels and 16 pixels, respectively. A small number of pixels do not contain sufficient image details, leading to errors in the marker positions (See § 4.2.4) and bubble radius and aspect ratio (See § 4.3.1). It is important to note that when the distance is

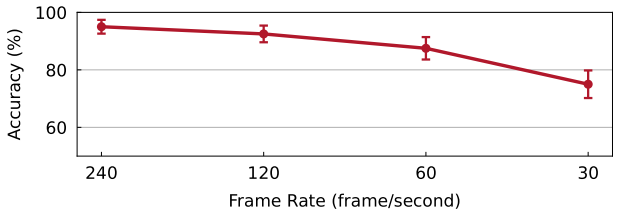

Figure 16: Figure depicts the accuracy with respect to varying video frame rates.

beyond 50cm, *LiquidHash* is not able to detect bubbles accurately as the bubble width is less than 10 pixels.

5.4.5 Varying Video Recording Frame Rate. We evaluate *LiquidHash* against different video frame rates. The video frame rate of the smartphone camera affects the capability of capturing bubble trajectories with sufficient information. This experiment simulates recording the bubbles in different smartphones with standard frame rates at 240fps, 120fps, 60fps and 30fps [1, 64] while maintaining the resolution at 1080p. Most commodity smartphones support the frame rate of 240fps, but for some low-end smartphones, such as Xiaomi Redmi Note 10S, Samsung Galaxy A12 and Vivo Y20s [22–24], only slow-motion recording of 120fps and normal video recording of 30fps and 60fps are supported. Figure 16 illustrates the effect of decreasing frame rate on the accuracy to classify both authentic and fake olive oil products. We observe a general trend of decreasing accuracy as the frame rate decreases. We attribute the performance degradation to the reduction in number of frames available to capture a bubble, hence introducing noise to extracted features (See § 4.3), and making the distinction between different liquids unclear. It is important to note that a high frame rate is required to obtain sufficient number of frames per bubble in the use case of vodka, where bubbles move extremely fast. *LiquidHash* does not work when frame rate is below 240fps in this case. On the contrary, *LiquidHash* achieves a high accuracy when the frame rate is only 30fps in the use case of honey.

6 DISCUSSION

We present discussions on *LiquidHash*'s deployment considerations and its limitations.

6.1 Deployment Considerations

Bottle cap accessory. Recall from § 4.5 that we design a bottle cap accessory to further improve the detection pipeline. We envision that this cap accessory can be attached to the original bottle cap with no modification needed to the original packaging, and can be peeled off upon opening of the bottle. To enable this, we are inspired by commodity bottle cap and lid designs. For example, many olive oil bottles ship with tube-like cap (depicted in Figure 5(b)) [3] to slow down the pouring process. Similarly, we are also inspired by the commodity peel-off lids, which is an additional layer in between the liquid content and the bottle cap [9, 47, 59]. Furthermore, we envision that more manufacturers may adopt the bottle cap accessory as there are economic benefits for manufacturers to be compliant with *LiquidHash*, which ensures liquid authenticity and may enhance consumers' trust in the products.

Extending *LiquidHash* to other applications. While we propose *LiquidHash* as an aid for the average consumers for verifying authenticity prior to purchasing liquid food products in offline stores, we envision other potential use cases. For example, different stakeholders within a product supply chain – from production, distributor, retailer, and logistics – may utilize *LiquidHash* as a proof to verify the integrity of the product.

Complementing other solutions. Many existing works propose to measure liquid properties using smartphone in-built sensors [32, 83], but requires opening the containers. *LiquidHash* could rely on these techniques to identify untrained types of adulteration after opening the bottle.

6.2 Limitations and Future Work

Experiment setup. Despite adopting ***background lighting*** in our experiments, it is not always necessary as *LiquidHash* only requires additional lighting in the use cases of honey, which is a dark-colored liquid. Furthermore, lighting conditions may affect the performance of bubble detection and segmentation. However, with more *training* bubble images under various lighting conditions, *LiquidHash* would be robust against different lighting conditions. Furthermore, we evaluate *LiquidHash* using a ***tripod*** as a proof-of-concept. However, we note that the tripod may not be required when the container has labels acting as reference markers. The relative positions of bubbles with respect to the bottle can be obtained. This can handle the case when the camera is moving. In addition, there are many other methods to rest the camera in a stationary position (e.g., resting the smartphone on a shelf).

Bottle and liquid content transparency. *LiquidHash* requires semi- or fully transparent bottles to allow observations of the bubbles. While many liquid food products satisfy this requirement, there are many products that come in opaque bottles. From our on-going explorations with many non-transparent liquid contents, we find that additional lighting (e.g., flashlight and lamps) significantly help to enable *LiquidHash* to observe bubbles. Furthermore, we envision adopting different light sources to help in observing the bubbles that may transmit through dark or opaque materials (e.g., utilizing light sources outside the visible light spectrum, such as infrared).

Powerful counterfeiters. Recall that *LiquidHash* distinguishes the adulterants based on the differences due to liquid properties. However, powerful counterfeiters may carefully mix various types of adulterants to try and match the properties with authentic liquid product. While this is possible in theory, it is an extremely difficult, time-consuming, and expensive process. On the contrary, many reported real-world incidents are attributed to economic benefits of the counterfeiters, who strive to gain profit by mixing cheap adulterants.

Measurement of liquid properties. Recall that *LiquidHash* utilizes machine learning techniques to obtain classification results. To further extend *LiquidHash*, we envision building and training an improved machine learning model to increase the detection accuracy on a smaller amount of adulteration (e.g., less than 30%), estimate the liquid properties (i.e., *viscosity*, *density*, and *surface tension*), and compensate on temperatures, by providing the ground truth values of the liquid properties measured with laboratory equipment such as a rheometer.

Impact of *LiquidHash*. *LiquidHash* demonstrates the feasibility to detect differences in liquid contents by simple *human interaction* to induce a physics phenomenon, and *video analysis* to extract useful information. Through *LiquidHash*, we aim to inspire the research community that *physical interactions* can augment computer vision and mobile sensing to uncover hidden information to solve many important and practical problems, especially with the increased processing power of mobile devices. We hope that this paper would encourage the mobile computing community to explore other directions of research that leverage physical interactions to augment sensing in mobile devices.

7 RELATED WORK

We now present related work on liquid testing and how *LiquidHash* could complement these solutions.

Laboratory Setting-based Solutions. Conventional laboratory-based techniques, such as viscometer, rheometer, refractometer and chromatography, test physical properties from liquid samples which require opening of bottles and costly equipment [40, 41, 56]. Recent works on Near-Field (NIR) and Raman spectroscopy test liquid properties without opening bottles by utilizing absorption of electromagnetic radiation to obtain chemical information of the liquid contents [15, 43, 48]. However, these solutions are not available to the general public as they also require specialized and costly equipment. On the contrary, *LiquidHash* is a pervasive and low-cost method that the general public can utilize to perform *early testing* on liquid food products, and send the detected counterfeit products to the authorities for more comprehensive testing in the laboratory setting.

Ubiquitous Sensing-based Solutions. Recent research studies techniques that utilize wireless signals to identify the liquid types utilizing interactions between liquid content and radio waves [13, 16, 26, 27, 74, 75, 79, 81, 84]. However, these solutions also require specialized equipment and setup such as large antennas which are not portable. The wireless measurement is also susceptible to environmental effect such as multi-path interference. For example, *LiquID* [13] tries to identify liquids using the liquid's electric permittivity, which would not be hindered by dark bottles or liquids as long as the container is not metallic. However, it requires a specific container and has not shown in-bottle identification abilities. *RF-EATS* [26] attaches a radio-frequency identification (RFID) tag to the bottle and utilizes the coupling effect between the liquid content and radio-frequency (RF) signals to detect counterfeit liquid food products. However, it requires stable placement of the RFID reader and small differences between how the RFID tag is placed could affect its accuracy. It could be a viable solution to complement *LiquidHash* in the cases of opaque bottles or dark-colored liquids. Another body of research utilizes commodity devices (e.g., smartphones) to test liquid properties such as surface tension and viscosity [21, 32, 33, 76, 83]. However, these solutions require opening of bottles to transfer liquid content to controlled settings, rendering the solutions impractical in our setting. For example, *CapCam* [83] uses a smartphone camera to estimate liquid surface tension by capturing the capillary waves induced by smartphone vibrations. However, it requires transferring the liquid content to a paper or plastic container for desirable capillary waves. Similarly, *Vi-Liquid* [32] utilizes a smartphone vibro-motor and accelerometer to induce and receive vibrations from liquids, to estimate liquid viscosity. It requires transferring the liquid content to a specific container to transmit strong vibration signals and optimize smartphone placement. On the contrary, as *LiquidHash* does not require the opening of bottles and transferring of liquid contents, it is more practical for consumers to verify liquid authenticity prior to purchasing liquid food products in offline stores and for different stakeholders within a product supply chain to verify the integrity of the product. After purchasing, consumers could then rely on *CapCam* and *Vi-Liquid* to identify untrained types of adulteration after opening the bottle.

8 CONCLUSION

We present *LiquidHash*, a novel counterfeit liquid food product detection system that uses a *commodity smartphone camera* to *detect* adulterated liquid products *without opening the bottles*. *LiquidHash* leverages the physical phenomenon that different liquid contents exhibit unique properties, which can be inferred by the shape and movement of the air bubbles forming inside the bottle. We evaluate the performance of *LiquidHash* with real-world experiments under varying conditions with a total of more than 500 minutes of video recording and observe an overall detection accuracy of up to 95%.

ACKNOWLEDGMENTS

This research was partially supported by grants from the Singapore Ministry of Education Academic Research Fund Tier 1 (R-252-000-B48-114) and Yonsei University Research Fund (Grant No. 2021-22-0337).

REFERENCES

[1] Aaron. 2019. Understanding Video Frame Rate. https://planpackcapture.com/understanding-video-frame-rate/

[2] Agilent. 2021. Protecting our honey against food adulteration | Agilent. https://www.agilent.com/about/features/en/honey-supplies.html

[3] Alibaba. 2021. 31-500 Black Pp Plastic Tamper Evident Pourer Caps,31.5*28mm Plastic Cap With Plastic Insert For Oil Bottle - Buy Tamper Proof Cap/31.5*24mm Olive Oil Pourer Cap For Marasca Glass Bottle Plastic Caps With Safety Ring (31.5mm),Olive Oil Pourer Cap/green Color Plastic Cap And Sealed Inner Stopper For Screw Cap Glass Bottle,Bottle Cap Pourer/plastic Olive Oil Glass Bottle Ropp Cap Aluminium Caps 31.5mm Sping Dosierer Product on Alibaba.com. https://www.alibaba.com/product-detail/31-500-Black-PP-Plastic-Tamper_60704395266.html?spm=a2700.7724857.normal_offer.d_title.5f854a0a0av5JN

[4] S. Baz-Rodríguez, A. Aguilar-Corona, and A. Soria. 2012. Rising velocity for single bubbles in pure liquids. http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-27382012000200006

[5] Liza Brown. 2021. Best Slow-Motion Cameras in 2021. https://filmora.wondershare.com/video-editing-tips/best-slow-motion-cameras.html

[6] J. G. Calvert. 1990. Glossary of atmospheric chemistry terms (Recommendations 1990). *Pure and Applied Chemistry* 62, 11 (1990), 2167–2219. https://doi.org/10.1351/pac199062112167

[7] Enrico Casadei, Enrico Valli, Filippo Panni, James Donarski, Jordina Farrús Gubern, Paolo Lucci, Lanfranco Conte, Florence Lacoste, Alain Maquet, Paul Brereton, Alessandra Bendini, and Tullia Gallina Toschi. 2021. Emerging trends in olive oil fraud and possible countermeasures. *Food Control* 124 (June 2021), 107902. https://doi.org/10.1016/j.foodcont.2021.107902

[8] Roland Clift, John R Grace, and Martin E Weber. 2005. Bubbles, drops, and particles.

[9] The Cary Company. 2021. Lift 'n' Peel™ Heat Induction Liners. https://www.thecarycompany.com/containers/closures/cap-liners/lift-n-peel

[10] NEOGEN Corporation. 2019. Food fraud: The adulterated alcohol trend. https://www.neogen.com/neocenter/blog/food-fraud-the-adulterated-alcohol-trend/?utm_medium=SocialShare

[11] Adam Cox, Ansgar Wohlschlegel, Lisa Jack, and Edward Smart. 2020. *The cost of food crime.* Technical Report. Food Standards Agency. https://doi.org/10.46756/YUOF6077

[12] Erict Culinarylore. 2021. Fake Olive Oil Brands? | CulinaryLore. https://culinarylore.com/food-myths:fake-olive-oil-brands/

[13] Ashutosh Dhekne, Mahanth Gowda, Yixuan Zhao, Haitham Hassanieh, and Romit Roy Choudhury. 2018. LiquID: A Wireless Liquid IDentifier. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services* (Munich, Germany) *(MobiSys '18)*. Association for Computing Machinery, New York, NY, USA, 442–454. https://doi.org/10.1145/3210240.3210345

[14] DiscoveringNatural. 2021. FAKE vs REAL Olive Oil | Which One Are You Using on your Natural Hair? | DiscoveringNatural. https://www.youtube.com/watch?v=ORycKDD4aA4

[15] D. I. Ellis, H. Muhamadali, Y. Xu, R. Eccles, I. Goodall, and R. Goodacre. 2019. Rapid through-container detection of fake spirits and methanol quantification with handheld Raman spectroscopy. *The Analyst* 144, 1 (2019), 324–330. https://doi.org/10.1039/C8AN01702F

[16] Chao Feng, Jie Xiong, Liqiong Chang, Ju Wang, Xiaojiang Chen, Dingyi Fang, and Zhanyong Tang. 2019. WiMi: Target Material Identification with Commodity Wi-Fi Devices. , 700–710 pages. https://doi.org/10.1109/ICDCS.2019.00075 ISSN: 2575-8411.

[17] Yoav Freund and Robert E Schapire. 1997. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. System Sci.* 55, 1 (Aug. 1997), 119–139. https://doi.org/10.1006/jcss.1997.1504

[18] Yucheng Fu and Yang Liu. 2019. BubGAN: Bubble generative adversarial networks for synthesizing realistic bubbly flow images. *Chemical Engineering Science* 204 (2019), 35–47.

[19] Sky Gidge. 2016. Resealable Brand-name Alcohol Bottles on Taobao. http://www.thatsmags.com/shenzhen/post/15319/resealable-brand-name-alcohol-bottles-on-taobao

[20] Nature's Glory. 2020. Real Honey vs Fake Honey in Malaysia. https://www.natures-glory.com/blogs/news/real-honey-vs-fake-honey-in-malaysia

[21] Nicolas-Alexandre Goy, Zakari Denis, Maxime Lavaud, Adrian Grolleau, Nicolas Dufour, Antoine Deblais, and Ulysse Delabre. 2017. Surface tension measurements with a smartphone. *The Physics Teacher* 55, 8 (Nov. 2017), 498–499. https://doi.org/10.1119/1.5008349 Publisher: American Association of Physics Teachers.

[22] GSMArena. 2021. Samsung Galaxy A12 - Full phone specifications. https://www.gsmarena.com/samsung_galaxy_a12-10604.php

[23] GSMArena. 2021. vivo Y20s - Full phone specifications. https://www.gsmarena.com/vivo_y20s-10530.php

[24] GSMArena. 2021. Xiaomi Redmi Note 10S - Full phone specifications. https://www.gsmarena.com/xiaomi_redmi_note_10s-10769.php

[25] Monica Gumulya, Jyeshtharaj B. Joshi, Ranjeet P. Utikar, Geoffrey M. Evans, and Vishnu Pareek. 2016. Bubbles in viscous liquids: Time dependent behaviour and wake characteristics. *Chemical Engineering Science* 144 (April 2016), 298–309. https://doi.org/10.1016/j.ces.2016.01.051

[26] Unsoo Ha, Junshan Leng, Alaa Khaddaj, and Fadel Adib. 2020. Food and Liquid Sensing in Practical Environments using RFIDs. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*. USENIX Association, Santa Clara, CA, 1083–1100. https://www.usenix.org/conference/nsdi20/presentation/ha

[27] Unsoo Ha, Yunfei Ma, Zexuan Zhong, Tzu-Ming Hsu, and Fadel Adib. 2018. Learning Food Quality and Safety from Wireless Stickers. In *Proceedings of the 17th ACM Workshop on Hot Topics in Networks (HotNets '18)*. Association for Computing Machinery, New York, NY, USA, 106–112. https://doi.org/10.1145/3286062.3286078

[28] Trevor Hastie, Saharon Rosset, Ji Zhu, and Hui Zou. 2009. Multi-class AdaBoost. *Statistics and Its Interface* 2, 3 (2009), 349–360. https://doi.org/10.4310/SII.2009.v2.n3.a8

[29] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. , 2961–2969 pages.

[30] Genevieve Howland. 2019. Olive Oil Scam REVEALED (And How to Spot the Real Stuff). https://www.mamanatural.com/virgin-olive-oil-scam-fraud/

[31] Bailey Hu. 2017. 13 Poisoned by Fake Alcohol in South China, 4 Sent to ICU. https://www.thatsmags.com/china/post/21534/13-poisoned-by-fake-alcohol-in-guangdong

[32] Yongzhi Huang, Kaixin Chen, Yandao Huang, Lu Wang, and Kaishun Wu. 2021. ViLiquid: Unknown Liquid Identification with Your Smartphone Vibration. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking* (New Orleans, Louisiana) *(MobiCom '21)*. Association for Computing Machinery, New York, NY, USA, 174–187. https://doi.org/10.1145/3447993.3448621

[33] Yongzhi Huang, Kaixin Chen, Lu Wang, Yinying Dong, Qianyi Huang, and Kaishun Wu. 2021. Lili: liquor quality monitoring based on light signals. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. Association for Computing Machinery, New York, NY, USA, 256–268. http://doi.org/10.1145/3447993.3483246

[34] Daniel D Joseph. 2003. Rise velocity of a spherical cap bubble. *Journal of Fluid Mechanics* 488 (2003), 213–223.

[35] Olumayowa T. Kajero, Mukhtar Abdulkadir, Lokman Abdulkareem, and Barry James Azzopardi. 2020. The Effect of Liquid Viscosity on the Rise Velocity of Taylor Bubbles in Small Diameter Bubble Column. https://doi.org/10.5772/intechopen.92754 Publication Title: Vortex Dynamics Theories and Applications.

[36] Gernot Kalchgruber. 2020. Cases of Counterfeit and Tainted Alcohol are Reaching Epidemic Levels Hurting Consumers and Damaging Legitimate Brewers, Distillers and Vintners. https://www.authenticvision.com/cases-of-counterfeit-and-tainted-alcohol-are-reaching-epidemic-levels-hurting-consumers-and-damaging-legitimate-brewers-distillers-and-vintners/

[37] Ibrahim Sadek Khattab, Farzana Bandarkar, Mohammad Amin Abolghassemi Fakhree, and Abolghasem Jouyban. 2012. Density, viscosity, and surface tension of water+ethanol mixtures from 293 to 323K. *Korean Journal of Chemical Engineering* 29, 6 (June 2012), 812–817. https://doi.org/10.1007/s11814-011-0239-6

[38] Yewon Kim and Hyungmin Park. 2021. Deep learning-based automated and universal bubble detection and mask extraction in complex two-phase flows. *Scientific reports* 11, 1 (2021), 1–11.

[39] N. Kugou, K. Ishida, and A. Yoshida. 2003. Experimental study on motion of air bubbles in seawater (terminal velocity and drug coefficient of air bubble rising in seawater). , 145–158 pages.

[40] Labcompare. 2021. Rheometers / Viscometers. https://www.labcompare.com/Laboratory-Analytical-Instruments/599-Rheometers-Viscometers/

[41] Labcompare. 2021. Viscometers: The Science of Measuring Fluid Flow. http://www.labcompare.com/10-Featured-Articles/143121-Viscometers-The-Science-of-Measuring-Fluid-Flow/

[42] P. Laplace. 1805. Trait´e de m´ecanique c´eleste.

[43] Zheng Li and Kenneth S. Suslick. 2018. A Hand-Held Optoelectronic Nose for the Identification of Liquors. *ACS Sensors* 3, 1 (Jan. 2018), 121–127. https://doi.org/10.1021/acssensors.7b00709

[44] Liu Liu, Hongjie Yan, Guojian Zhao, and Jiacai Zhuang. 2016. Experimental studies on the terminal velocity of air bubbles in water and glycerol aqueous solution. *Experimental Thermal and Fluid Science* 78 (2016), 254–265.

[45] Michael. 2013. Alcohol density chart – the most comprehensive list available. https://bartenderly.com/tips-tricks/alcohol-density-chart/

[46] Tom Mueller. 2017. The Olive Oil Scam That's Gone Global. https://goop.com/wellness/health/the-heartbreak-of-global-olive-oil-fraud-and-what-to-do-about-it/ Section: Detoxes & Cleanses.

[47] PKN Packaging News. 2018. Bottle cap solves an accessibility issue - PKN Packaging News. http://www.packagingnews.com.au/materials/bottle-cap-solves-an-accessibility-issue

[48] University of Manchester. 2017. Scientists develop counterfeit booze-detecting device. https://phys.org/news/2017-09-scientists-counterfeit-booze-detecting-device.html

[49] Rui M. Pires, Henrique F. Costa, Abel G. M. Ferreira, and Isabel M. A. Fonseca. 2007. Viscosity and Density of Water + Ethyl Acetate + Ethanol Mixtures at 298.15 and 318.15 K and Atmospheric Pressure. *Journal of Chemical &*

Engineering Data 52, 4 (2007), 1240–1245. https://doi.org/10.1021/je600565m arXiv:https://doi.org/10.1021/je600565m

[50] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* 28 (2015), 91–99.

[51] RheoSense. 2017. What is the Viscosity of ...? https://blog.rheosense.com/what-is-the-viscosity-of

[52] Cecilia Rodriguez. 2016. The Olive Oil Scam: If 80% Is Fake, Why Do You Keep Buying It? https://www.forbes.com/sites/ceciliarodriguez/2016/02/10/the-olive-oil-scam-if-80-is-fake-why-do-you-keep-buying-it/ Section: Lifestyle Old.

[53] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. arXiv:1505.04597 http://arxiv.org/abs/1505.04597

[54] SafeProof. 2017. Smirnoff Vodka's Popularity Makes it a Fake Alcohol Target - SafeProof.org. https://www.safeproof.org/fake-smirnoff-vodka/ Section: Liquor Enforcement.

[55] Shreya N. Sahasrabudhe, Veronica Rodriguez-Martinez, Meghan. O'Meara, and Brian E. Farkas. 2017. Density, viscosity, and surface tension of five vegetable oils at elevated temperatures: Measurement and modeling. , 17 pages. https://doi.org/10.1080/10942912.2017.1360905

[56] Ask A Scientist. 2019. What is Chromatography and How Does it Work? https://www.thermofisher.com/blog/ask-a-scientist/what-is-chromatography/

[57] Haydn Simpson. 2017. Food and beverages: fighting counterfeits worldwide. https://www.worldtrademarkreview.com/anti-counterfeiting/food-and-beverages-fighting-counterfeits-worldwide

[58] Deepak Singh. 2019. 10 Best Phones To Capture Slow Motion Videos In 2021 - Smartprix.com. https://www.smartprix.com/bytes/best-phones-to-capture-slow-motion-videos/

[59] Hazel Metal Packaging Solutions. 2021. Peel Off Ends | Food Can Components | Products | Welcome to Hazel Metal Packaging Solutions. https://www.hazelmetalpackaging.com/products/products-details.php?catid=1&sid=3

[60] Michael Sommers. 2020. The Real Reason Your Olive Oil Is Probably Fake. https://www.mashed.com/281801/the-real-reason-your-olive-oil-is-probably-fake/

[61] Sharon Spielman. 2020. Food fraud is not only an economic drain but also a supply chain safety concern. https://www.foodengineeringmag.com/articles/98984-food-fraud-is-not-only-an-economic-drain-but-also-a-supply-chain-safety-concern

[62] India com Lifestyle Staff. 2020. How To Check If Your Honey Is Pure Or Adulterated. https://www.india.com/lifestyle/how-to-check-if-your-honey-is-pure-or-adulterated-4240130/

[63] Denis Storey. 2020. The Real Cost of Food Fraud. https://www.tracegains.com/blog/the-real-cost-of-food-fraud

[64] StudioBinder. 2021. WATCH: What is FPS? — Frame Rates Explained (Video Essay). https://www.studiobinder.com/blog/video-frame-rate/

[65] Monica H. Swahn. 2019. Counterfeit alcohol, sometimes containing jet fuel or embalming fluid, is a growing concern for tourists abroad. https://www.pri.org/stories/2019-07-11/counterfeit-alcohol-sometimes-containing-jet-fuel-or-embalming-fluid-growing

[66] Synced. 2019. Here's To Drinking Safely With AI! | Synced. https://syncedreview.com/2019/12/21/heres-to-drinking-safely-with-ai/

[67] Alessandro Tacconelli. 2020. The king of wine counterfeiting is free and (maybe) back to business. https://authena.io/the-king-of-wine-counterfeiting-is-free-and-maybe-back-to-business/

[68] Mário A. R. Talaia. 2007. Terminal Velocity of a Bubble Rise in a Liquid Column.

[69] Paola Tamma. 2017. Honeygate: How Europe is being flooded with fake honey. https://www.euractiv.com/section/agriculture-food/news/honey-gate-how-europe-is-being-flooded-with-fake-honey/ Section: Agrifood.

[70] Engineering ToolBox. 2003. Fluid Flow - Equivalent Diameter. https://www.engineeringtoolbox.com/equivalent-diameter-d_205.html

[71] Oscar Edwin Piamba Tulcan, Danielle Oliveira De Andrade, Tavares De Andrade, and Roberto Guimarães Pereira. 2008. ANALYSIS OF PHYSICAL CHARACTERISTICS OF VEGETABLE OILS. , 4 pages.

[72] Nick Vanstone, Andrew Moore, Perry Martos, and Suresh Neethirajan. 2018. Detection of the adulteration of extra virgin olive oil by near-infrared spectroscopy and chemometric techniques. *Food Quality and Safety* 2, 4 (Dec. 2018), 189–198. https://doi.org/10.1093/fqsafe/fyy018

[73] Von Malegowski. 2013. How to Re-seal a Water Bottle. https://www.youtube.com/watch?v=abosduPhns4

[74] Ju Wang, Jie Xiong, Xiaojiang Chen, Hongbo Jiang, Rajesh Krishna Balan, and Dingyi Fang. 2017. TagScan: Simultaneous Target Imaging and Material Identification with Commodity RFID Devices. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking* (Snowbird, Utah, USA) *(MobiCom '17)*. Association for Computing Machinery, New York, NY, USA, 288–300. https://doi.org/10.1145/3117811.3117830

[75] Ju Wang, Jie Xiong, Xiaojiang Chen, Hongbo Jiang, Rajesh Krishna Balan, and Dingyi Fang. 2021. Simultaneous Material Identification and Target Imaging with Commodity RFID Devices. *IEEE Transactions on Mobile Computing* 20, 2 (2021), 739–753. https://doi.org/10.1109/TMC.2019.2946072

[76] Mengxi Wei, Shuo Huang, Jing Wang, Haihong Li, Hujiang Yang, and Shihong Wang. 2015. The study of liquid surface waves with a smartphone camera and an image recognition algorithm. *European Journal of Physics* 36, 6 (Sept. 2015), 065026. https://doi.org/10.1088/0143-0807/36/6/065026 Publisher: IOP Publishing.

[77] James O. Wilkes. 2017. Physical Properties—Density, Viscosity, and Surface Tension.

[78] Ryan Williams. 2018. Fake alcohol found in authentic bottles. https://www.redpoints.com/blog/illegal-alcohol-authentic-bottles/

[79] Binbin Xie, Jie Xiong, Xiaojiang Chen, Eugene Chai, Liyao Li, Zhanyong Tang, and Dingyi Fang. 2019. Tagtag: Material Sensing with Commodity RFID. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems* (New York, New York) *(SenSys '19)*. Association for Computing Machinery, New York, NY, USA, 338–350. https://doi.org/10.1145/3356250.3360027

[80] Peng Yan, Haibo Jin, Guangxiang He, Xiaoyan Guo, Lei Ma, Suohe Yang, and Rongyue Zhang. 2020. Numerical simulation of bubble characteristics in bubble columns with different liquid viscosities and surface tensions using a CFD-PBM coupled model. https://www.sciencedirect.com/science/article/abs/pii/S0263876219305593

[81] Hui-Shyong Yeo, Gergely Flamich, Patrick Schrempf, David Harris-Birtill, and Aaron Quigley. 2016. RadarCat: Radar Categorization for Input & Interaction. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. Association for Computing Machinery, New York, NY, USA, 833–841. https://doi.org/10.1145/2984511.2984515

[82] T. Young. 1805. An essay on the cohesion of fluids. , 65-87 pages.

[83] Shichao Yue and Dina Katabi. 2019. Liquid Testing with Your Smartphone. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services* (Seoul, Republic of Korea) *(MobiSys '19)*. Association for Computing Machinery, New York, NY, USA, 275–286. https://doi.org/10.1145/3307334.3326078

[84] Diana Zhang, Jingxian Wang, Junsu Jang, Junbo Zhang, and Swarun Kumar. 2019. On the Feasibility of Wi-Fi Based Material Sensing. In *The 25th Annual International Conference on Mobile Computing and Networking*. ACM, Los Cabos Mexico, 1–16. https://doi.org/10.1145/3300061.3345442