Challenges of Depth Estimation for Transparent Objects*

Jean-Baptiste Weibel[1 [0000−0003−0201−4740]], Paolo Sebeto[1], Stefan Thalhammer[1 [0000−0002−0008−430X]], and Markus Vincze[1 [0000−0002−2799−491X]]

Vision for Robotics Laboratory, Automation and Control Institute, TU Wien, Austria
{weibel, sebeto, thalhammer, vincze}@acin.tuwien.ac.at

Abstract. Transparent objects and surfaces are pervasive in man-made environments and need to be considered in any vision system. Accurate depth data is a key factor for such systems reliability, requiring transparency to be inferred, due to the sensing challenges. However, the current state-of-the-art methods to predict the depth of such objects are not reliable enough to ensure safe operation of robots in arbitrary complex scenes. In order to better understand and improve upon existing solutions, we evaluate the performance of a variety of depth estimation methods. Doing so, we disentangle the different factors impacting their performance. Among our findings, neural radiance fields offer the best accuracy, but are very sensitive to the number of images used to understand the scene, and do not benefit from any level of object understanding to help them fill in the gaps.

Keywords: Transparent objects perception · Depth Estimation · Depth Completion.

1 Introduction

Vision systems need to provide sufficient information for the task and scene at hand to enable reliable and safe operations, whether in an industrial context, or when considering a service robot in a household. The COCO and LVIS [4] challenge have demonstrated the very significant progress [11,5,14] made in object detection and the robustness of such approaches to support scene understanding. An important aspect of that requirement is the ability of vision systems to reliably understand the geometry of the environment, which becomes necessary as soon as an agent is expected to act in that environment.

While widely available depth sensors have provided a solid baseline to recover the scene's geometry, they assume surfaces to be lambertian. Recovering the shape of transparent objects is therefore still an open challenge. Their appearance strongly depends on the environment in which they are observed for all wavelengths commonly used in vision sensors. Either no depth is predicted, preventing any interaction, or the depth of the transparent object's background is

* Supported by the EU-program EC Horizon 2020 for Research and Innovation under grant agreement No. 101017089, project TraceBot.

estimated, potentially leading to unsafe robot's movement in the scene. Learned methods have been introduced to address this specific problem using the color image to complete the depth [12,16,3], but their generalisation ability when encountering such transparent objects in environments with large scene shift remains to be proven.

This work presents a representative comparison of recent depth completion and NeRF methods for transparent object depth retrieval. The aim is to highlight the advantages and disadvantages of both types of approaches, and to quantitatively evaluate the expected error. We collected diverse data to empirically investigate the reliability of depth estimation methods for transparent objects. By using glass and plastic objects, filled with liquid or empty, properties like opacity and index of refraction are varied. Additionally, scene properties, such as viewing angle, object arrangements and lighting are varied to create diverse evaluation scenarios. In order to provide ground truth depth for evaluation, objects are coated and scanned with a high-quality sensor for creating $3D$ models. Image sets for testing are captured using the uncoated objects. The $6D$ pose annotations are created by registering the object models against the captured images. Ground truth depth is computed using these annotations. Evaluations on these data are provided for a set of methods including monocular depth estimation [15], RGB-D transparent object depth estimation [12,16,3], and neural radiance fields [6] as illustrated in Figure 1. In summary, our contributions are:

- an in-depth evaluation of the performance of depth estimation methods for transparent objects on common ground.
- a set of principles and recommendations inferred from the requirements of the individual approaches.

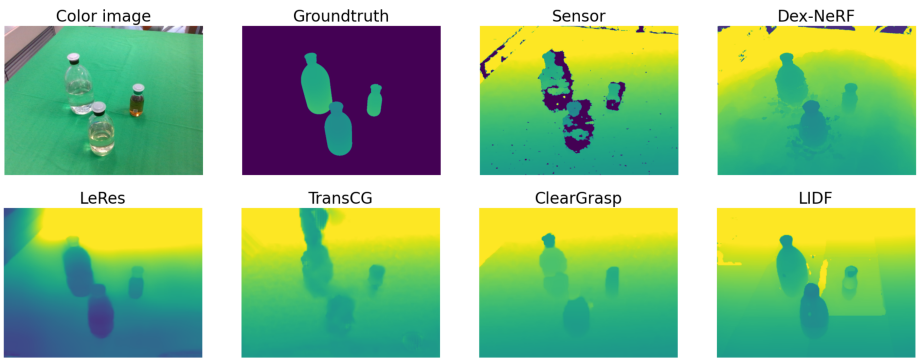

Fig. 1. Visual comparison. Depth prediction examples of the evaluated depth estimation approaches, the sensor and the ground truth.

We now present relevant state-of-the-art methods in Section 2, our proposed evaluation scheme in Section 3, our experimental results in Section 4 before presenting our conclusions in Section 5.

2 Related Works

This section introduces the state-of-the-art methods in the field of depth estimation as well as other relevant datasets of transparent objects.

2.1 Depth estimation

While active depth sensors such as the Microsoft Kinect, the Asus XTion and the Intel Realsense are widely available and provide accurate dense depth maps, none of them provide reliable depth information for transparent objects. Structured light ones tend to provide no depth at all, while active stereo can recover a few points on the edges. In the worst case, such sensor will provide depth values corresponding to the surface behind the transparent object, leading to potentially unsafe motion for a robot.

Another approach is to directly predict the depth from an RGB image using a learned method [1,15]. A major challenge in this context is to predict metric depth, as a single image does not provide information about the absolute scale. In [15], the authors split the task in two steps: first they predict monocular depth, second, they refine the scale and a focal length. A point cloud is created using the initial guess of the camera intrinsics and the estimated depth map and fed to a module that predicts that refinement.

Depth Completion: A few works focused on completing the missing depth maps produced by depth sensors using information from the corresponding RGB image. The first one to do so, ClearGrasp [12], proposed to predict a mask and surface normals of transparent objets, as well as their outline. From this information, an optimisation step would fill in the gap of the sensor depth map. LIDF [16] introduces a new local neural representation from ray-voxels pairs, and use this representation to predict the occupancy of said voxels from which the depth can be inferred. TransCG [3], on the other hand, proposes a more standard but very efficient convolutional neural network designed for depth completion.

Neural Radiance Fields: Neural Radiance fields [9], or NeRF for short, introduced a method to generate novel views of a scene from a set of posed views by learning an implicit representation. A multi-layer perceptron learns to predict density values and emitted colors for every position and direction within the scene the field represents. They are in turn used in a volume rendering scheme to recreate views of the scene. While this process originally took many hours to train on a single scene, improvements introduced by Instant-NGP [10] reduced this to less than 15 minutes. This speed-up is the result of a more efficient position encoding using a multi-resolution hash encoding combined with more efficient architectures.

In Dex-NeRF [6], the authors noticed that the density values learned by NeRF present small local maxima along rays passing through transparent objects. By setting a low threshold, the distance to the first density value crossing that threshold along the ray is shown to produce depth estimation for transparent objects. In the follow-up work Evo-NeRF [8], optimisation are made to the pipeline to learn the implicit representation faster and predict grasp points directly from a Dex-NeRF predicted depth map. Neural radiance fields have demonstrated their ability to recover scene geometry, even for transparent objects, and are very actively researched.

Transparent Objects Dataset Transparent object handling has gained considerable momentum, leading to the creation of a number of datasets of varying complexities. Cleargrasp [12] introduced its own synthetic dataset as well as a very small scale real dataset for testing purposes (286 images). LIDF [16] introduced the Omniverse large-scale synthetic dataset but no real counterpart. TransCG [3] introduced a large-scale real dataset focused on bin picking setups, mostly using top views of scenes. Most recently, [7] introduced a dataset comparing different depth sensors, including active and passive stereo, and direct and indirect time-of-flight sensors. This work does include transparent objects but is not focused on them.

The most relevant dataset to our benchmark task, ClearPose [2], introduced a large-scale dataset of scenes containing transparent objects. The annotations are obtained by placing object models in the 3D scene and hand-adjusting their positions based on the reprojection over multiple images. We use a very similar approach that was introduced in 3D-DAT [13], which uses the reconstruction of NeRF to auto-align objects with the scenes. However, ClearPose only considers transparent object with similar index of refraction, without filling. This work utilises more diverse data for evaluation, featuring different materials, containers with and without liquid filling. This allow us to reason about specific properties of objects, as detailed in the Section 3.

3 Measuring the Quality of Estimated Depth maps

This section introduces the methodology to evaluate the quality of depth estimation from a variety of methods, specifically looking into the viewpoint from which scenes are observed, object properties, and scene properties such as background and lighting. Formally, this work is concerned with predicting depth data I_D from a single or multiple color image inputs I_C, and optionally, a sensed and incomplete depth map $\hat{I}_D$. The previous section summarizes methods suited for this task. We choose a representative subset and evaluate them in comparable settings.

3.1 Depth Retrieval in Comparison

Generally, the clear advantages of depth completion approaches [12,16,3,15] is that depth can be predicted from single images without corresponding cam-

Table 1. Overview of compared methods. This table presents the inputs and inference parameters for each of the compared ones. For NeRF-based methods, † indicate results obtained using an Nvidia RTX 2070 Max-Q.

Method	Multi-view	Input	Inference time (seconds)	Inference resolution
ClearGrasp [12]	✗	RGB-D	4.8	640 × 360
LIDF [16]	✗	RGB-D	0.25	320 × 240
TransCG [3]	✗	RGB-D	0.16	320 × 240
LeRes [15]	✗	RGB	0.06	320 × 240
LeRes [15] (scaled)	✗	RGB	0.06	448 × 448
Dex-Nerf [6]	✓	RGB	14.3/6.25†	1280 × 720
Dex-Nerf [6] (half)	✓	RGB		
Dex-NGP [6,10]	✓	RGB	360/219† (training)	
NeRF [9] (Expected)	✓	RGB		

era pose. This allows broader deployment. Disadvantages are that these approaches need to be trained, and thus inherently contain a bias with respect to the training data, and require preparation time before deployment. NeRF-based approaches [6,10,9] are unbiased with respect to the scene, since models are trained directly on the observed data of the scene of interest. This however, requires availability of multiple views and corresponding camera poses. We present an overview of the input and inference speed (together with the inference resolution) in Table 1. Time were measured using a Nvidia 1080Ti and an Intel i7-7700K, unless mentioned otherwise. We also report timings on a Nvidia RTX 2070 Max-Q for NeRF-based methods since the ray-tracing cores provide significant speed-ups. For those methods, as training has to be performed on every scene, we also report training time. In order to compare the depth retrieval error on a common ground we create a test dataset of transparent objects. The following section outlines this process. The dataset is illustrated in Figure 2.

Fig. 2. Dataset images. Visualised are different scenes setups and viewing angles.

3.2 Data Collection

The dataset is collected by moving a camera attached to a robot arm around a scene. The same viewpoints are collected for every scene, and the camera poses are obtained through inverse kinematics of the robot arm. We use 3D-DAT [13] for annotation, placing object models in the virtual 3D scene, and manually correcting their poses based on their reprojection error in the different RGB views.

To obtain 3D object models, the physical objects are coated using a mat spray paint after collecting the different scenes. A high-quality depth sensor (Photoneo MotionCam-3D scanner[1]) is used to reconstruct them. The set of objects used in our experiments is illustrated in Figure 3, and includes plastics and glass objects, filled or empty with a variety of shapes, and a variety of sizes.

A total of 23 scenes is collected using a Intel Realsense D435, saving both the RGB image and the depth image at a resolution of 1280×720 pixels. The robotic arm performs a circular motion around the scene with the camera oriented toward the scene center, placing the camera at four different heights and corresponding polar angles (68°, 60°, 48° and 33°). For each circle, either 16 or 26 views are collecting resulting in a total of 64 or 104 views per scene. The light is uniform and comes from the top of the scene. For four scenes, we add a strong light projector to the side of the scene, producing caustics and other refraction and reflection effects at the interface of transparent objects. Those scenes also have more textured backgrounds, as opposed to the uniform background of the others.

Fig. 3. Object set. Objects used for evaluating depth estimation. Properties are diversified with respect to size, shape, material and filling.

3.3 Evaluation Methodology

We report the same metrics as the ones presented in ClearGrasp [12]. In particular, with GT_p the grountruth depth at pixel p and D_p^m the depth predicted by method m at pixel p, we consider δ_T the percentage of pixels having a relative depth prediction falling within a threshold T, that is, for P the set of pixels considered:

$$\delta_T = \frac{1}{P} \sum_{p \in P} \begin{cases} 1 & if \max\left(\frac{GT_p}{D_p^m}, \frac{D_p^m}{GT_p}\right) < T \\ 0 & otherwise \end{cases}$$

[1] https://www.photoneo.com

The threshold considered are 5%, 10%, 25%.

With $E_p = GT_p - D_p^m$ the difference at pixel p between the groundtruth depth and the depth predicted by method m, we also report the root mean square error $RMSE = \sqrt{\frac{1}{P}\sum_{p \in P} E_p^2}$, and the mean absolute error $MAE = \frac{1}{P}\sum_{p \in P} |E_p|$.

For all metrics and evaluations only pixels lying on object surfaces are considered, excluding any scene pixel. To provide a common basis for evaluation, given that different methods produce depth maps of different ratios and different sizes, all predicted depths are rescaled using bilinear interpolation, and crop them to fit the $\frac{4}{3}$ ratio (final resolution of 960×720 pixels).

We evaluate Cleargrasp [12], LIDF [16] and TransCG [3], all designed for depth completion for transparent objects from RGB-D pairs. The pre-trained model is used together with the default parameters. In addition to using the results obtained with the original depth sensor capturing the scene as baseline (Intel Realsense D435), the depth maps produced by a monocular depth estimation method (LeRes [15]) are also evaluated. Since LeRes does not have access to any scale information, the results on metric depth prediction are predictably poor. As such, we propose to rescale the predicted depth using the median value of all the ratio $\frac{GT_p}{D_p^{LeRes}}$, giving a sizeable boost to the results and enabling the evaluation of the shape predicted. Except in Table 2, only the rescaled results are presented. Finally, we also present the results of depth maps generated by Neural Radiance Fields. Such approaches use all RGB images of the scene during training, and need to be trained on a per-scene basis, but do not require any depth measurement. Multiple strategies can be used to extract depth maps once trained. We evaluate the results when using the expected depth value obtained from the volume rendering procedure [9], as well as Dex-NeRF depth rendering. To provide a more complete view, we also report the results obtained when training Dex-NeRF with only half of the views collected, meaning that the method never saw half of the views it is evaluated on. We also report the result when combining Dex-NeRF depth rendering with Instant-NGP [10], a neural radiance field using a different position encoding for faster training, that we refer to as Dex-NGP.

4 Results and Findings

We now present our findings regarding the behavior of depth estimation methods. A summary of the results is presented in Table 2. We notice that, while accuracy is underwhelming, all depth completion methods improve over the sensor output, but Dex-NeRF-based methods present the most accurate results, as long as enough views are available. Training with half of the views recorded, that is 32/52 views, seems indeed to be too few views for accurate reconstruction. Indeed RMSE is more strongly affected by incomplete results (missing depth) than the δ_T, which favors ClearGrasp, LIDF and TransCG that all produce "complete" output. We only observe a modest improvement for depth completion methods in $\delta_{1.05}$ values (corresponding to points within a 2.5cm error at

50cm, or 5cm at 1m) over the sensor output. TransCG, in particular, does not produce very accurate results but its prediction error remains within smaller bounds than others, showing the lowest RMSE value. LIDF produces the most accurate results out of the depth completion methods, that is the highest $\delta_{1.05}$. Finally, LeRes is surprisingly competitive after re-scaling, which is probably due to its access to a much larger training set. It should be emphasized that LeRes predicts a depth from a single RGB image, and is not designed for transparent object depth prediction but general monocular depth prediction. The fact that we use a single scale factor for the entire image suggest that the method is very good at predicting the shape of transparent objects, although not in a metric way.

Table 2. Depth estimation comparison. Results are compared using different metrics. Number are averaged over objects and scenes.

Method	$\delta_{1.05}\uparrow$	$\delta_{1.10}\uparrow$	$\delta_{1.25}\uparrow$	MAE↓	RMSE↓
Sensor (D435)	43.4	57.7	66.1	0.204	0.343
ClearGrasp [12]	46.3	68.1	87.7	0.073	0.109
LIDF [16]	49.7	72.8	92.0	0.055	0.092
TransCG [3]	43.9	68.5	89.5	0.057	**0.071**
LeRes [15]	3.8	7.3	17.3	0.582	0.652
LeRes [15] (scaled)	22.2	36.9	61.5	0.161	0.218
Dex-Nerf [6]	57.5	80.5	**92.6**	0.058	0.088
Dex-Nerf [6] (half)	33.9	51.7	61.3	0.194	0.216
Dex-NGP [6,10]	**78.0**	**86.5**	91.8	**0.054**	0.136
NeRF [9] (Expected)	24.9	51.4	85.7	0.098	0.141

4.1 Impact of the Viewing Angle

As our scenes are captured at four different camera angles, relative to the vertical orientation, Figure 4 presents the evaluation for each angle. TransCG, and to a lesser extent, Cleargrasp and LIDF, improve the closer the camera gets to a top view. This is a direct manifestation of the training data bias, as TransCG was designed with bin-picking applications in mind. This underlines that none of these learned methods gained a true understanding of transparent objects but are bound by the quality of their training data.

4.2 Impact of the Object Properties

We evaluated the impact of different object properties on the depth estimation methods. We hypothesize, that the more the object impacts the trajectory of the light through refraction and reflection, the more visible it will be. In other words, the stronger the difference of the index of refraction of adjacent mediums and the thicker the medium, the easier it should be to infer the object shape.

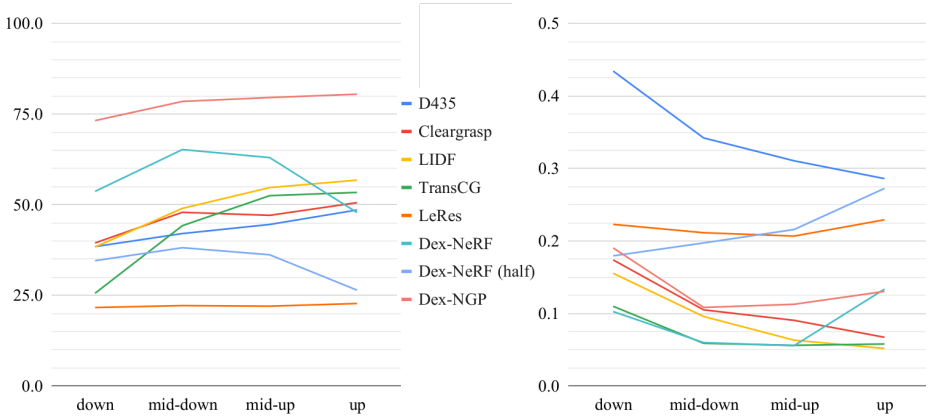

Fig. 4. Influence of the scene viewing angle. Left: Percentage of pixels with predicted depths within 5% of groundtruth ($\delta_{1.05}$). Right: RMSE

The size of transparent objects has a noticeable impact on the results, as illustrated in Figure 5. We believe this is a subtle manifestation of the impact of the object on the path of the light. Indeed, the strongest light effects happen at the border of objects, while the center tends to be much less noticeable. Larger objects will naturally have a smaller ratio of "border" pixels relative to all the pixels their silhouette covers. This overall trend is slightly weaker for depth completion methods, which can be explained by the fact that these methods benefit from an object prior, having been exposed to many object types during their training. This contrasts with NeRF-based methods that are trained from scratch for every scene. We did not however notice any strong dependency on the material (plastic or glass), or the thickness of the transparent object surfaces. We hypothesize that these effects did not manifest themselves due to the entanglement of the different properties of the chosen objects.

4.3 Impact of the Scene Lighting

In Table 3, the results for different types of scenes are reported. As described in Section 3, the additional directional light on the side of the scene leads to stronger refraction and reflection, and the additional scene texture should make the transparent object's distortion of the light path easier to distinguish. These scene changes are positively affecting the RMSE of NeRF-approaches. Conversely, this is detrimental to ClearGrasp and LIDF, but not to TransCG.

ClearGrasp and LIDF are trained from rendered data. It is still challenging to model all the effects of transparent objects using rendering methods. We hypothesize that this is the cause of the bad depth prediction of ClearGrasp and LIDF reflected in the result.

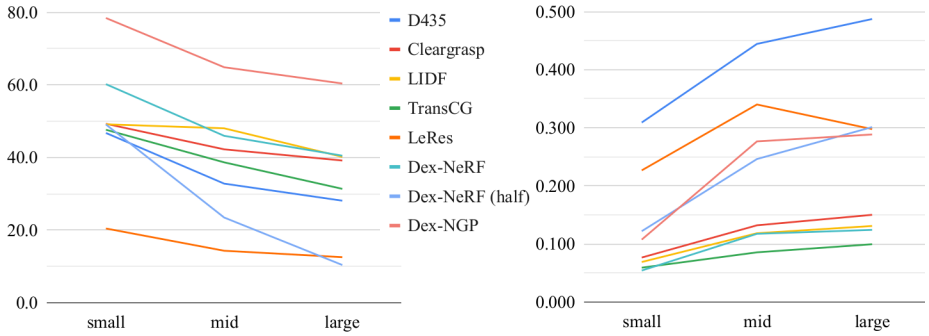

Fig. 5. Influence of the object size. Left: Percentage of pixels with predicted depths within 5% of groundtruth ($\delta_{1.05}$). Right: RMSE

Table 3. Influence of lighting and background First two rows: Percentage of pixels with predicted depths within 5% of groundtruth ($\delta_{1.05}$) per method. Higher is better. Last two rows: RMSE per method.

Scene	D435	Clearg.	LIDF	TransCG	LeRes	Dex-NeRF	Dex-NeRF (h.)	Dex-NGP
$\delta_{1.05}$								
Uniform	43.4	48.3	51.9	42.6	20.2	59.3	22.4	**76.2**
Proj.+Clut.	46.0	38.1	39.1	48.0	28.2	57.3	54.5	**86.1**
RMSE								
Uniform	0.357	0.118	0.097	**0.076**	0.243	0.108	0.297	0.161
Proj.+Clut.	0.305	0.097	0.084	0.055	0.137	0.044	0.081	**0.043**

4.4 Discussion

We now succintly present the main issues facing transparent object depth prediction. As for any learning problem, data is the key to good performance. The training dataset of [3] does not cover every part of the viewing sphere equally and rendered data created as part of [12,16] does not accurately model every light effects induced by transparent objects. This latter statement should however be continuously revised under the light of the progress made in the very active field of computer graphics. Depth completion methods provide more complete but less accurate depth maps, and can do so from a single view, with very short inference time. They indeed benefit from a level of understanding of object shapes implicitly learned during training that helps them to be more robust to varying object's size. Building on the surprisingly good results from [15], larger dataset with high variety seem essential to improve these approaches. The fairly simple but effective architecture presented in [3] also questions the need for architectures designed specifically for transparent objects, as opposed to the more general problem of depth completion.

On the opposite end of the spectrum, NeRF-based methods are the most accurate, and circumvent the issue of training data bias as they perform trans-

ductive learning. They also can provide basic guarantees about their convergence. Indeed, as they are designed to render views of the scene, comparing their current rendering to the captured images let us quickly identify the accuracy of the renderings in their close vicinity. Radiance fields are a very recent research direction, and significant progress have already been made in convergence speed, and more are expected. The modeling of transparent objects in [6] is quite simple, and more advanced modeling of light propagation within the learned volume could yield significant improvement in the quality of the recovered geometry, not only for transparent objects, but any scene with complex materials.

5 Conclusion

In this study, we evaluated a large variety of methods for estimating the depth of transparent objects. This includes depth completion methods tailored for transparent objects, general monocular depth estimation methods, as well as neural radiance fields-based methods. We demonstrated that, when possible, NeRF-based methods provide the most accurate results. These methods do not have priors about the scene, which is generally good, but detrimental when it comes to larger objects, where depth far from the border has to be inferred from plausible object shapes. Furthermore, we underlined once again the importance of a high quality, as bias-free as possible, training dataset.

Future work will disambiguate influencing effects on the reconstruction quality more in depth, i.e. clutter, lighting, object fill level, as well as the effect of transparent objects occluding other transparent objects. More detailed analysis on the influence of data distribution shifts will follow, by training depth estimation methods on real and on rendered data, and evaluating the reconstruction quality on transparent objects unseen during training. Furthermore, a natural next step for this work is to investigate the suitability of the depth estimated in the context of down-stream tasks such as object pose estimation. Such an evaluation would help underline how accurately shapes are preserved, as opposed to pixel-level evaluation.

References

1. Chen, W., Fu, Z., Yang, D., Deng, J.: Single-image depth perception in the wild. In: Proceedings of the 30th International Conference on Neural Information Processing Systems. p. 730–738. NIPS'16, Curran Associates Inc., Red Hook, NY, USA (2016)
2. Chen, X., Zhang, H., Yu, Z., Opipari, A., Jenkins, O.C.: Clearpose: Large-scale transparent object dataset and benchmark. In: European Conference on Computer Vision (2022)
3. Fang, H., Fang, H.S., Xu, S., Lu, C.: Transcg: A large-scale real-world dataset for transparent object depth completion and a grasping baseline. IEEE Robotics and Automation Letters **7**(3), 7383–7390 (2022). https://doi.org/10.1109/LRA.2022.3183256

4. Gupta, A., Dollar, P., Girshick, R.: LVIS: A dataset for large vocabulary instance segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2019)
5. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
6. Ichnowski*, J., Avigal*, Y., Kerr, J., Goldberg, K.: Dex-NeRF: Using a neural radiance field to grasp transparent objects. In: Conference on Robot Learning (CoRL) (2021)
7. Jung, H., Ruhkamp, P., Zhai, G., Brasch, N., Li, Y., Verdie, Y., Song, J., Zhou, Y., Armagan, A., Ilic, S., Leonardis, A., Navab, N., Busam, B.: On the importance of accurate geometry data for dense 3d vision tasks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 780–791 (June 2023)
8. Kerr, J., Fu, L., Huang, H., Avigal, Y., Tancik, M., Ichnowski, J., Kanazawa, A., Goldberg, K.: Evo-nerf: Evolving nerf for sequential robot grasping of transparent objects. In: Liu, K., Kulic, D., Ichnowski, J. (eds.) Proceedings of The 6th Conference on Robot Learning. Proceedings of Machine Learning Research, vol. 205, pp. 353–367. PMLR (14–18 Dec 2023), https://proceedings.mlr.press/v205/kerr23a.html
9. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 405–421. Springer International Publishing, Cham (2020)
10. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Trans. Graph. **41**(4) (jul 2022). https://doi.org/10.1145/3528223.3530127, https://doi.org/10.1145/3528223.3530127
11. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
12. Sajjan, S., Moore, M., Pan, M., Nagaraja, G., Lee, J., Zeng, A., Song, S.: Clear grasp: 3d shape estimation of transparent objects for manipulation. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 3634–3642 (2020). https://doi.org/10.1109/ICRA40945.2020.9197518
13. Suchi, M., Neuberger, B., Salykov, A., Weibel, J.B., Patten, T., Vincze, M.: 3d-dat: 3d-dataset annotation toolkit for robotic vision. In: 2023 IEEE International Conference on Robotics and Automation (ICRA) (2023)
14. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9627–9636 (2019)
15. Yin, W., Zhang, J., Wang, O., Niklaus, S., Mai, L., Chen, S., Shen, C.: Learning to recover 3d scene shape from a single image. In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. (CVPR) (2021)
16. Zhu, L., Mousavian, A., Xiang, Y., Mazhar, H., Eenbergen, J.v., Debnath, S., Fox, D.: Rgb-d local implicit function for depth completion of transparent objects. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4647–4656 (2021). https://doi.org/10.1109/CVPR46437.2021.00462