2025 IEEE/SICE International Symposium on System Integration (SII)
January 21-24, 2025, Munich, Germany

A Pipeline for Transparency Estimation of Glass and Plastic Bottle Images for Neural Scanning

Floris Erich, Jerome Susgin, Noriaki Ando, Yusuke Yoshiyasu

***Abstract*— Reduction of the gap between simulation and reality (sim2real gap) is essential for robots to learn how to manipulate objects in real scenarios. Estimation of an alpha value for transparent pixels is necessary to render novel views of common objects such as bottles and cups. While it is straightforward to estimate an alpha value for transparent objects, many practical objects have a mixture of transparent areas and opaque areas. In this paper we present a pipeline for automatically segmenting bottles into label, cap and body and estimating an alpha value for the body. We train a segmentation network (Detectron2) for the task of transparent object segmentation, based on a *bottle* dataset distilled from the PACO dataset. We combine the segmentation masks into a trimap, which is then used as input for an off-the-shelf matting deep neural network (ViTMatte). In our experiments, we show that the per-pixel error for transparent pixels can be reduced by 44% using our pipeline, compared to the baseline of not applying transparency estimation.**

I. INTRODUCTION

3D Scanning is an automation technology for reverse engineering the geometry of an object and storing it as a triangle mesh model. A mesh model is composed of vertices and edges, typically forming triangular faces. Optionally the colors of an object can be stored in its faces or in an external texture file. 3D Scanning is a topic that has been explored for decades, but some open problems still exist, such as the accurate scanning of transparent objects [1]. Most 3D scanners are *active*, i.e. they use projected light, which passes through transparent areas and is thus inaccurate. Recently, techniques such as Neural Radiance Fields (NeRF) [2] directly learn to estimate density and coloration, thus offering an efficient technique for capturing transparent objects in natural scenes. In an earlier work, we presented a method that uses NeRF for capturing objects in an artificial scanning environment [3]. The method can accurately estimate the geometry and color of opaque objects and can also estimate the geometry of transparent objects. However, estimating the coloration of transparent objects has remained an issue, as objects captured using neural scanning would use the background color as the color assigned to the triangle mesh. For transparent objects, instead of estimating a color for transparent parts, an alpha value should be estimated instead. Figure 1 highlights the difference between a binary segmented image and an image with an estimated alpha channel as estimated by our pipeline.

Fig. 1: Processing images for Neural Scanning with a black background applied to the images. Left: Binary segmentation, the background color of the scanning environment becomes part of the object. Right: Result of our transparency estimation pipeline, transparent parts are alpha blended with the image background.

II. RELATED WORK

We introduced a framework for scanning objects using a turntable and an active sensor [1]. Due to the limitations with active scanning discussed above, this framework only supported scanning of opaque objects. Another limitation of using active sensors is that they typically produce lower quality images than cameras designed for high resolution photography. We then introduced a pipeline for scanning opaque and transparent objects using NeRF [3]. Even though it could estimate the geometry of all objects, it could not accurately estimate the coloration for transparent parts of an object. We introduced a technique for allowing active scanners to accurately estimate the geometry of transparent objects, however it requires training on augmented versions of the target objects [4], whereas our approach in this paper requires no augmentations. We introduced a toolset for easily creating labeled datasets, including geometry of transparent objects [5], similar to HANDAL [6]. It can be used to capture in-the-wild data to enable depth sensors to accurately estimate the geometry of transparent objects, however it does not solve the coloration problem. We previously introduced a method for segmenting the body and cap of bottles [7],

All authors are with National Institute of Advanced Industrial Science and Technology, Japan.

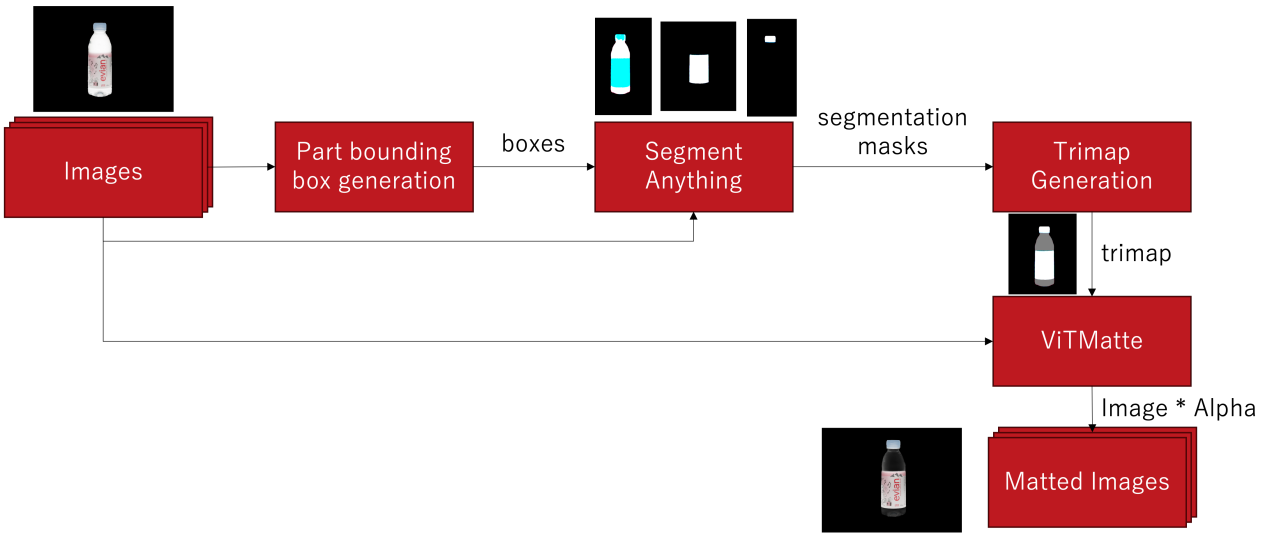

Fig. 2: Processing pipeline. We generate part-based bounding boxes using a Detectron2-based model trained on a bottle subset of PACO, and then segment these parts using Segment Anything. The segmented parts are recombined as a trimap, which is given as input to ViTMatte. Finally, matted images with proper transparency are generated.

however in this current work we solve a different problem, segmenting the bottle into opaque and transparent parts, which in practice implies segmenting bottles into cap, body and label. We explore how the generated segmentation masks can be used as an input for the ViTMatte [8] image matting model to estimate alpha values for pixels that should have transparency.

III. METHODOLOGY

We first collect images $I \in \mathbb{R}^{H \times W \times 3}$ of an object using our imaging device, which consists of a 5 camera (Canon DSLR) rigid scanning arm (Ortery MultiArm 2000) and a photo box with rotation table (Ortery PhotoBench 280) [3]. The cameras are placed in an arc, and by rotating the object the images are captured from positions approaching a hemisphere. We use the bottom three cameras in this work, as qualitatively the top two cameras did not produce good results during downstream tasks. The rotation table is turned in 20 steps, thus the total amount of images used during downstream tasks is 60.

We are interested in segmenting the object into cap, body and label. We first separate the bottle from the background using an initial segmentation step using Segment Anything (SAM) [9] with bounding boxes drawn by the user on an image which draws all images taken by one camera, as discussed in our earlier work [3]. We adopt a deep learning approach, in which a segmentation deep neural network [10] takes an image $x_{\text{img}} \in I$ and generates a segmentation map $x_{\text{segmented}} \in \mathbb{R}^{H \times W \times \text{Categories}}$. Because we are capturing images in a controlled environment, we are not interested in a segmentation network trained on a wide variety of objects, instead we want a segmentation network that is trained specifically on bottle data. Because the design of labels on bottles is highly diverse, a large dataset is necessary to train such a segmentation network, however collecting such a dataset can be expensive and time consuming. In this work we instead opted to take an existing dataset, PACO [11], and filter out any non-relevant items in order to be dedicated to the segmentation of bottle parts. Bottles in PACO are subdivided into more parts than is necessary in our case, for example instead of a single *body* category there are *upper body* and *lower body*. We define a static map and perform the conversion at inference time. Instead of directly using the segmentation maps, we generate bounding boxes and process these once more using SAM in bounding-box mode to generate binary segmentation masks $x_{\text{cap}}, x_{\text{bottle}}, x_{\text{label}} \in \mathbb{Z}_2$. In our experience this generates more accurate segmentation masks than directly using the results from the fine-tuned segmentation network.

After segmenting an object into transparent and opaque parts, we could assign a static alpha value to transparent pixels. However, we go one step further and apply a matting deep neural network (ViTMatte) [8] to automatically estimate per pixel alpha values. ViTMatte is a Vision Transformer [12] that converts a trimap $x_{\text{trimap}} \in \mathbb{Z}_3$ in combination with an image $x_{\text{img}} \in \mathbb{R}^{H \times W \times 3}$ into an image with alpha channel $x_{\text{result}} \in \mathbb{R}^{H \times W \times 4}$. The trimap consists of three regions, a background region that will automatically get alpha value 0 (fully transparent), a transparent region for which an alpha value will be estimated, and an opaque region that will automatically get alpha value 1 (fully opaque). In the case of bottle estimation, the background is assigned to the background region, the body to the transparent region and the label and cap to the opaque region.

Figure 2 gives an overview of the full processing pipeline. After processing our images using the segmentation and matting steps we are ready to use them for downstream tasks such as novel view rendering and generation of mesh models with transparency.

TABLE I: Examples of the data collected for the quantitative experiment.

Without background	Segmented	Binary combination
		
With background	**Only background**	**Combination with alpha blending**
		

TABLE II: Quantitatively evaluating our method compared to ablations

Method	Bottle	MAE	RMSE
Object SAM	Natural mineral water	64.06	70.76
	Sparkling mineral water	52.69	61.23
	Flavored mineral water	86.26	91.18
	Mean	67.67	74.39
Parts SAM	Natural mineral water	71.21	78.35
	Sparkling mineral water	77.47	86.49
	Flavored mineral water	81.17	93.69
	Mean	76.61	86.18
Object SAM + ViTMatte	Natural mineral water	62.60	72.80
	Sparkling mineral water	67.40	86.01
	Flavored mineral water	42.61	55.23
	Mean	57.54	71.35
Full	Natural mineral water	35.60	63.20
	Sparkling mineral water	48.16	61.43
	Flavored mineral water	30.34	49.93
	Mean	38.03	46.75

IV. EVALUATION

We evaluate our method on the following objects: *Natural mineral water* (containing a liquid), *Sparkling mineral water* (containing a liquid) and *Flavored mineral water* (emptied). Figure 4 shows qualitative results of our method on these objects, selecting processed images from three camera rows at a fixed interval. Our method produces matted images with good quality, but we can notice some limitations with the current system. For example, our method does not remove specular reflections and refractions from the source images. During image capture, we try to remove specular reflections by using polarization filters on both the cameras and on the light sources. Refracted light seems to confuse the transparency estimation method, causing too high estimates of transparency for pixels in which the light rays are refracted.

To estimate the accuracy of our proposed method, we created the following experimental scenario:

1) We place the object in the imaging device as usual and we capture image data.
2) We make a ghost image of the object in its start position and store it temporarily.
3) We remove the object from the imaging device, place a black background sheet and we capture image data.
4) We place the object on top of the background sheet and align it to the ghost image.

We repeat this process for all three bottles in our experimental dataset. Table I shows examples of the experimental data collected for natural mineral water. We do all of our evaluation using the images collected from the third camera, as this is the only camera in which the background covers the bottle in every frame. For each bottle we have 20 images captured by the third camera, corresponding to the number of rotation steps. The images for *Without background*, *With background* and *Only background* were directly captured by the imaging device. The image for *Segmented* was generated by manually labeling the image for *Without background* using a bounding-box and applying SAM in bounding-box mode, in this case the white background pixels have alpha value 0, while the bottle has alpha value 1. The image for *Binary combination* was generated by filling in pixels of *Only background* with the pixels of *Segmented*, where the alpha value of *Segmented* equals to 1. The image for *Combination with alpha blending* was generated by filling

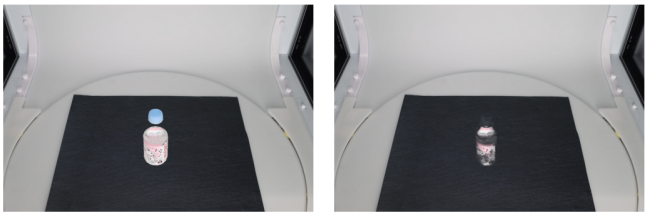

Fig. 3: Qualitative results of using other methods. Left: Part SAM, which completely ignores transparent parts. Right: Object SAM + ViTMatte, which does not discriminate between opaque parts and transparent parts for estimating the alpha value.

in pixels of *Only background* with the pixels of *Segmented*, using the formula *Combination with alpha blending* $= \alpha *$ *Segmented* $+ (1 - \alpha) *$ *Only background*, where α is the generated alpha channel using our method.

To quantitatively evaluate our method, we apply the Mean Absolute Error (MAE) metric, which is defined as $\frac{1}{N}\sum_i^N \|x_i - \hat{x}_i\|$, and the Root Mean Squared Error (RMSE) metric, which is defined as $\sqrt{\frac{1}{N}\sum_i^N (x_i - \hat{x}_i)^2}$, where N is the number of pixels, x is the estimated image and $\hat{x}$ is the ground truth image. The MAE metric is a direct measure of error, whereas the RMSE metric more significantly punishes outliers. We calculate these metrics only on areas for which a transparency value should be estimated. We use a NeRF-based labeling tool for generating ground truth segmentation masks.

Table II shows the quantitative results on the three bottles and the mean MAE and mean RMSE for all bottles. We compare our full method with the naive baselines of (a) Object SAM: only using SAM to segment the object (Table I: *Binary combination*), (b) Part SAM: using SAM to segment the object and using our part bounding box generation and a second iteration of SAM on the part bounding boxes (Left side of Fig. 3) and (3) Object SAM + ViTMatte: using SAM to segment the object and directly applying ViTMatte on the whole object (Right side of Fig. 3). Our full method uses SAM to segment the object, uses our part bounding box generation and a second iteration of SAM on the part bounding boxes and applies ViTMatte on the generated part segmentation masks. We can notice that our method (*Full*, Table I: *Combination with alpha blending*) significantly outperforms the alternative methods. Especially noteworthy is the improvement of our method over simply using object segmentation and ViTMatte, showing that separation into parts is important to generate an accurate estimate of the pixel alpha values.

The images with estimated alpha values can be used in downstream tasks such as generating meshes with transparency. By loading the original images (without an alpha channel) in NeuS2 [13] and applying the marching cubes algorithm, we can obtain a *whole bottle* mesh. By loading the matted images and applying marching cubes, we can obtain an *opaque bottle parts only* mesh, due to the density of the transparent volume being much lower than of the opaque volume. Figure 5 demonstrates how we can generate a separate mesh for the whole bottle and for the opaque parts. Figure 6 then shows how this kind of mesh data can be combined and used for rendering the bottles with transparency in software such as Blender [14].

V. DISCUSSION AND CONCLUSION

Even though our method outperforms various baselines, there are still open issues with specular reflections and refraction of light. In future research we plan to explore using deep neural networks for automatically removing specular reflections. We also plan to create a training dataset for finetuning the part segmentation network using data collected in our own environment. A limitation of our current method is that it only handles bottles and not other products that have transparency such as food products with a clear plastic wrapper. This is also a topic that we want to address in future research. Our method also relies on pretrained SAM and ViTMatte networks, which is one factor that can affect the effectiveness of the method, however as a benefit our method does not require any manual labeling apart of one object bounding box for the initial whole object segmentation. Our method also does not use the segmentation masks generated by Detectron2, instead we use generated part bounding boxes only and use the more modern SAM segmentation model. A more elegant method would integrate both models to reduce resource usage. Finally, we will use the generated transparent object models generated using our method in a robot learning scenario with a robot manipulating transparent objects.

VI. ACKNOWLEDGEMENT

This research is subsidized by New Energy and Industrial Technology Development Organization (NEDO) under a project JPNP20016. This paper is one of the achievements of joint research with and is jointly owned copyrighted material of ROBOT Industrial Basic Technology Collaborative Innovation Partnership.

REFERENCES

[1] F. Erich and N. Ando, "A Framework for 3D Scanning using RGB-D Cameras and an Automated Rotary Table," in *2022 IEEE/SICE International Symposium on System Integration (SII)*, Jan. 2022, pp. 614–619.

[2] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, Dec. 2021.

[3] F. Erich, B. Bourreau, C. K. Tan, G. Caron, Y. Yoshiyasu, and N. Ando, "Neural Scanning: Rendering and determining geometry of household objects using Neural Radiance Fields," in *2023 IEEE/SICE International Symposium on System Integration (SII)*, 2023, pp. 1–6.

[4] F. Erich, B. Leme, N. Ando, R. Hanai, and Y. Domae, "Learning depth completion of transparent objects using augmented unpaired data," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023.

[5] F. Erich, N. Chiba, A. Mustafa, Y. Yoshiyasu, N. Ando, R. Hanai, and Y. Domae, "NeuralLabeling: A versatile toolset for labeling vision datasets using Neural Radiance Fields," in *2024 IEEE/RSJ International Conference on Robots and Systems (IROS)*, 2024.

Fig. 4: Qualitative results on *Natural mineral water*, *Sparkling mineral water* and *Flavored mineral water*.

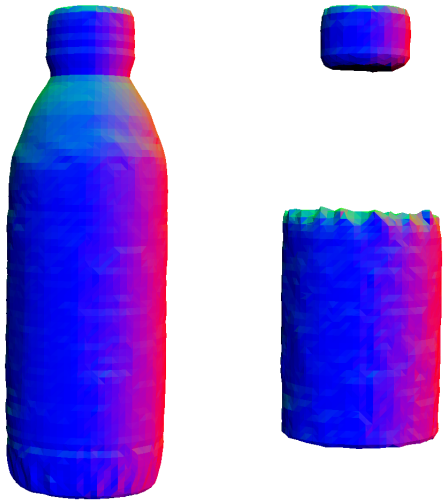

Fig. 5: Qualitative mesh export results on *Natural mineral water*. Left: Whole bottle. Right: Opaque parts.

Fig. 6: Rendering the merged meshes using Blender.

[6] A. Guo, B. Wen, J. Yuan, J. Tremblay, S. Tyree, J. Smith, and S. Birchfield, "HANDAL: A Dataset of Real-World Manipulable Object Categories with Pose Annotations, Affordances, and Reconstructions," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Detroit: IEEE, 2023.

[7] F. Erich, N. Ando, and Y. Yoshiyasu, "Scanning and Affordance Segmentation of Glass and Plastic Bottles," in *2024 IEEE/SICE International Symposium on System Integration (SII)*, Jan. 2024, pp. 514–519.

[8] J. Yao, X. Wang, S. Yang, and B. Wang, "ViTMatte: Boosting image matting with pre-trained plain vision transformers," *Information Fusion*, vol. 103, p. 102091, 2024.

[9] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment Anything," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. Paris, France: IEEE, Oct. 2023, pp. 3992–4003.

[10] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," 2019. [Online]. Available: https://github.com/facebookresearch/detectron2

[11] V. Ramanathan, A. Kalia, V. Petrovic, Y. Wen, B. Zheng, B. Guo, R. Wang, A. Marquez, R. Kovvuri, A. Kadian, A. Mousavi, Y. Song, A. Dubey, and D. Mahajan, "PACO: Parts and Attributes of Common Objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 7141–7151.

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy

[13] Y. Wang, Q. Han, M. Habermann, K. Daniilidis, C. Theobalt, and L. Liu, "NeuS2: Fast learning of neural implicit surfaces for multi-view reconstruction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[14] B. O. Community, *Blender - a 3D Modelling and Rendering Package*, Stichting Blender Foundation, Amsterdam, 2018. [Online]. Available: http://www.blender.org