Abstract: Liquid state estimation is important for robotics tasks such as pouring; however, estimating the state of transparent liquids is a challenging problem. We propose a novel segmentation pipeline that can segment transparent liquids such as water. Unlike previous works, our method does not require heating up the liquid during training, observing the liquid from multiple viewpoints or time instances, checkerboard backgrounds, weight measurements, or any liquid motion. Our method can segment transparent liquids from a static, RGB image without requiring any manual annotations or heating of the liquid for training. Instead, we use a generative model that is capable of translating unpaired images of colored liquids into synthetically generated transparent liquid images. Segmentation labels of colored liquids are obtained automatically using background subtraction. We use paired samples of synthetically generated transparent liquid images and background subtraction for our segmentation pipeline. Our experiments show that we are able to accurately predict a segmentation mask for transparent liquids without requiring any manual annotations.

Keywords: Image Segmentation, Transparent Liquids, Deformable Object Manipulation

1 Introduction

Liquid segmentation is a key component to the problem of visually estimating the state of liquids like water or oil. Once state estimation is accomplished robots will be able to pour effectively. This would enable us to automate tasks such as pouring pizza sauce in kitchens, transferring soups between containers by a robotic waiter, pouring medicines into vials in pharmacies, or watering our plants. Transparent liquids are hard to perceive in images since their texture is usually a distortion of the background caused due to refraction of light. Obtaining depth measurements for liquids is also difficult since the liquid will refract the projected infrared light.

Previous works have explored robotic pouring in various settings [1, 2, 3, 4, 5]. Several methods for liquid segmentation require heating up the liquid during training to obtain ground-truth labels when viewed by a thermal camera [6, 7, 8]; however, heating up the liquid for training is a tedious process that limits how much training data can be easily collected. Other approaches require observing the liquid from multiple viewpoints [1], checkerboard backgrounds [4], weight measurements [4], or liquid motion [9, 10, 11, 12].

In this work, we propose a method for learning to segment transparent liquids such as water inside transparent containers. Our method operates on static images (we do not require liquid motion), and requires no manual annotations or heating of liquids during training. To accomplish this, our method uses a generative model that learns to translate images of colored liquid into synthetically generated images of transparent liquid. We also use a background subtraction model to predict liquid regions in an image given a sequence of empty cup images and a full cup image. We use this synthetically generated image paired with annotations from the background subtraction model to segment transparent liquids.
2 Related Work

**Transparent object perception:** Perceiving transparent objects is particularly challenging because transparent objects can refract, reflect and absorb light. Some previous work focuses on perceiving transparent containers; methods have been developed for transparent object segmentation [13, 14], depth estimation [15, 16], keypoint estimation [17], and transparent object matting [18]. Other methods for segmenting transparent objects use light field cameras [19, 20] or light polarization [21]. On the manipulation side, other recent works have been developed to directly grasp transparent objects without first estimating their 3D shape [22]. Our approach builds on [13] for transparent container segmentation; however, our focus is on segmenting the transparent liquid inside the container. Unlike some of these papers that use manual annotations for training [13], our work does not rely on any manual annotations.

**Transparent liquid perception:** Transparent liquid perception is different from transparent object perception because liquids are by nature without shape or geometry; they conform to the container surfaces and can change shape based on external forces. They surface contour can change along with their texture, reflection and refraction patterns. A number of methods have been proposed for transparent liquid perception. One approach is to use heated liquid observed by a thermal camera to obtain ground-truth labels for liquid [6, 7, 8]. However, the requirement to heat the liquid before recording the ground-truth is a tedious process; our method does not require heated liquid. To segment liquid while it is being poured, one can use optical flow [9] or audio signals [10, 11]. Our method can segment static liquid, which is important for liquid state estimation before a pouring task is initiated. Some methods reason about the refraction of the infrared light emitted by a depth sensor [1, 12]; multiple noisy readings from different viewpoints [1], or from different time points during pouring [12], integrated probabilistically. In contrast, our method can segment the liquid from just a single RGB image. Another approach is to use a depth sensor to estimate the height of the liquid surface [2]; however, such depth readings are inaccurate for transparent liquids. A different strategy is to pour liquid in front of a checkerboard background; edges in the distorted checkerboard image (as a result of light refraction through the transparent liquid) are used to estimate the liquid location [4]. These estimates are also combined with weight readings from a scale [4]. Our method does not require a checkerboard background or a scale. Finally, other papers forgo perceiving transparent liquid and focus on accurate pouring of colored liquid, which can be more easily detected using background subtraction [3].

**Image Translation:** Many methods have been developed to translate an image from a source domain into a corresponding image from the target domain. One popular approach for this task is to use generative adversarial algorithms [23], such as incorporating cycle [24] and semantic consistency constraints [25] to map unpaired images between domains. Later work [26] introduces multi-layer patch-wise contrastive learning which encourages the generative model to encode commonalities between the two image domains while being invariant to texture. We build on this work to translate unpaired images of colored liquid into images of transparent liquid.

3 Proposed Method

We describe our method for transparent liquid segmentation when liquids are placed within transparent containers. An overview of our method can be found in Figure 3. First, we collect a dataset of colored liquid and another (unpaired) dataset of transparent liquid. We then use an image translation method to learn to translate an image of colored liquid into a synthetically generated image of transparent liquid that is identical to the input image, except that the liquid is now transparent. Next, we use background subtraction to find the colored liquid pixels in the colored liquid dataset. We treat the colored liquid segmentation as ground-truth label for the synthetically generated transparent liquid. We then train a network to segment transparent liquid, using paired samples of the synthetically generated transparent liquid and colored liquid ground-truth labels.

As a pre-processing step, we use previous work to segment transparent containers in the image [13]. We then crop the image around the container segmentation (see Appendix for details); this cropped image is input to our method, described below.
Figure 1: Our method has three components; (a) uses image translation to learn to translate an image of colored liquid into a synthetically generated image of transparent liquid. Next, in (b) we use background subtraction to find the colored liquid pixels, which we treat as ground-truth labels for the synthetically generated transparent liquid. Lastly using (c), we train a network to segment transparent liquid, using synthetically generated transparent liquid image and ground truth labels from (b).

3.1 Image Translation

First, we collect one dataset of colored liquids in transparent containers; we as well as collect a second (unpaired) dataset of transparent liquids in transparent containers. Given these two datasets, we learn an image translation from colored to transparent liquids. To do so, we use CUT [26], which we train to convert an image of a colored liquid into an image of a transparent liquid (see Figure 3).

We briefly review the main elements of CUT and how we adapt it for our method. The main component of CUT is a generator that translates an image of the source domain into an image of the target domain: the generator $G$ is divided into an encoder $G_{enc}$ and a decoder $G_{dec}$, such that the output $y = G(x) = G_{dec}(G_{enc}(x))$, for an image $x$ from the source domain.

Additionally, CUT uses a patch-wise contrastive loss to encourage corresponding patches between the input and output images to be similar to each other in feature space. Specifically, given an image from the source domain $X$, the image is translated into an image of the target domain $Y$. The patch-wise contrastive loss maximizes the mutual information between $H(G_{enc}(x))$ and $H(G_{enc}(y))$, for

$$L_{GAN}(G, D, X, Y) = E_{y \sim Y} \log D(y) + E_{x \sim X} \log(1 - D(G(x)))$$ (1)
(a) (b) (c) (d) (e)

Figure 2: Background subtraction for automatic ground truth annotation of colored liquid: (a) Empty cup from which we build the background model (b) Cup filled with transparent liquid (c) Background subtraction for transparent liquid; note that this mask is not accurate and hence our method does not rely on background subtraction of transparent liquid (d) Cup filled with colored liquid (e) Background subtraction for colored liquid; this mask is much more accurate than the one in (c), hence we use this mask to train the transparent liquid segmentation model, trained on synthetically generated images of transparent liquid.

If we wish to segment transparent liquids, one might ask why we don’t apply background subtraction directly to the images of transparent liquid? As observed in Fig 2c, background subtraction is not able to detect the transparent liquid pixels between Fig 2a & Fig 2b. This is because the GMM detects if the difference from the background model exceeds a threshold. Transparent liquids do not offer sufficient visual difference in the image to exceed this threshold; reducing the threshold too...
Figure 3: Image translation from colored liquid to transparent liquid; **Top Row:** Real world colored liquid images, **Bottom Row:** Generated transparent liquid images

low will lead to noisy foreground estimation. Instead, we perform background subtraction only on images of colored liquids, which is relatively easy and accurate, as shown in Fig 2d and e.

We then convert the image of the colored liquid $x$, to an image of transparent liquid $G(x)$, using the generator described above in Section 3.1. We use the segmentation label $I_{gt}$ (obtained using background subtraction on colored liquid) as the ground-truth for the generated image $G(x)$ of transparent liquid. We assume that the location of the transparent liquid in the generated image $G(x)$ is the same as the location of the colored liquid in the image $x$; based on this assumption, we treat the segmentation label $I_{gt}$ as a ground-truth label for the generated image $G(x)$. As described in the previous section, this assumption generally holds, due to the patch-wise contrastive loss.

### 3.3 Transparent Liquid Segmentation

We train a transparent liquid segmentation model from scratch to classify liquid pixels in an image. We use paired samples of synthetic transparent liquid images and ground truth obtained from background subtraction to train the segmentation model. The segmentation model is trained using the Binary Cross Entropy loss between the predicted liquid segmentation mask and the ground truth. We use the same architecture for the segmentation network $S$ as the generator $G$ used in CUT.

Both the generator $G$ and the segmentation network $S$ take in an RGB image and down-sample it twice with convolutional layers before passing it through nine Resnet blocks latent, followed by two up-sampling deconvolutional layers. Each layer of $G$ and $S$ is normalized using a 2DConv, 2DBatchNorm layer, followed by a ReLU activation function. Other network architectural details, hyper-parameters, and implementation details are described in the appendix.

### 4 Experiments and Results

#### 4.1 Evaluation procedure

The image translation model (Section 3) is trained using just 50 images of transparent containers filled to various volumes with colored liquid, as well as 112 images of transparent containers filled
to various volumes with transparent liquid (water). We use the 50 images of colored liquid to gen-
erate 50 synthetic images of transparent liquid; we obtain a ground-truth mask using background
subtraction applied to the colored liquid and then we use these 50 images to train our segmentation
model. Note that 50 training examples is a fairly small amount for a deep-learning based segmenta-
tion model.

To evaluate our method, we create a test set of 65 images of transparent containers that have varying
amounts of transparent liquids (water) in them, placed at different distances from the camera. We
manually annotate the location of the transparent liquids for the test set; however, we do not use
such annotations for training.

4.2 Results

We show examples of the image translation in Fig 3. We observe that the network is able to translate
the input image containing green-colored water pixels into images of clear water while still capturing
the same background and refraction patterns as that in the source image. Most importantly, the
transparent liquid in the synthetic images is in the same location as the colored liquid in the original
images. This property allows us to apply the background mask from the colored liquids as the
ground-truth label for the synthetic images of transparent liquids.

Our results for transparent liquid segmentation can be found in Table 1 as well as in Fig 4 (bottom
row). Our method generally succeeds at segmenting the liquid pixels in the image. As shown in
Table 1 and in Fig 4, our method gets lower performance for images in which the cup is filled to a
high level. We theorize that such examples have regions of liquid that are far away from a liquid-cup
or liquid-air boundary; thus these cases are harder to classify when there are no refractive patterns
to indicate the presence of liquid. We leave further improvements on these cases to future work.

<table>
<thead>
<tr>
<th></th>
<th>Ablation 1</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low liquid</td>
<td>0.42</td>
<td>0.82</td>
</tr>
<tr>
<td>Medium liquid</td>
<td>0.31</td>
<td>0.83</td>
</tr>
<tr>
<td>High liquid</td>
<td>0.24</td>
<td>0.76</td>
</tr>
<tr>
<td>All</td>
<td>0.32</td>
<td>0.82</td>
</tr>
</tbody>
</table>

Table 1: Average Intersection over Union (IoU) scores on a test set of 65 transparent liquid images,
each filled with water to varying amounts. We show the performance for subsets of images with
varying amounts of liquid in the cup (low, medium, and high) as well as an average over all images.

4.3 Ablations

We perform ablations to test the importance of each component of our method and to answer the
following questions:

**Ablation: What is the benefit of training the segmentation model on synthetically generated
transparent (instead of colored) liquid?** To answer this question, we train a segmentation model
on colored liquid with color jitter and evaluate it on transparent liquid. We jitter the brightness,
contrast and hue to obtain the input image for training the segmentation model (see Appendix for
details). This ablation explores whether such color augmentation is sufficient to train a model for
transparent liquid segmentation.

The results of this ablation are shown in column 1 of Table 1, compared to our method in column 2.
The average IoU of 0.32 shows that the segmentation model trained from scratch is mostly to unable
classify liquid vs non-liquid pixels. Qualitative results of this ablation are shown in Fig 4 (3rd row).
The model mostly fails to detect the correct liquid height during our evaluation. This analysis shows
that using color jitter on a colored liquid image is not sufficient domain randomization to capture the
texture and patterns required to segment transparent liquids. Instead, the network predicts an "av-
erage" segmentation mask that fails to capture the variation in liquid height. This analysis supports
the idea that we need to train the segmentation model directly on images of transparent liquid.
5 Conclusion

In this paper, we propose a method to segment transparent liquid placed inside transparent containers using static RGB images. A generative model is used to translate colored liquids to transparent texture. We show that an encoder-decoder network can be used to predict the segmentation mask directly from RGB images of transparent liquids without using any additional input modalities. We use background subtraction on colored liquids to obtain the ground truth for training the segmentation model and we do not require any manual annotations. Our method shows good results for most test cases of transparent liquid segmentation. We hope that our method paves the way for robot pouring of transparent liquids in transparent cups.
References


