Noname manuscript No.
(will be inserted by the editor)

Learning Transparent Object Matting

Guanying Chen* · **Kai Han*** · **Kwan-Yee K. Wong**

Received: date / Accepted: date

Abstract This paper addresses the problem of image matting for transparent objects. Existing approaches often require tedious capturing procedures and long processing time, which limit their practical use. In this paper, we formulate transparent object matting as a refractive flow estimation problem, and propose a deep learning framework, called *TOM-Net*, for learning the refractive flow. Our framework comprises two parts, namely a multi-scale encoder-decoder network for producing a coarse prediction, and a residual network for refinement. At test time, TOM-Net takes a single image as input, and outputs a matte (consisting of an object mask, an attenuation mask and a refractive flow field) in a fast feed-forward pass. As no off-the-shelf dataset is available for transparent object matting, we create a large-scale synthetic dataset consisting of $178K$ images of transparent objects rendered in front of images sampled from the Microsoft COCO dataset. We also capture a real dataset consisting of 876 samples using 14 transparent objects and 60 background images. Besides, we show that our method can be easily extended to handle the cases where a trimap or a background image is available. Promising experimental results have been achieved on both synthetic and real data, which clearly demonstrate the effectiveness of our approach.

Guanying Chen*
The University of Hong Kong, Hong Kong, China
E-mail: gychen@cs.hku.hk

Kai Han*
University of Oxford, Oxford, United Kingdom
E-mail: khan@robots.ox.ac.uk
(* indicates equal contribution)

Kwan-Yee K. Wong
The University of Hong Kong, Hong Kong, China

Keywords transparent object · image matting · convolutional neural network

1 Introduction

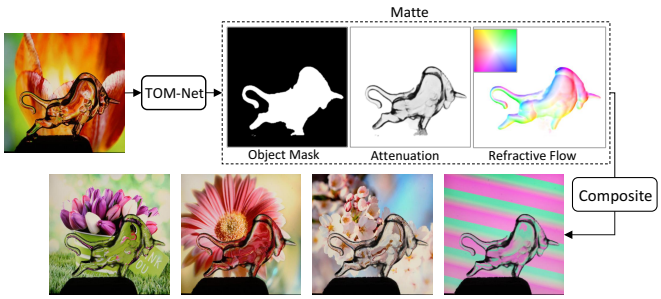


Fig. 1 Given an image of a transparent object as input, our model can estimate the environment matte (consisting of an object mask, an attenuation mask and a refractive flow field) in a feed-forward pass. The transparent object can then be composited onto new background images with the extracted matte.

Image matting refers to the process of extracting the foreground matte of an image by locating the region of the foreground object and estimating the opacity of each pixel inside the foreground region. The foreground object can then be composited onto a new background image using the *matting equation* [22]

$$C = F + (1 - \alpha)B, \quad \alpha \in [0, 1], \tag{1}$$

where C denotes the composited color, F the foreground color, B the background color, and α the opacity.

Image matting has been widely used in image editing and film production. However, most of the existing

methods are tailored for opaque objects, and cannot handle transparent objects whose appearance depends on how light is refracted from the background.

To model the effect of refraction, Zongker *et al.* [28] introduced *environment matting* as

$$C = F + (1 - \alpha)B + \Phi, \quad \alpha \in [0, 1], \tag{2}$$

where Φ is the contribution of environment light caused by refraction or reflection at the foreground object. Besides estimating the foreground shape, environment matting also describes how objects interact with the background.

Many efforts [4,24,17,27,8,6] have been devoted to improving the seminal work of [28]. The resulting methods often require either a huge number of input images to achieve a higher accuracy, or specially designed patterns to reduce the number of required images. They are in general all very computational expensive.

In this paper, we focus on environment matting for transparent objects. It is highly ill-posed, if not impossible, to estimate an accurate environment matte for transparent objects from a single image with an arbitrary background. Given the huge solution space, there exist multiple objects and backgrounds which can produce the same refractive effect. In order to make the problem more tractable, we simplify our problem to estimating an environment matte that can produce visually realistic refractive effect from a single image, instead of estimating a highly accurate refractive flow. We define the environment matte in our model as a triplet consisting of an object mask, an attenuation mask and a refractive flow field. Realistic refractive effect can then be obtained by compositing the transparent object onto new background images (see Fig. 1). We then show that the performance of the proposed method can be improved when a trimap or a background image is available.

Inspired by the great successes of convolutional neural networks (CNNs) in high-level computer vision tasks, we propose a convolutional neural network, called TOM-Net, for simultaneous learning of an object mask, an attenuation mask and a refractive flow field from a single image with an arbitrary background. The key contributions of this paper can be summarized as follows:

- We introduce a simple and efficient model for transparent object matting as simultaneous estimation of an object mask, an attenuation mask and a refractive flow field.
- We propose a convolutional neural network, TOM-Net, to learn an environment matte of a transparent object from a single image. To the best of our knowledge, TOM-Net is the first CNN that is capable of learning transparent object matting.
- We create a large-scale synthetic dataset and a real dataset as a benchmark for learning transparent object matting. Our TOM-Net has produced promising results on both the synthetic and real datasets.
- We propose two convolutional neural networks, denoted as TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$, for handling the cases where a trimap or a background image is available, respectively.

A preliminary version of this work appeared in [2]. This paper extends [2] in several aspects. First, we provide a more comprehensive comparison between our method and previous methods. Second, we present a more detailed ablation study, more experimental results, as well as analysis for failure cases. Third, we showcase an interesting application of image editing of transparent objects by manipulating the extracted environment matte. Fourth, we investigate how the performance of our method can be improved when a trimap or a background image is available. Last, we discuss in detail the limitations and potential extensions of the current model. In particular, we introduce a potential formulation for handling colored objects with specular highlights.

The remainder of this paper is organized as follows. Section 2 briefly reviews existing methods for environment matting and recent CNN based methods for image matting. Section 3 introduces a simplified environment matting formulation for transparent object matting from a single image. Section 4 describes the proposed two-stage framework and learning details. Section 5 presents our synthetic and real dataset. Experimental results on both synthetic and real dataset are shown in Section 6. Limitations and potential extensions are discussed in Section 7, followed by conclusions in Section 8.

Our code, trained model and datasets can be found at https://guanyingc.github.io/TOM-Net.

2 Related Work

In this section, we briefly review representative works on environment matting and recent works on CNN based image matting.

Environment matting Zongker *et al.* [28] introduced the concept of environment matting, and assumed each foreground pixel being originated from a single rectangular region of the background. They obtained the environment matte by identifying the corresponding background region for each foreground pixel using three

Table 1 Comparison of different environment matting methods. k indicates the image size and mapping type stands for how a foreground point is composited by the point(s) in the background image.

Methods	Asymptotic # images	# images ($k = 1024$)	Typical runtime ($k = 1024$)	Mapping type	Materials	Remarks
Ours	$O(1)$	1	0.5 secs when $k = 512$ (run on a GPU)	single-pixel	colorless, specularly refractive	aims for visually realistic effect
RTCEM [4]	$O(1)$	1	2 mins	single-pixel	colorless, specularly refractive	requires a coded background and off-line processing
Yeung et. al [26]	$O(1)$	1	30 secs	single-pixel	colored refractive	requires human interaction, aims for visually realistic effect
Zongker *et al.* [28]	$O(\log k)$	20	20 mins when $k = 512$	single-region	colored refractive, translucent, highly specular	assumes rectangular support region
Chuang *et al.* [4]	$O(k)$	1800	not available	multi-region	Zongker *et al.* [28] + (color dispersion, multiple mapping, glossy reflection)	requires solving a complex optimization problem
Wavelet [23]	$O(k)$	2400	12 hours	multi-region	same as Chuang *et al.* [4]	runtime includes data acquisition
Frequency [27]	$O(k)$	4096	$5 - 10$ mins	multi-pixel	Zongker *et al.* [28] + (color dispersion, glossy reflection)	slow data acquisition
Duan *et al.* [7]	$O(s \log(k^2/s))$	340	2.8 mins	multi-region	same as Chuang *et al.* [4]	s denotes the sparsity of a signal
Qian *et al.* [18]	$O(s \log(2k/s))$	400	3.3 mins	multi-pixel	same as Frequency [27]	s denotes the sparsity of a signal

monitors and multiple images. Chuang *et al.* [4] extended [28] in two ways. First, they replaced the single rectangular supporting area for a foreground pixel with multiple 2D oriented Gaussian strips. This makes it possible for their method to model the effects of color dispersion, multiple mapping and glossy reflection. Second, they simplified the environment matting equation by assuming the object being colorless and perfectly transparent. This allows them to achieve real time capture environment matting (RTCEM). The environment matte was then extracted with one image taken in front of a pre-designed pattern. However, RTCEM requires background images to segment the transparent objects, and depends on a time-consuming off-line processing. Wexler *et al.* [24] introduced a probabilistic model based method which assumes each background point has a probability to make contribution towards the color of a certain foreground point. Their approach does not require pre-designed patterns during data acquisition, but it still needs multiple images and can only model thin transparent objects. Peers and Dutré [17] used a large number of wavelet basis backgrounds to obtain the environment matte, and their method can also model the effect of diffuse reflection. Based on the fact that a signal can be decomposed uniquely in the frequency domain, Zhu and Yang [27] proposed a frequency-based approach to extract an accurate environment matte. They used Fourier analysis to solve the decomposition problem. Both [17] and [27] require a large number of images to extract the matte (e.g., [17] needs $2,400$ images and [27] needs $4,096$ images for an image of size 1024×1024), making them not very practical. Recently, compressive sensing theory has been applied to environment matting to reduce the number of images required. Duan *et al.* [7] applied this theory in the spatial domain and Qian *et al.* [18] applied it in the frequency domain. However, the number of images needed is still in the order of hundreds. In contrast, our work can estimate an environment matte from a single image in a fast feed-forward computation without the need for pre-designed patterns or additional background images.

Yeung *et al.* [26] proposed an interactive way to estimate an environment matte given an image containing a transparent object. Their method requires users to manually mark the foreground and background in the image, and models the refractive effect using a thin-plate-spline transformation. Their method does not produce an accurate environment matte, but instead a visually pleasing refractive effect. Our method shares the same spirit, but does not involve any human interaction.

Tab. 1 shows a comparison of different environment matting methods. Compared with other methods, our method requires only a single image and can extract a matte in 0.5 second without the need for any predefined backgrounds.

CNN based image matting Although the potential of CNN on transparent object matting has not yet been explored, some existing work have adopted CNNs for solving the general image matting problem. Shen *et al.* [20] introduced a CNN for image matting of color portrait images. Cho *et al.* [3] proposed a network to predict a better alpha matte by taking the matting results of the traditional method and normalized color images as input. Xu *et al.* [25] introduced a deep learning framework that can estimate an alpha matte from an image and its trimap. However, none of these methods can be applied directly to the task of transparent object matting as object opacity alone is not sufficient to model the refractive effect.

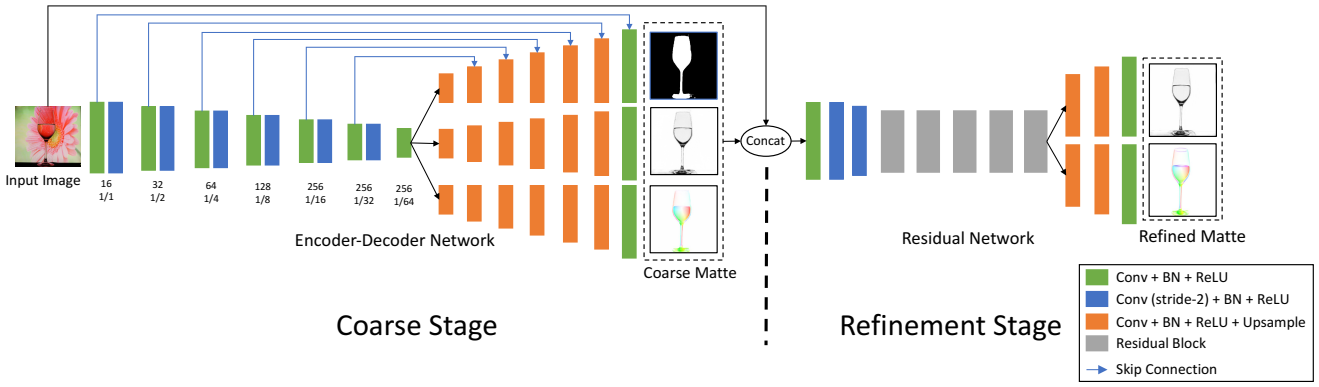


Fig. 2 TOM-Net architecture. The left subnetwork is the CoarseNet and the right subnetwork is the RefineNet. (Cross-link and multi-scale outputs are not shown for simplicity.)

3 Matting Formulation

As a transparent object may have multiple optical properties (e.g., color attenuation, translucency and reflection), estimating an accurate environment matte for a generic transparent object from a single image is very challenging. Following the work of [4], we cast environment matting to a refractive flow estimation problem by assuming that each foreground pixel only originates from one point in the background due to refraction. Compared to the seminal work of [28], which models each foreground pixel as a linear combination of a patch in the background, our formulation is more tractable and can be easily encoded using a CNN.

In [28], the per-pixel environment matting is obtained through leveraging color information from multiple background images. Given a set of pre-designed background patterns, matting is formulated as

$$C = F + (1-\alpha)B + \sum_{i=1}^{k} R_i \mathcal{M}(\mathbf{T}_i, \mathbf{A}_i), \tag{3}$$

where F, B and α denote the ambient illumination, background color and opacity, respectively. The last term in (3) accounts for the environment light accumulated from k pre-designed background images ($k = 3$ in [28]). R_i is a factor describing the contribution of light emanating from the i-th background image $\mathbf{T}_i$. $\mathcal{M}(\mathbf{T}_i, \mathbf{A}_i)$ denotes the average color of a rectangular region $\mathbf{A}_i$ on the background image $\mathbf{T}_i$.

To obtain an environment matte, the transparent object is placed in front of the monitor(s), and multiple pictures of the object are captured with the monitor(s) displaying different background patterns[1]. Generally, a surface point receives light from multiple directions, especially for a diffuse surface. When it comes to a perfectly transparent object, however, a surface point will only receive light from one direction as determined by the law of refraction. Consider a single background image as the only light source (i.e., no ambient illumination), the problem can be modeled as

$$C = (1-\alpha)B + R\mathcal{M}(\mathbf{T}, P), \tag{4}$$

where $\mathcal{M}(\mathbf{T}, P)$ is a bilinear sampling operation at location P on the background image $\mathbf{T}$. Further, by assuming a colorless transparent object, R becomes a light attenuation index ρ (a scalar value). The formulation in (4) can be simplified to

$$C = (1-\alpha)B + \rho\mathcal{M}(\mathbf{T}, P), \tag{5}$$

where $\rho \in [0, 1]$ denotes the attenuation index.

Here, we use refractive flow to model the refractive effect of a transparent object. The refractive flow of a foreground pixel is defined as the offset between the foreground pixel and its refraction correspondence on the background image.

We further introduce a binary foreground mask to define the object region in the image. The matting equation can now be rewritten as

$$C = (1-m)B + m\rho\mathcal{M}(\mathbf{T}, P), \tag{6}$$

where $m \in \{0, 1\}$ denotes background ($m = 0$) or foreground ($m = 1$). The matte can then be estimated by solving m, ρ and P for each pixel in the input image containing the transparent object[2].

[1] For an image of size 512×512, 18 pictures and around 20 minutes processing time are needed.

[2] For an image with n pixel, we have 7 unknowns (3 for B, 2 for P, 1 for m, and 1 for ρ) for each pixel, resulting in a total of $7n$ unknowns.

4 Learning Transparent Object Matting

In this section, we present a two-stage deep learning framework, called TOM-Net, for learning transparent object matting (see Fig. 2). The first stage, denoted as CoarseNet, is a multi-scale encoder-decoder network that takes a single image as input, and predicts an object mask, an attenuation mask and a refractive flow field simultaneously. CoarseNet is capable of predicting a robust object mask. However, the estimated attenuation mask and refractive flow field lack local structural details. To overcome this problem, we introduce the second stage of TOM-Net, denoted as RefineNet, to achieve a sharper attenuation mask and a more detailed refractive flow field. RefineNet is a residual network [11] that takes both the input image and the output of CoarseNet as input. After training, our TOM-Net can predict an environment matte from a single image in a fast feed-forward pass.

4.1 Encoder-Decoder for Coarse Prediction

The first stage of our TOM-Net (i.e., CoarseNet) is based on mirror-link CNN introduced in [21]. Mirror-link CNN was proposed to learn non-lambertian object intrinsic decomposition. Its output consists of an albedo map, a shading map and a specular map. It shares a similar output structure with our transparent object matting task (i.e., three output branches sharing the same spatial dimensionality). Therefore, it is reasonable for us to adapt mirror-link CNN for our CoarseNet.

The mirror-link CNN adapted for our CoarseNet consists of one shared encoder and three distinct decoders. The encoder contains six down-sampling convolutional blocks, leading to a down-sampling factor of 64 in the bottleneck layer. Features in the encoder layers are connected to the decoder layers having the same spatial dimensions through skip connections [19]. Cross-links [21] are introduced to make different decoders share the same input in each layer, so that decoders can better utilize the correlation between different predictions.

Learning with multi-scale loss has been proven to be helpful in dense prediction tasks (e.g., [9,10]). Since we formulate the problem of transparent object matting as refractive flow estimation, which is a dense prediction task, we augment our mirror-link CNN with multi-scale loss similar to [10]. We use four different scales in our model, where the first scale starts from the decoder features with a down-sampling factor of 8 and the largest scale has the same spatial dimensions as the input.

In contrast to the recent two-stage framework for image matting [25], our TOM-Net has a shared encoder and three parallel decoders to accommodate different outputs. Besides, we augment our CoarseNet with multi-scale loss and cross-link. Moreover, TOM-Net is trained from scratch while the encoder in [25] is initialized with the pre-trained VGG16.

4.2 Loss Function for Coarse Stage

CoarseNet takes a single image as input and predicts the environment matte as a triplet consisting of an object mask, an attenuation mask and a refractive flow field. The learning of CoarseNet is supervised by the ground-truth matte using an object mask segmentation loss $\mathcal{L}_{ms}$, an attenuation regression loss $\mathcal{L}_{ar}$, and a refractive flow regression loss $\mathcal{L}_{fr}$. Besides, the predicted matte is expected to render an image as close to the input image as possible when applied to the ground-truth background. Hence, in addition to the supervision of the matte, we also take image reconstruction loss $\mathcal{L}_{ir}$ into account. Note that the ground-truth background is only used to calculate the reconstruction error during training but not needed during testing. CoarseNet can therefore be trained by minimizing

$$\mathcal{L}^c = \alpha^c_{ms}\mathcal{L}_{ms} + \alpha^c_{ar}\mathcal{L}_{ar} + \alpha^c_{fr}\mathcal{L}_{fr} + \alpha^c_{ir}\mathcal{L}_{ir}, \tag{7}$$

where $\alpha^c_{ms}, \alpha^c_{ar}, \alpha^c_{fr}, \alpha^c_{ir}$ are weights for the corresponding loss terms.

Object mask segmentation loss Object mask segmentation is simply a spatial binary classification problem. The output of the object mask decoder has a dimension of $2 \times H \times W$, where H and W denote the height and width of the input. We normalize the output with *softmax* and compute the loss using the binary cross-entropy function

$$\mathcal{L}_{ms} = -\frac{1}{HW}\sum_{ij}(\tilde{M}_{ij}\log(P_{ij}) + (1-\tilde{M}_{ij})\log(1-P_{ij})), \tag{8}$$

where $\tilde{M}_{ij} \in \{0,1\}$ and $P_{ij} \in [0,1]$ represent ground truth and normalized foreground probability of the pixel at (i,j), respectively.

Attenuation regression loss The predicted attenuation mask has a dimension of $1 \times H \times W$. The value of this mask is in the range of $[0,1]$, where 0 indicates no light can pass and 1 indicates the light will not be attenuated. We adopt a mean square error (MSE) loss

$$\mathcal{L}_{ar} = \frac{1}{HW}\sum_{ij}(A_{ij} - \tilde{A}_{ij})^2, \tag{9}$$

where A_{ij} is the predicted attenuation index and $\tilde{A}_{ij}$ the ground truth at (i,j).

Refractive flow regression loss The predicted refractive flow field has a dimension of $2 \times H \times W$, where we have one channel for the horizontal displacement and another for the vertical displacement. We normalize the refractive flow with *tanh* activation and multiply it by the width of the input, such that the output is constrained in the range of $[-W, W]$. We adopt an average end-point error (EPE) loss

$$\mathcal{L}_{fr} = \frac{1}{HW} \sum_{ij} \sqrt{(F_{ij}^x - \tilde{F}_{ij}^x)^2 + (F_{ij}^y - \tilde{F}_{ij}^y)^2}, \tag{10}$$

where (F^x, F^y) and $(\tilde{F}^x, \tilde{F}^y)$ denote the predicted flow and the ground truth, respectively.

Image reconstruction loss We use MSE loss to measure the dissimilarity between the reconstructed image and the input image. Denoting the reconstructed image by I and the ground-truth image (i.e., the input image) by $\tilde{I}$, the reconstruction loss is given by

$$\mathcal{L}_{ir} = \frac{1}{HW} \sum_{ij} \|I_{ij} - \tilde{I}_{ij}\|_2^2. \tag{11}$$

Implementation details In all experiments, we empirically set $\alpha_{ms}^c = 0.1, \alpha_{ar}^c = 1, \alpha_{fr}^c = 0.01$, and $\alpha_{ir}^c = 1$. The loss weights for different scales are $\frac{1}{2^{(4-s)}}$, where $s \in \{1, 2, 3, 4\}$ denotes the scale. CoarseNet contains $8M$ parameters and it takes about 2.5 days to train with Adam optimizer [14] on a single NVIDIA Titan X Pascal GPU. We first train the CoarseNet from scratch until convergence and then train the RefineNet.

4.3 Residual Learning for Matte Refinement

As the attenuation mask and the refractive flow field predicted by the CoarseNet lack structural details, a refinement stage is needed to produce a detailed matte. Observing that residual learning is particularly suitable for tasks whose input and output are largely similar [13,16], we propose a residual network, denoted as RefineNet, to refine the matte predicted by the CoarseNet. Similar strategy has also been successfully applied to progressively refine the estimated optical flow in [12].

We concatenate the input image and the output of the CoarseNet to form the input of the RefineNet. As the object mask predicted by the CoarseNet is already plausible, the RefineNet only outputs an attenuation mask and a refractive flow field. The parameters of the CoarseNet are fixed when training the refinement stage.

Loss for the refinement stage The overall loss for the refinement stage is

$$\mathcal{L}^r = \alpha_{ar}^r \mathcal{L}_{ar} + \alpha_{fr}^r \mathcal{L}_{fr}, \tag{12}$$

where $\mathcal{L}_{ar}$ is the refinement attenuation regression loss, $\mathcal{L}_{fr}$ the refinement flow regression loss, and α_{ar}^r, α_{fr}^r their weights. The definitions of these two losses are identical to those defined in the first stage. We found that adding the image reconstruction loss in the refinement stage did reduce the image reconstruction error during training, but was not helpful in preserving sharp edges of the refractive flow field (e.g., mouth of a glass). This could be explained by the fact that a lower image reconstruction loss does not guarantee a better refractive flow field. As the matte estimated by the CoarseNet has already achieved a small reconstruction error, simultaneously optimizing the flow regression loss and image reconstruction loss in the refinement stage may compromise the flow estimation. Since our goal in the refinement stage is to estimate a more detailed matte, we remove the image reconstruction loss to make our network focus on reducing the flow regression loss.

Implementation details We set $\alpha_{ar}^r = 1$, $\alpha_{fr}^r = 1$ for the refinement. RefineNet contains $1M$ parameters and it takes about 2 days to train with Adam optimizer on a single NVIDIA Titan X Pascal GPU. RefineNet is randomly initialized during training.

4.4 Improvement with Trimap and Background Image

As the problem of transparent object matting from a single image is highly ill-posed, we investigate how to reinforce our framework by utilizing additional information. In particular, we consider the cases where a trimap or a background image is available. Our framework can be easily extended to make use of these additional information by taking the concatenation of the input image and the background image (or trimap) as input, while keeping the overall network architecture unchanged.

TOM-Net$^{+\text{Trimap}}$ Trimap can provide a rough location of the transparent object to help the model better locate the transparent object. The trimap used in this paper is a single channel image with 3 different values, where values 0, 1, and 2 indicate background, unknown, and foreground regions, respectively. During training, we randomly generate trimaps based on the ground-truth object mask. We first perform random erosion and cropping on the object mask to form the known (rough) foreground region. The unknown region is then generated by subtracting the foreground region from a

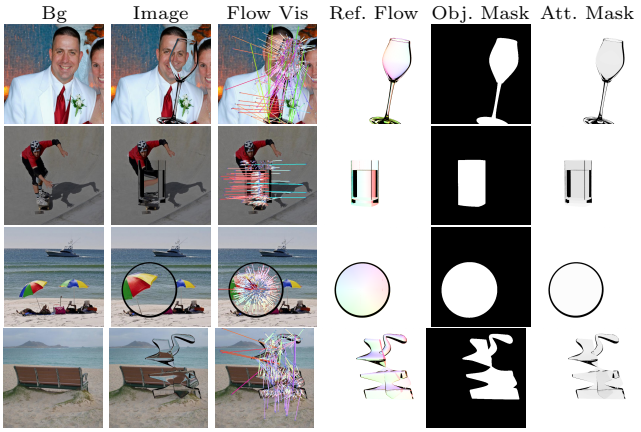


Fig. 3 Examples of synthetic data. Up to down: examples of glass, glass with water, lens and complex shape, respectively. First three columns: background image, rendered image, refractive flow visualization (sparse). Last three columns: ground-truth refractive flow field, object mask, attenuation mask. (Best viewed in PDF with zoom.)

tight bounding box of the object mask, leaving the rest of the regions as the background region. The variant model, denoted as TOM-Net$^{+\text{Trimap}}$, takes both the input image and trimap as input, giving rise to an input channel number of 4 in the first convolutional layer.

TOM-Net$^{+\text{Bg}}$ Given the background image, the model can easily identify the accurate location of the transparent object based on the difference of the input and background images. Moreover, having access to the background image allows the model to better estimate the refractive flow field. The variant model, denoted as TOM-Net$^{+\text{Bg}}$, takes both the input and background images as input, giving rise to an input channel number of 6 in the first convolutional layer.

TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ are trained with the same procedure as TOM-Net. Our experimental results show that with the additional information, our framework can achieve better results on both synthetic and real dataset.

5 Dataset for Learning and Evaluation

As no off-the-shelf dataset for transparent object matting is available, and it is very tedious and difficult to produce a large real dataset with ground-truth object masks, attenuation masks and refractive flow fields, we created a large-scale synthetic dataset by using *POV-Ray* [1] to render images of synthetic transparent objects. Besides, we also captured a real dataset for evaluation. We will show that our TOM-Net trained on the synthetic dataset can generalize well to real world objects, demonstrating its good transferability.

Table 2 Statistics of our synthetic datasets.

Type	Glass	Glass & Water	Lens	Complex	Total
Synthetic Train	$52K$	$26K$	$20K$	$80K$	$178K$
Synthetic Test	250	250	200	200	900

5.1 Synthetic Dataset

We used a large number of background images and 3D models to render our training samples. We randomly changed the pose of the models, as well as the viewpoint and focal length of the camera in the rendering process to avoid overfitting to a fixed setting.

Background images We employed two types of background images, namely scene images and synthetic patterns. For scene images, we randomly sampled images from the Microsoft COCO [15] dataset[3]. The background images for the synthetic training set are sampled from COCO Train2014 and Test2015, while that for the synthetic test dataset are from COCO Val2014, giving rise to $100K$ scene images in total. For synthetic patterns, we rendered $40K$ patterns of size 512×512 using *POV-Ray* built-in textures.

Transparent objects We divided common transparent objects into four categories, namely glass, glass with water, lens, and complex shape (see Fig. 3 for examples). We constructed parametric 3D models for the first three categories, and generated a large number of models using random parameters. For complex shapes, we constructed parametric 3D models for basic shapes like sweeping-spheres and squashed surface of revolution (SOR) parts, and composed a larger number of models using these basic shapes. We generated $178K$ 3D models in total, with each model assigned a random refractive index $\lambda \in [1.3, 1.5]$. The distribution of these models in four categories is shown in Tab. 2.

Ground-truth matte generation We obtained the ground-truth object mask of a model by rendering it in front of a black background image and setting its color to white. Similarly, we obtained the ground-truth attenuation mask of a model by simply rendering it in front of a white background image. Finally, we obtained the ground-truth refractive flow field (see Fig. 3) of a model by rendering it in front of a sequence of Gray-coded patterns. Technical details for the data rendering can be found at https://github.com/guanyingc/TOM-Net_Rendering

[3] Other large-scale datasets like ImageNet [5] can also be used.

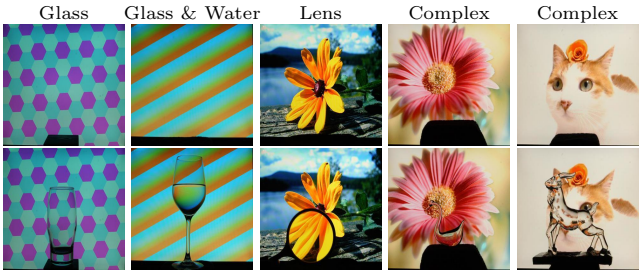


Fig. 4 Sample images in real dataset. The first row shows the background images and the second row shows the images of transparent objects.

Table 3 Statistics of our real dataset. The first and second rows show the number of objects and the number of backgrounds used during data acquisition, respectively. The last row shows the number of captured samples. Note that the category of glass with water are created by filling five of the glasses with different amount of water, and some backgrounds are shared between different shape categories.

	Glass	Glass & Water	Lens	Complex
# Objects	7	(5 glasses used)	1	6
# Backgrounds	60	38	4	18
# Samples	470	103	61	242

Data augmentation To improve the diversity of the training data and narrow the gap between real and synthetic data, extensive data augmentation was carried out on-the-fly. For an image of size 512×512 with color intensity normalized to $[0, 1]$, we randomly performed color (brightness, contrast and saturation) augmentation (in a range of $[-0.2, 0.2]$), image scaling (in a range of $[0.875, 1.05]$), noise perturbation (in a range of $[-0.05, 0.05]$), and horizontal/vertical flipping. Besides, we also blurred the object boundary to make the synthetic data visually more natural. A patch with a size of 448×448 was then randomly cropped from an augmented image and used as input to train CoarseNet. To speed up the training and save memory, a smaller patch with a size of 384×384 was used to train RefineNet after the training of CoarseNet.

5.2 Real Dataset

To validate the transferability of TOM-Net, we introduce a real dataset, which was captured using 14 objects[4] and 60 background images, resulting in a dataset of 876 images. Note that the background images for real data have not been used in the synthetic training or test dataset. The data distribution is summarized in Tab. 3. During the data capturing process, the objects were placed under different poses, with the distances between the camera, object and background uncontrolled. Fig. 4 shows some sample images from the real dataset. Note that we do not have the ground-truth matte for the real dataset. We instead captured images of the backgrounds without the transparent objects to facilitate evaluation.

[4] The objects consist of 7 glasses, 1 lens and 6 complex objects. Glasses with water are implicitly included.

Table 4 Ablation study. F, A, I, and M are short for flow, attenuation, image reconstruction, and object mask, respectively. (The first value for EPE is measured on the whole image and the second measured within the object region. A-MSE and I-MSE are computed on the whole image.)

MSE ($\cdot 10^{-2}$)	↓ better	↑ better		
ID	Model Variants	F-EPE	A-MSE	I-MSE	M-IoU
0	Background	6.5 / 41.0	1.58	0.87	0.15
1	CoarseNet - ($\mathcal{L}^c_{fr}$)	3.9 / 26.5	0.24	0.23	0.98
2	CoarseNet - (cross-link)	2.5 / 17.2	0.30	0.21	0.97
3	CoarseNet - (multi-scale)	2.4 / 16.6	0.69	0.25	0.94
4	CoarseNet - ($\mathcal{L}^c_{ir}$)	2.3 / 15.7	0.25	0.22	0.98
5	CoarseNet	**2.2** / **15.4**	0.28	0.18	0.97
6	CoarseNet + RefineNet	**2.0** / **13.7**	0.25	0.19	0.97
7	CoarseNet + (RefineNet+$\mathcal{L}^r_{ir}$)	2.0 / 13.9	0.24	0.18	0.97

6 Experiments and Results

In this section, we present experimental results and analysis. We performed ablation study for TOM-Net, and evaluated our approach on both synthetic and real data. For synthetic data, we evaluated end-point error (EPE) for refractive flow fields, intersection over union (IoU) for object masks, mean square error (MSE) for attenuation masks and image reconstruction results, respectively. For real data, due to the absence of ground-truth matte, evaluation on the absolute error with respect to the ground truth is not possible. Instead, we reconstructed the input images using the estimated mattes and background images, and then evaluated the PSNR and SSIM metrics [23] between each pair of input image (i.e., photograph) and reconstructed image (i.e., composite). In addition, a user study was conducted to validate the realism of TOM-Net composites.

We showcased an application of image editing of transparent object by manipulating the extracted matte, and analyzed typical failure cases. We also investigated how the performance of our method can be improved when a trimap or a background image is available.

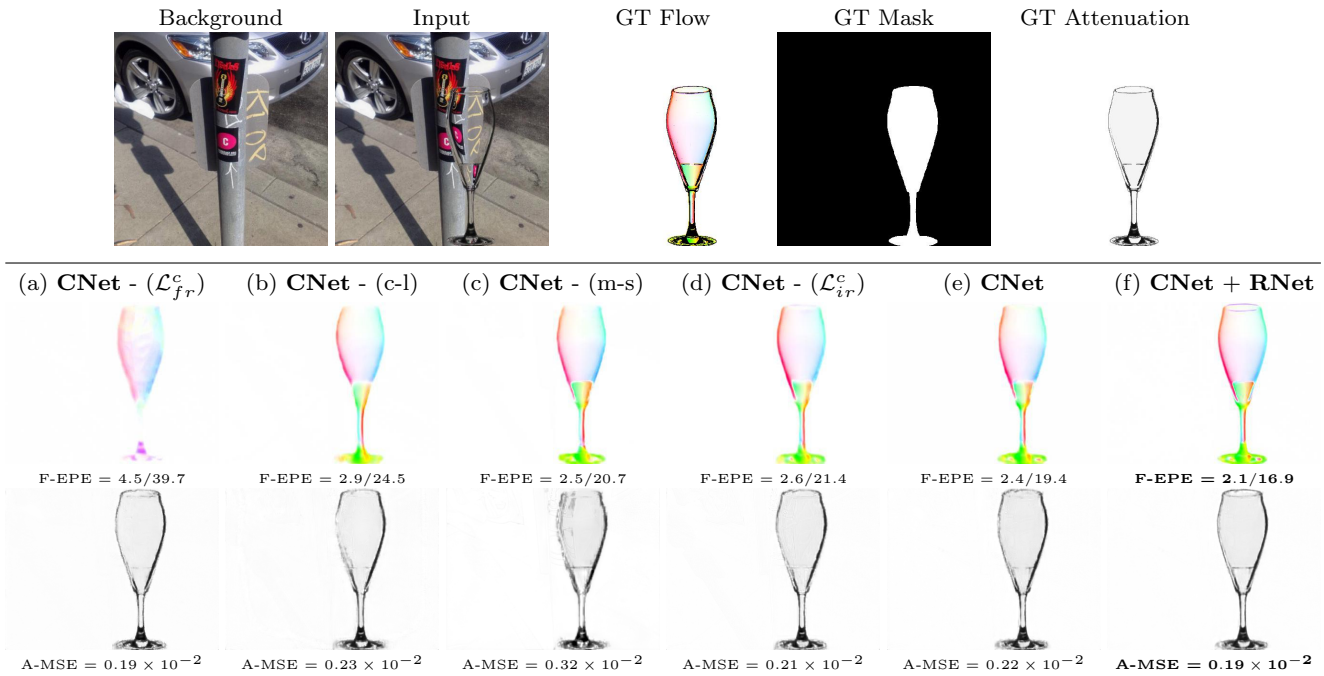


Fig. 5 Qualitative comparison of different model variants in ablation study. The first row shows a sample of *glass with water* from the synthetic test dataset. The second and third rows show the estimated refractive flow fields and attenuation masks by different variants, respectively. (**CNet** and **RNet** are short for CoarseNet and RefineNet.)

6.1 Ablation Study for Network Structure

We quantitatively analyzed different components of TOM-Net using synthetic dataset[5]. We first verified the effectiveness of *refractive flow regression loss* ($\mathcal{L}^c_{fr}$), *cross-link*, *multi-scale loss* and *image reconstruction loss* ($\mathcal{L}^c_{ir}$) in the coarse stage by removing each of them from *CoarseNet* during training. We then validated the effectiveness of *RefineNet* in recovering details of the refractive flow field. RefineNet was evaluated by adding it to a trained CoarseNet and was trained while fixing the parameters of CoarseNet. For comparison, we also included a naive baseline, denoted as *Background*, by considering a zero matte case (i.e., whole image as object mask, no attenuation, and no refractive flow) where the reconstructed image is the same as the background image. The quantitative results are summarized in Tab. 4 and the qualitative comparisons are shown in Fig. 5. Overall, the baseline *Background* was outperformed by all TOM-Net variants with a large margin for all the evaluation metrics, which clearly shows that TOM-Net can successfully learn the matte.

Effectiveness of refractive flow regression loss Comparing experiments with IDs 1 & 5 in Tab. 4, it can be clearly seen that the CoarseNet trained with the refractive flow regression loss significantly outperformed that without it in refractive flow estimation. This result indicates that image reconstruction loss alone is not enough to supervise the learning of refractive flow. Fig. 5 (a & e) qualitatively show that the refractive flow regression loss improved the performance of refractive flow estimation.

[5] Complex shape is excluded in experiments here to speed up training.

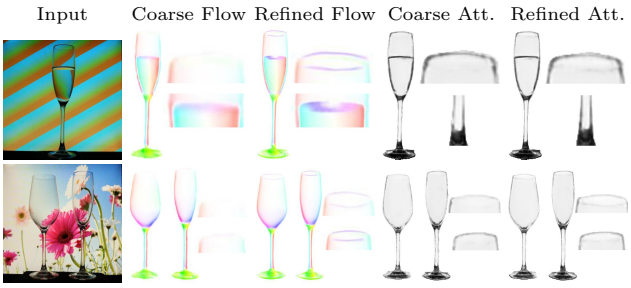


Fig. 6 Visualization of the effectiveness of the refinement stage on real data. After refinement, the refractive flow and attenuation mask have more clear structural details (e.g., glass mouth).

Effectiveness of cross-link Comparing experiments with IDs 2 & 5 in Tab. 4, we can see that augmenting the decoders of CoarseNet with the cross-link helped improve the performance in all metrics, suggested that utilizing correlation is helpful for the matte estimation. Fig. 5 (b & e) qualitatively show the results without and with the cross-link during training.

Table 5 Quantitative results on the synthetic test dataset. (The first value for EPE is measured on the whole image and the second measured within the object region. A-MSE and I-MSE are computed on the whole image.)

	Glass				Glass with Water				Lens				Complex Shape				Average			
	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU
Background	3.6 / 30.3	1.33	0.48	0.12	6.4 / 53.2	1.54	0.68	0.12	10.3 / 39.2	1.94	1.57	0.24	6.8 / 56.8	2.50	0.85	0.11	6.8 / 44.9	1.83	0.90	0.15
CoarseNet	2.1 / 15.8	0.22	0.14	0.97	3.1 / 23.5	0.31	0.23	0.97	2.0 / 6.7	0.17	0.28	0.99	4.5 / 34.4	0.38	0.33	0.92	2.9 / 20.1	0.27	0.24	0.96
TOM-Net	1.9 / 14.7	0.21	0.14	0.97	2.9 / 21.8	0.30	0.22	0.97	1.9 / 6.6	0.15	0.29	0.99	4.1 / 31.5	0.37	0.32	0.92	2.7 / 18.6	0.26	0.24	0.96

MSE ($\cdot 10^{-2}$)
↓ better
↑ better

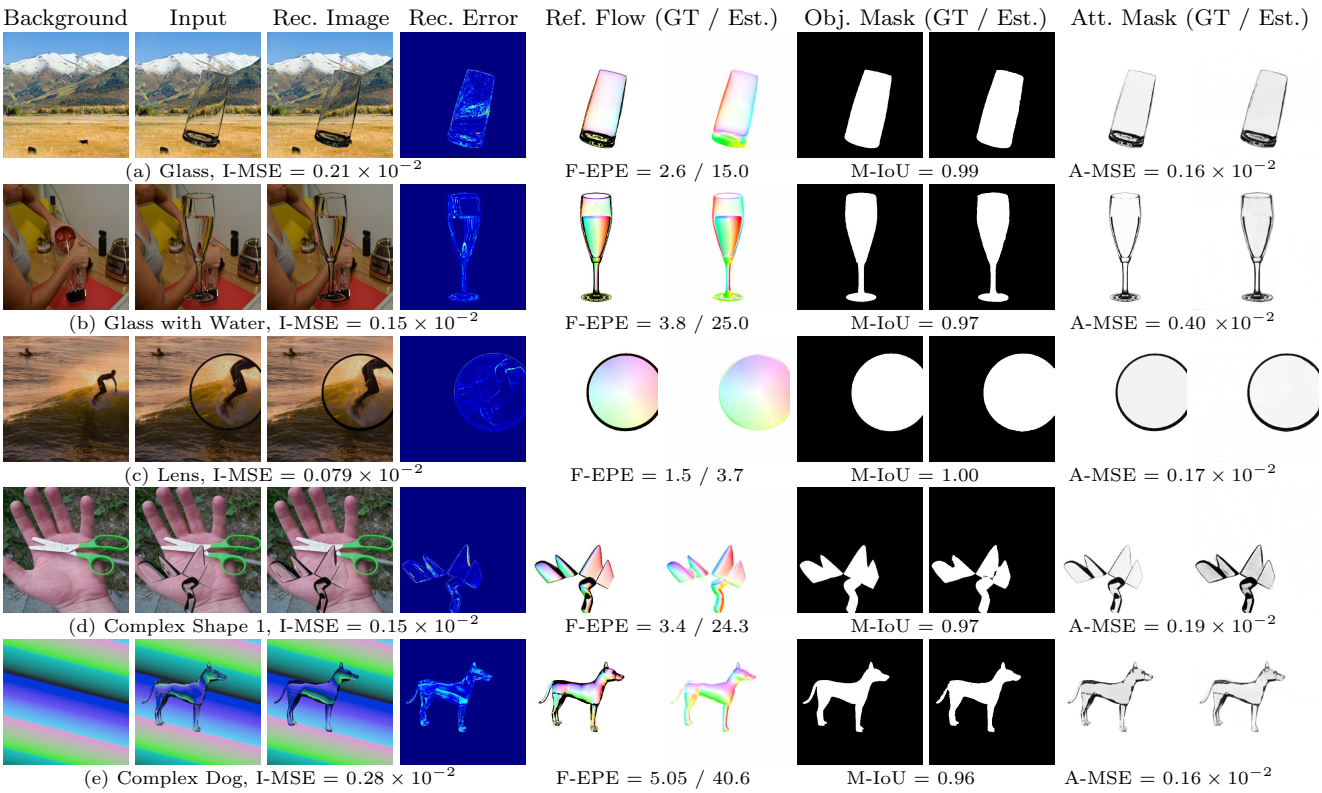


Fig. 7 Qualitative results on synthetic data. The first to the fourth columns show background, input image, reconstructed image, and reconstruction error map, respectively. Quantitative results are shown below each example. Dark region in GT flow indicates no valid flow. (Best viewed in PDF with zoom.)

Effectiveness of multi-scale loss Comparing experiments with IDs 3 & 5 in Tab. 4, we can see that multi-scale loss boosted performance of CoarsNet in all of the evaluation metrics, particularly the attenuation mask MSE (see Fig. 5 (c & e) for qualitative comparison).

Effectiveness of image reconstruction loss Comparing experiments with IDs 4 & 5 in Tab. 4, we can see that adding image reconstruction loss in the coarse stage slightly improved the performance of refractive flow estimation and was very effective for reducing the image reconstruction error (see Fig. 5 (d & e) for qualitative comparison).

Effectiveness of RefineNet Comparing experiments with IDs 5 & 6 in Tab. 4, we can clearly see that RefineNet can significantly improve the refractive flow estimation. Fig. 5 (e & f) and Fig. 6 show that RefineNet can infer sharp details on both the synthetic and real data based on the outputs of CoarseNet, demonstrating the effectiveness of the RefineNet. We also found that image reconstruction loss is not helpful for refractive flow estimation in the refinement stage (experiments with IDs 6 & 7 in Tab. 4). This is reasonable since the matte produced by CoarseNet already gives a small image reconstruction error, and further reducing the image reconstruction error does not guarantee a better refractive flow field.

6.2 Results on Synthetic Data

Quantitative results for synthetic test dataset are presented in Tab. 5. We compared TOM-Net against *Background* and CoarseNet. Here, to accelerate training convergence, we first trained CoarseNet from scratch using our synthetic dataset excluding the complex shape subset. The trained CoarseNet was then fine-tuned using

Table 6 Quantitative results on real data. (Value the higher the better.)

	Glass		G & W		Lens		Complex		Avg	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Background	22.05	0.894	20.75	0.886	18.60	0.860	16.85	0.816	19.56	0.864
CoarseNet	25.09	0.921	23.53	0.911	21.13	0.895	17.89	0.835	21.91	0.891
TOM-Net	25.06	0.920	23.53	0.911	20.89	0.893	17.88	0.835	21.84	0.890

Table 7 User study results. P, C, and N are short for votes for photograph, composite, and not distinguishable.

	Glass			G & W			Lens			Complex			All		
	P	C	N	P	C	N	P	C	N	P	C	N	P	C	N
Photographs	522	275	31	163	97	16	74	48	16	91	35	12	850	455	75
Composites	531	266	31	145	113	18	73	52	13	78	51	9	827	482	71

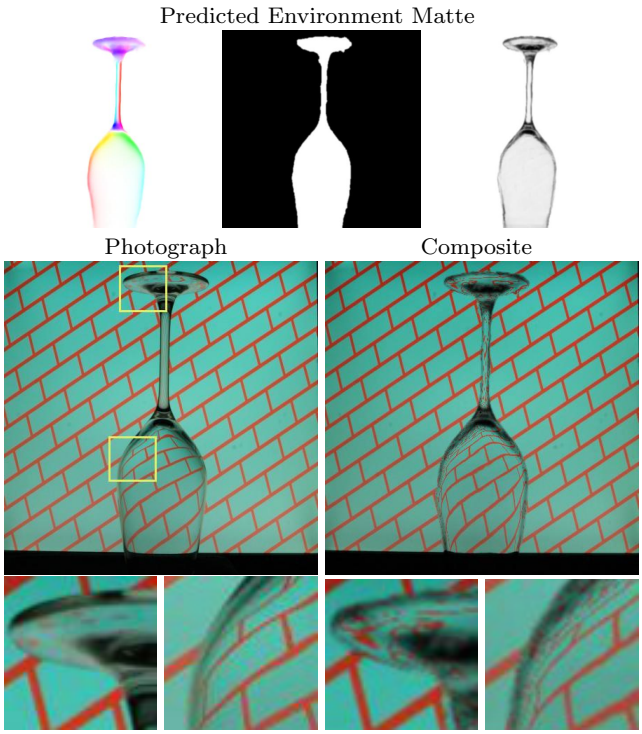


Fig. 8 The first row shows the predicted matte, which is estimated by taking the photograph as input to our method. The second row compares the photograph and composite and the third row shows the zoom-in comparisons. When looking at the photograph and composite simultaneously, users can easily spot some imperfections of the composites (mostly in the boundary region).

the entire training set including complex shapes, followed by training of RefineNet on the entire training set with random initialization. Similar to previous experiments, TOM-Net outperformed *Background* by a large margin, and slightly outperformed CoarseNet in both EPE and MSE, which implies more local details can be learned by RefinedNet.

The average IoU for object mask estimation is 0.96, indicates that TOM-Net can robustly segment the transparent object given only a single image as input. Although TOM-Net is not expected to learn highly accurate refractive flow, the average EPE errors (2.7/18.6)[6] are very small compared with the size of the input image (448×448). In this sense, our predicted flow is capable of producing visually plausible refractive effect. The errors of complex shape category are larger than that of others, because complex shapes contain more sharp regions that will induce more errors. Fig. 7 shows the qualitative results on synthetic dataset. The objects in the first four rows come from the test set where each row shows a specific object category. Although the background images and objects in the test set never appear in the training set, TOM-Net can still predict robust matte. The last row shows a sample of complex dog shape, which was rendered using a 3D dog model. The pleasing result on the complex dog shape demonstrates that our model can generalize well from simple shapes to complex shapes.

6.3 Results on Real Data

We evaluated TOM-Net on our captured real dataset, which consists of 876 images of real objects. The results are shown in Tab. 6. The average PSNR and SSIM are above 21.0 and 0.89 respectively. The values are a bit lower for complex shapes, due to the opaque base of complex objects as well as the sharp regions of the objects that might induce large errors. After training, TOM-Net generalized well to common real transparent objects (see Fig. 9). It is worth to note that during training, each sample contains only one object, while TOM-Net can predict reliable matte for images containing multiple objects (see Fig. 9 (c)), which indicates the transferability and robustness of TOM-Net.

User study A user study was carried out to validate the realism of TOM-Net composites. 69 subjects participated in our user study. At the beginning, we showed each participant photographs of the transparent objects that will be seen during the user study. The objects consisted of 3 different glasses, 1 glass with water, 1 lens, and 1 complex shape. 40 samples, including 20 photographs[7] and the corresponding 20 TOM-Net composites, were then randomly presented to each subject. When showing each sample, we also showed the corresponding background image to the subject for refer-

[6] The first value is measured on the whole image and the second measured within the object region.

[7] glass ×12, glass & water ×4, lens ×2, and complex shape ×2.

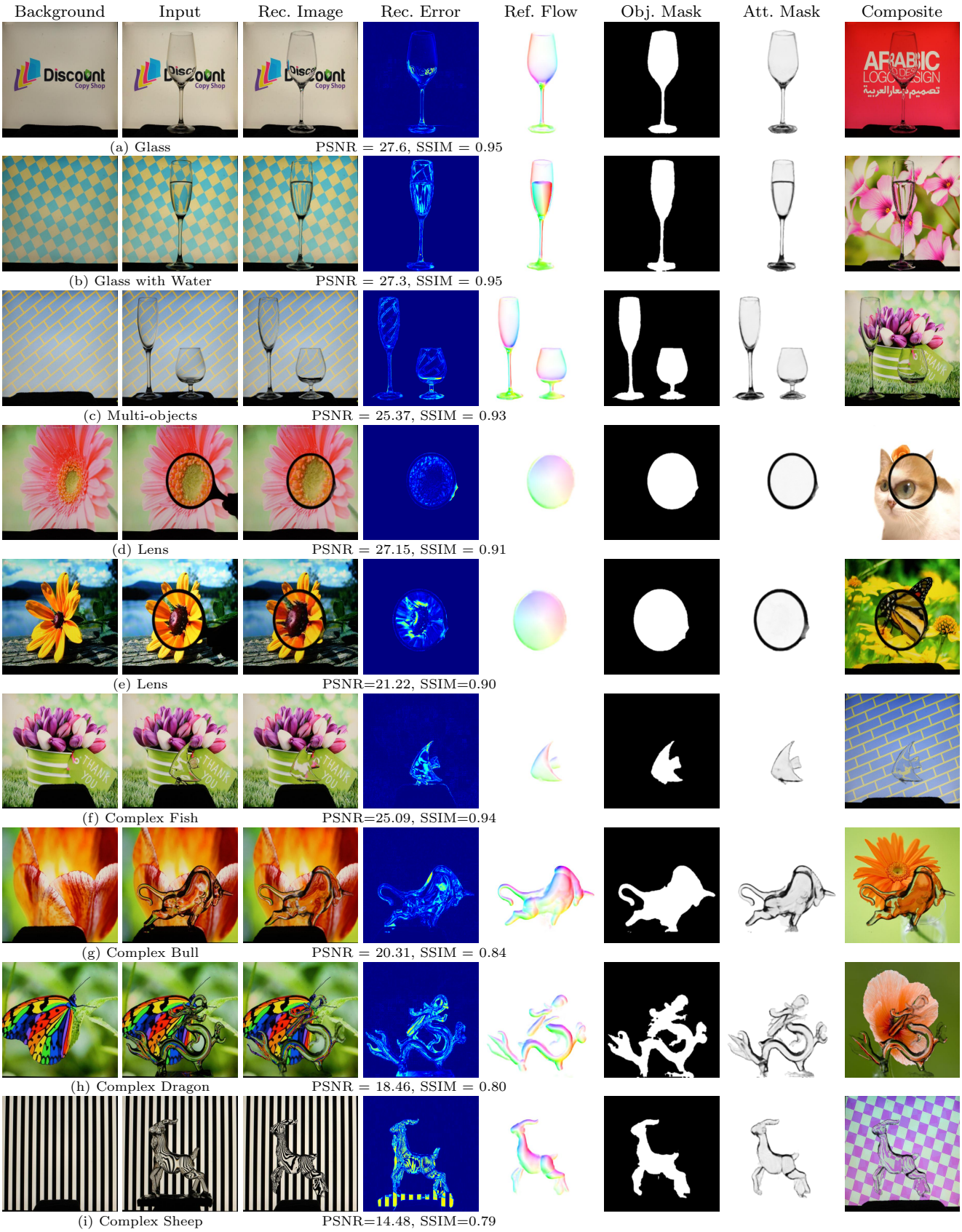
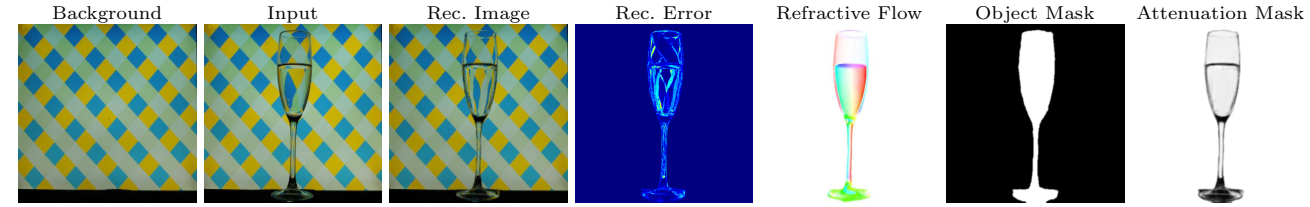
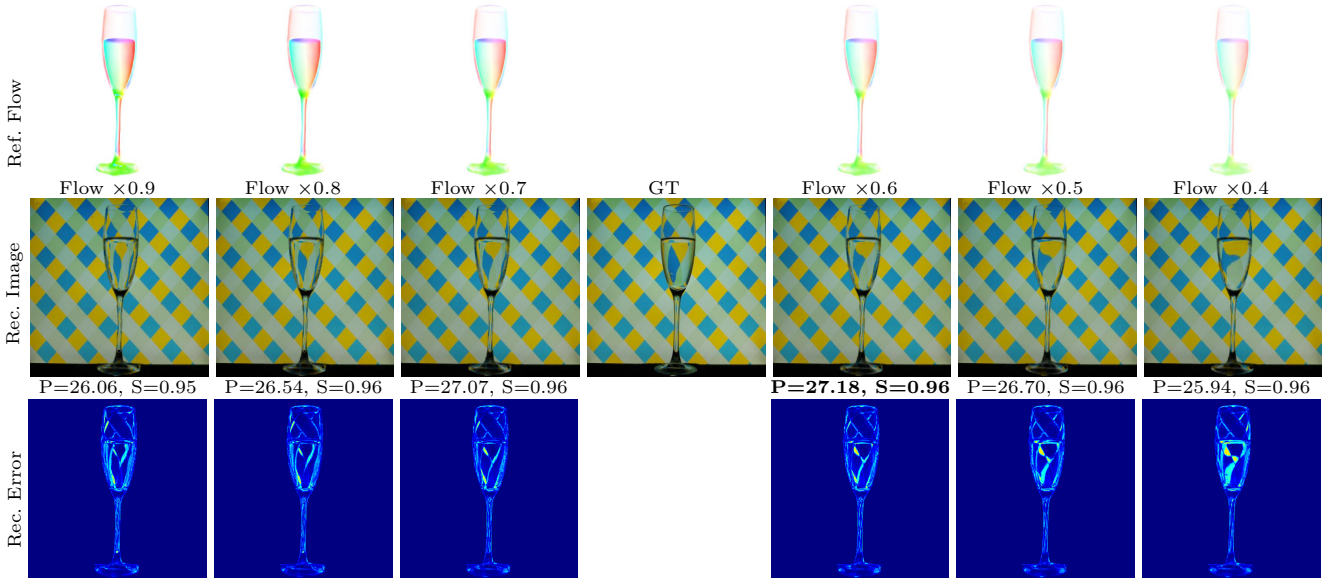


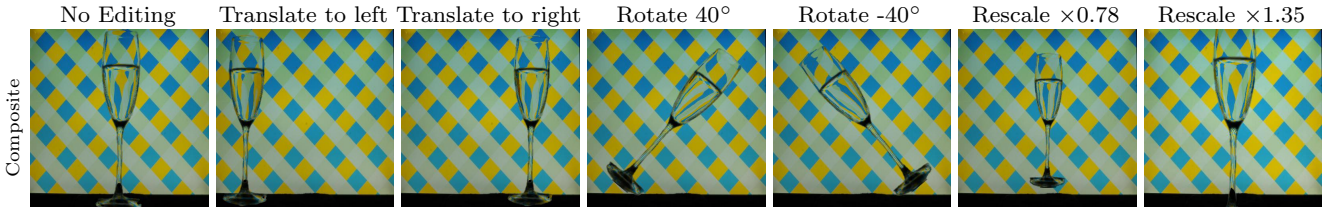
Fig. 9 Qualitative results on real data. The PSNR and SSIM between input photographs and reconstructed images are shown below each example. The last column shows the composites on novel backgrounds given the estimated matte. (Best viewed in PDF with zoom.)



(a) An example result on *glass with water*. Reconstruction error: PSNR=25.69, SSIM=0.95



(b) Rescaling the magnitude of the estimated refractive flow field.



(c) Translate, rotate or rescale the environment matte.

Fig. 10 Various novel composites of a *glass with water* shape obtained by manipulating the predicted environment matte.

ence. We provided 3 options for each sample: (P) *photograph*, (C) *composite*, (N) *not distinguishable*. Tab. 7 shows the statistics of the user study. The 69 participants produced 1,380 votes for the 20 real photographs, and 1,380 votes for the 20 composites, respectively. The P:C:N ratios are 850 : 455 : 75 and 827 : 482 : 71 for photographs and composites respectively. The per-category ratios also follow a similar trend, indicating close chance of photographs and composites to be considered real, which further demonstrates TOM-Net can produce realistic matte.

Although we stress that TOM-Net can produce visually realistic composites, the results are still less than perfect. When looking at the real image and our composite side-by-side, users can spot some imperfections of the composite (mostly in the boundary region, see Fig. 8). Therefore, we did not include such a user study by showing the real image and our composite side-by-side. Otherwise, the result will be biased. In the future, we will strengthen our approach to produce more realistic composites, so that the real image and our composite are indistinguishable even when showing them side-by-side.

6.4 Transparent Object Editing by Manipulating Environment Matte

Given a single image as input, our TOM-Net can estimate the environment matte as a triplet (consisting of an object mask, an attenuation mask and a refractive flow field) in a fast feed-forward pass (see Fig. 10 (a) for an example). Note that the goal of the proposed TOM-Net is to extract an environment matte that can produce realistic refractive effect from a single

Table 8 Quantitative comparison between TOM-Net, TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ on the synthetic test dataset.

	Glass				Glass with Water				Lens				Complex Shape				Average			
	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU	F-EPE	A-MSE	I-MSE	M-IoU
Background	3.6 / 30.3	1.33	0.48	0.12	6.4 / 53.2	1.54	0.68	0.12	10.3 / 39.2	1.94	1.57	0.24	6.8 / 56.8	2.50	0.85	0.11	6.8 / 44.9	1.83	0.90	0.15
TOM-Net	1.9 / 14.7	0.21	0.14	0.97	2.9 / 21.8	0.30	0.22	0.97	1.9 / 6.6	0.15	0.29	0.99	4.1 / 31.5	0.37	0.32	0.92	2.7 / 18.6	**0.26**	0.24	0.96
TOM-Net$^{+\text{Trimap}}$	1.8 / 14.4	0.21	0.14	0.98	2.6 / 20.7	0.29	0.20	0.98	1.7 / 6.1	0.15	0.27	1.00	3.7 / 29.4	0.37	0.29	0.95	2.5 / 17.7	**0.26**	0.23	**0.98**
TOM-Net$^{+\text{Bg}}$	1.6 / 13.1	0.21	0.12	0.99	2.4 / 19.3	0.29	0.19	0.98	1.4 / 4.9	0.18	0.19	1.00	3.5 / 27.7	0.36	0.27	0.97	**2.2 / 16.2**	**0.26**	**0.19**	**0.98**

MSE ($\cdot 10^{-2}$)
↓ better
↑ better

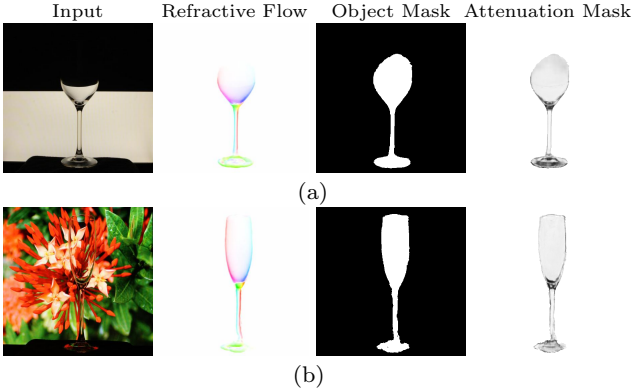


Fig. 11 Two failure cases in real data. In (a), our model fails to estimate the upper-part of the matte as there is no visual clue to find the object. In (b), the bottom part of the estimated matte is incomplete as the background image is heavily cluttered and the bottom part of the object is very dark.

image, instead of estimating highly accurate environment matte. The reconstructed image in Fig. 10 (a) looks realistic but does not have the same refractive effect as the original input, as the refractive effect of the estimated matte seems stronger. By decreasing the magnitude of the estimated refractive flow field[8], we can produce a similar refractive effect as the input image (see Fig. 10 (b)). When the scaling factor becomes 0.6, the reconstructed image achieves the lowest reconstruction error, with an improvement of 1.49 and 0.01 in PSNR and SSIM, respectively. Apart from rescaling the magnitude of the refractive flow field to adjust the refractive effect of the object, more interesting composites can be obtained by translating, rotating and rescaling the environment matte (see Fig. 10 (c)).

6.5 Failure Cases

Our model can robustly estimate environment matte for different transparent objects in front of different backgrounds, however, when there is no visual clue for the objects or the image is too cluttered to separate the object from the background, our model may fail. Fig. 11 shows two failure cases of our model on real data. In Fig. 11 (a), our model fails to extract the upper-part of the environment matte for the transparent glass due to the lack of visual clue. In Fig. 11 (b), although our model is still able to estimate a reasonable matte, the bottom part of the estimated matte is incomplete due to the very cluttered background.

[8] We simply multiply the refractive flow field by a scaling factor (< 1).

Table 9 Quantitative comparison between TOM-Net, TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ on real data.

	Glass		G & W		Lens		Complex		Avg	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Background	22.05	0.894	20.75	0.886	18.60	0.860	16.85	0.816	19.56	0.864
TOM-Net	25.06	0.920	23.53	0.911	20.89	0.893	17.88	0.835	21.84	0.890
TOM-Net$^{+\text{Trimap}}$	25.48	0.924	23.77	0.914	23.98	0.913	20.88	0.868	23.53	0.905
TOM-Net$^{+\text{Bg}}$	26.10	0.931	24.58	0.922	25.52	0.924	22.23	0.884	**24.61**	**0.915**

6.6 Improvement with Trimap and Background Image

At test time, the input trimaps for TOM-Net$^{+\text{Trimap}}$ were generated in the same way adopted in the training (as described in Subsection 4.4), except that the foreground regions were obtained by performing erosion operation on the ground-truth object mask with a fixed (rather than a random) kernel size of 10 pixels for evaluation. Tab. 8 shows the quantitative comparisons between TOM-Net, TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ on the synthetic test dataset. As expected, with the access to the additional information, both TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ performed better than TOM-Net. Due to the fact that a background image contains more useful information than a trimap, TOM-Net$^{+\text{Bg}}$ achieved the best results.

Tab. 9 presents the quantitative comparison on real data. Compared with TOM-Net, TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ achieved an improvement of 1.69 and 2.77 in average PSNR and an improvement of 0.015 and 0.024 in average SSIM, respectively. Fig. 12 shows the qualitative comparison on real data, where the foreground region of the trimap was marked by the user. It can be seen that with the additional information, TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ can identify the transparent object from the cluttered background more accurately than TOM-Net and model the opaque base of the transparent object (Fig. 12 (b)). As a result, the environment matte predicted by TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ can produce more realistic composites and achieve lower reconstruction errors, clearly demonstrating the effectiveness of our framework in handling

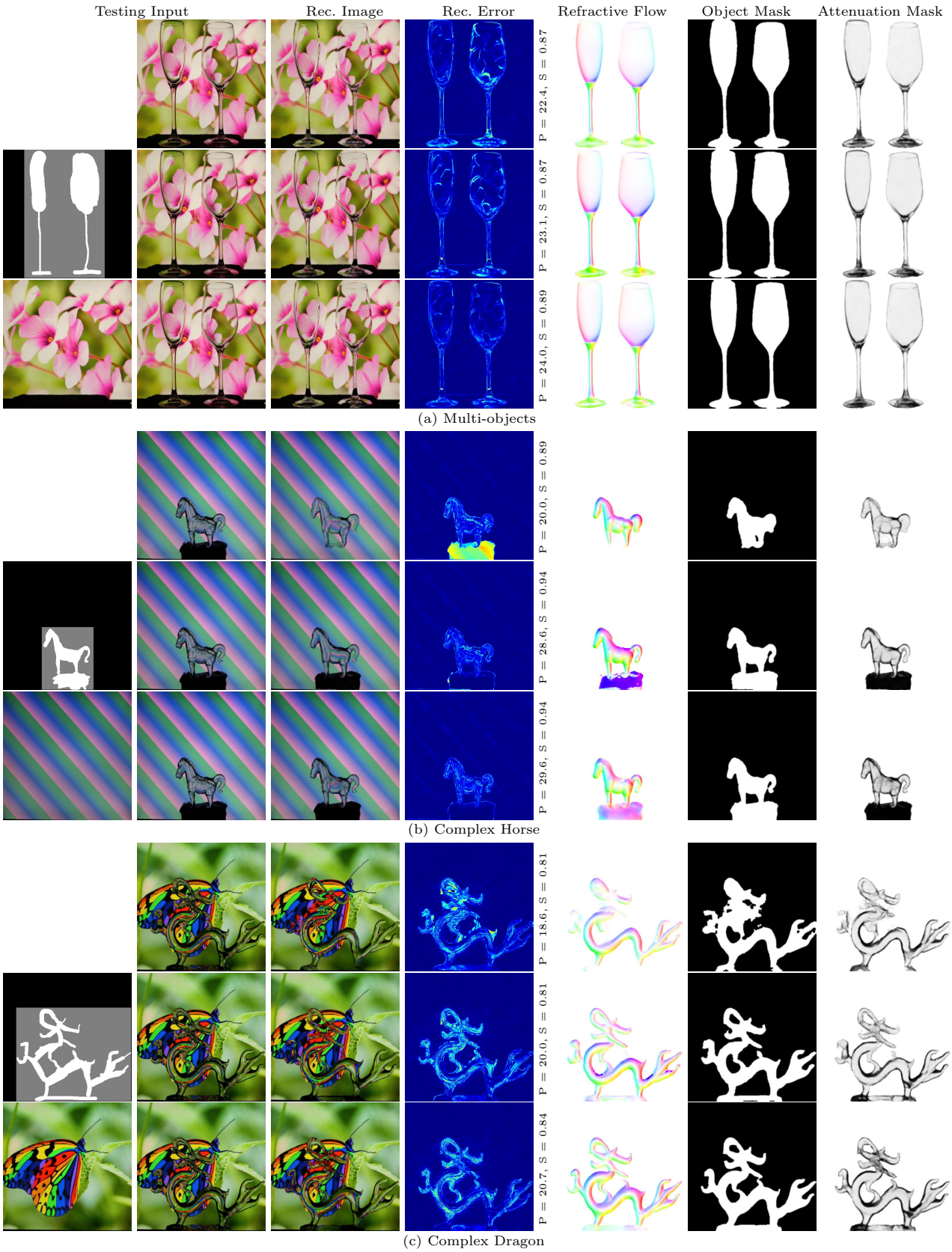


Fig. 12 Qualitative comparison between TOM-Net, TOM-Net$^{+\text{Trimap}}$ and TOM-Net$^{+\text{Bg}}$ on real data. For each testing object, the input to the model is shown on the first two columns, and the results of TOM-Net (up), TOM-Net$^{+\text{Trimap}}$ (middle) and TOM-Net$^{+\text{Bg}}$ (bottom) are shown on the rest of the columns. The PSNR and SSIM between input photographs and reconstructed images are shown right after the error maps.

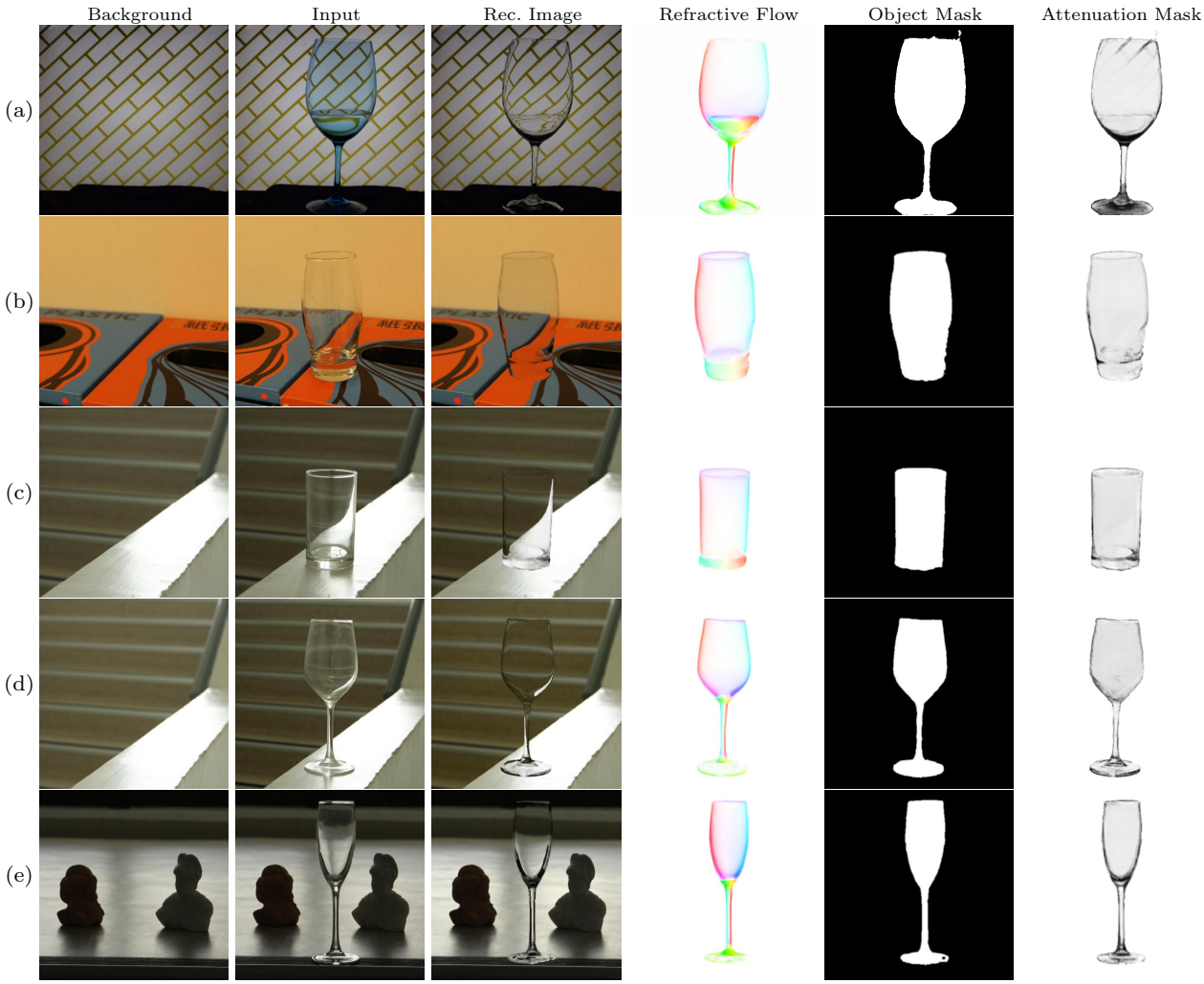


Fig. 13 Qualitative results of TOM-Net on colored transparent object (first row) and objects under natural illumination (last four rows).

cases where a trimap or a background image is available.

7 Discussion

7.1 Limitations

Although our method can produce plausible results for transparent object matting, there do exist limitations that require further study. First, our model assumes objects to be colorless so that the attenuation property of an object can be depicted as a scalar value ρ in our formulation. However, this is not applicable to colored transparent objects, as shown in see Fig. 13 (a). Although our method can estimate a reasonably good object mask and refractive flow field for the glass with water, the estimated attenuation mask cannot model the colored effect of the object.

Second, our model assumes a single planar background (following most of the previous works) as the only light source and simplifies the interaction between object and background image to a point-to-point (single) mapping. However, more complicated effects exist in the real world, such as specular highlights, translucent, multi-mapping (i.e., refraction and reflection happen simultaneously at a surface point), and color dispersion (i.e., different color components may have different supporting background regions). Fig. 13 (b)-(e) show four example results of TOM-Net on transparent objects under different types of natural illuminations. Regardless of the fact that TOM-Net can estimate a plausible object mask and refractive flow field, the composites do not look very realistic. This is because our current formulation does not consider the more sophisticated refractive properties of a transparent object under natural illumination like complex interaction with

environment lighting, specular highlight, Fresnel effect, and acoustic shadow.

7.2 Colored Objects and Specular Highlights

Here we sketch the potential solutions to colored transparent objects as well as the cases when specular highlights appear on transparent objects. In Section 3, we simplified matting equation as (6). To handle colored objects, the scalar attenuation index ρ should be expanded to a color attenuation 3-vector R, in which each value corresponds to an attenuation index for a specific color channel. The matting equation then becomes

$$C = (1 - m)B + mR \circ \mathcal{M}(\mathbf{T}, P), \tag{13}$$

where $\circ$ represents element-wise multiplication.

Consider a white near point light source, we can simplify the specular highlight effect with a specular highlight component S, then the generalized matting equation can be written as

$$C = (1 - m)B + mR \circ \mathcal{M}(\mathbf{T}, P) + S, \tag{14}$$

where S is a 3-vector containing three identical values. The problem of transparent object matting now becomes simultaneously estimating an object mask, a color attenuation mask, a refractive flow field and a specular highlight mask from a single image, while more efforts are needed to implement them for practical use and we leave this as our future work.

7.3 Difficulty in Comparison with Previous Works

Currently, it is not trivial to have a fair comparison with existing methods. On one hand, applying our method on the data used in the previous methods is difficult. Most of the previous methods require multiple images of the transparent object captured in front of predesigned patterns, which are not publicly available and lack enough textures for our method to estimate the refractive effect of the transparent object. The single image based methods RTCEM [4] and [26] have additional requirements. In particular, RTCEM [4] requires the object to be captured in front of a coded-pattern (also not publicly available), and the background image is needed to segment the foreground object. [26] requires human interaction to segment the foreground object and model the object's refractive effect with thin-plate-spline transformation. The data used in [26] does not follow our assumption that the light comes from a single background image, thus it cannot be directly processed by our method. On the other hand, there are no public implementations for the previous methods, and even if there were, those methods cannot be applied to our dataset which is created for single image transparent object matting.

Different from the previous methods, our method aims to estimate the foreground mask, attenuation mask and refractive flow field from a single natural image. Since our code and datasets have been made publicly available, it will ease the comparison for the following work. We believe our work can serve as a baseline and provide meaningful insight for future researches in this area.

8 Conclusion

We have introduced a simple and efficient model for transparent object matting, and proposed a CNN architecture, called TOM-Net, that takes a single image as input and predicts environment matte as an object mask, an attenuation mask, and a refractive flow field in a fast feed-forward pass. Besides, we created a large-scale synthetic dataset and a real dataset as a benchmark for learning transparent object matting. We have also shown that TOM-Net can perform better by incorporating a trimap or a background image in the input. Promising results have been achieved on both synthetic and real data, which clearly demonstrate the feasibility and effectiveness of the proposed approach. We consider exploring better models and architectures for transparent object matting as our future work.

Acknowledgments This project is supported by a grant from the Research Grant Council of the Hong Kong (SAR), China, under the project HKU 718113E. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

References

1. Persistence of vision (tm) raytracer. http://www.povray.org/ 7
2. Chen, G., Han, K., Wong, K.Y.K.: TOM-Net: Learning transparent object matting from a single image. In: CVPR (2018) 2
3. Cho, D., Tai, Y.W., Kweon, I.: Natural image matting using deep convolutional neural networks. In: ECCV (2016) 3
4. Chuang, Y.Y., Zongker, D.E., Hindorff, J., Curless, B., Salesin, D.H., Szeliski, R.: Environment matting extensions: Towards higher accuracy and real-time capture. In: SIGGRAPH (2000) 2, 3, 4, 17
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR (2009) 7

6. Duan, Q., Cai, J., Zheng, J.: Compressive environment matting. The Visual Computer (2015) 2
7. Duan, Q., Cai, J., Zheng, J., Lin, W.: Fast environment matting extraction using compressive sensing. In: ICME (2011) 3
8. Duan, Q., Zheng, J., Cai, J.: Flexible and accurate transparent-object matting and compositing using refractive vector field. In: Computer Graphics Forum (2011) 2
9. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NIPS (2014) 5
10. Fischer, P., Dosovitskiy, A., Ilg, E., Häusser, P., Hazırbaş, C., Golkov, V., van der Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. In: ICCV (2015) 5
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 5
12. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep networks. In: CVPR (2017) 6
13. Kim, J., Kwon Lee, J., Mu Lee, K.: Accurate image super-resolution using very deep convolutional networks. In: CVPR (2016) 6
14. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015) 6
15. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014) 7
16. Nah, S., Kim, T.H., Lee, K.M.: Deep multi-scale convolutional neural network for dynamic scene deblurring. In: CVPR (2017) 6
17. Peers, P., Dutré, P.: Wavelet environment matting. In: Eurographics workshop on Rendering (2003) 2, 3
18. Qian, Y., Gong, M., Yang, Y.H.: Frequency-based environment matting by compressive sensing. In: ICCV (2015) 3
19. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention (2015) 5
20. Shen, X., Tao, X., Gao, H., Zhou, C., Jia, J.: Deep automatic portrait matting. In: ECCV (2016) 3
21. Shi, J., Dong, Y., Su, H., Yu, S.X.: Learning non-Lambertian object intrinsics across shapenet categories. In: CVPR (2017) 5
22. Smith, A.R., Blinn, J.F.: Blue screen matting. In: SIGGRAPH (1996) 1
23. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE TIP (2004) 3, 8
24. Wexler, Y., Fitzgibbon, A.W., Zisserman, A., et al.: Image-based environment matting. In: Rendering Techniques (2002) 2, 3
25. Xu, N., Price, B., Cohen, S., Huang, T.: Deep image matting. In: CVPR (2017) 3, 5
26. Yeung, S.K., Tang, C.K., Brown, M.S., Kang, S.B.: Matting and compositing of transparent and refractive objects. ACM TOG (2011) 3, 17
27. Zhu, J., Yang, Y.H.: Frequency-based environment matting. In: Computer Graphics and Applications (2004) 2, 3
28. Zongker, D.E., Werner, D.M., Curless, B., Salesin, D.H.: Environment matting and compositing. In: SIGGRAPH (1999) 2, 3, 4