Reflection Separation via Multi-bounce Polarization State Tracing

Rui Li[★], Simeng Qiu[★], Guangming Zang, and Wolfgang Heidrich

King Abdullah University of Science and Technology, Thuwal, SA
{rui.li, simeng.qiu, guangming.zang, wolfgang.heidrich}@kaust.edu.sa

1 Scene Parameter Estimation

To compare the scene parameters estimation of our simulator, Tab. 1 shows our estimated scene parameters ($\tilde{\theta}_i$, $\tilde{d}$, $\tilde{\Delta}x$) and GT scene parameters (θ_i, d, Δx), experiment shows that our simulator accurately recover the core parameters.

Table 1: Scene parameters (θ_i, d, Δx) estimation.

$\theta_i \setminus d \setminus \Delta x$	1.50\ 10 \ 17.8	1.30\ 9.5 \ 15.9	1.20\ 8.5 \ 13.5	1.12\ 9.5 \ 14.3
$\tilde{\theta}_i \setminus \tilde{d} \setminus \Delta\tilde{x}$	1.49\ 9.84 \ 17.5	1.30\ 9.26 \ 15.5	1.30\ 8 \ 13.4	1.13\ 9 \ 13.6

2 Physically-Based Image Formation Model

In this section we provide an expanded version of Section 3 of the main paper, to have a more complete derivation of multi-bounce polarization state tracing and the differentiable implementation of the polarization simulation engine. We show the details of multi-bounce light propagation in double-surfaced transparent media, as well as the ghost images in both real scenes and our simulated data.

2.1 Reflectance and Transmittance

Snell's law. Considering a local coordinate frame of a light ray hitting a transparent surface (Fig. 1a), a plane of incidence (POI) subsumes the transmission angle θ_t and the reflection angle θ_r, which is equal to the incident angle $\theta = \theta_i$. The angles are related to the refractive indices according to Snell's law:

$$n_0 \sin \theta = n \sin \theta_t, \tag{1}$$

where the transparent medium has refractive index n, and n_0 is the refractive index of the ambient medium (e.g. $n_0 \approx 1$ for air). Therefore, a transmitted angle θ_t that acquired by Snell's law is:

$$\theta_t(\theta, n) = \arcsin(\frac{1}{n} \sin(\theta)), \tag{2}$$

[★] The two authors assert jointly first authorship.

Fresnel equations. An incident light passing through or reflected off a transparent media is partially polarized and consists of two orthogonal polarized components that perpendicular and parallel to the POI (Fresnel equations). We define reflectance R and transmittance T as the intensity ratio of reflected light and transmitted light to incident light, respectively. The subscripts $\parallel$ and $\perp$ represent the polarized components parallel and perpendicular to the POI. R is derived from two orthogonal polarized elements of reflectance, $R = (R_\parallel + R_\perp)/2$. Likewise, $T = (T_\parallel + T_\perp)/2$.

When the light travels from the outside (optically thinner medium) to the transparent object (optically thicker medium), the expression of the four orthogonal components are,

$$R_\perp^1(\theta, n) = \frac{\sin^2(\theta - \theta_t)}{\sin^2(\theta + \theta_t)},$$

$$R_\parallel^1(\theta, n) = \frac{\tan^2(\theta - \theta_t)}{\tan^2(\theta + \theta_t)}, \tag{3}$$

$$T_\perp^1(\theta, n) = \frac{\sin 2\theta \sin 2\theta_t}{\sin^2(\theta + \theta_t)},$$

$$T_\parallel^1(\theta, n) = \frac{\sin 2\theta \sin 2\theta_t}{\sin^2(\theta + \theta_t) \cos^2(\theta - \theta_t)}.$$

When the light travels from the inside (optically thicker) to the outside (optically thinner), the expression of the orthogonal components are,

$$R_\perp^2(\theta, n) = \frac{\sin^2(\theta_t - \theta)}{\sin^2(\theta_t + \theta)},$$

$$R_\parallel^2(\theta, n) = \frac{\tan^2(\theta_t - \theta)}{\tan^2(\theta_t + \theta)}, \tag{4}$$

$$T_\perp^2(\theta, n) = \frac{\sin 2\theta_t \sin 2\theta}{\sin^2(\theta_t + \theta)},$$

$$T_\parallel^2(\theta, n) = \frac{\sin 2\theta_t \sin 2\theta}{\sin^2(\theta_t + \theta) \cos^2(\theta_t - \theta)}.$$

2.2   Stokes Vector and Mueller Matrix

Stokes Vector. is a way to describe a complete polarization state of light. In the case the linear polarization, we only consider the 3D vector, corresponding to the first 3 components of the full stokes vector. In the coordinate frame of the color polarization camera sensor, we measure four different linear polarizer angles resulting in four images $I^{0°}$, $I^{45°}$, $I^{90°}$, and $I^{135°}$, which are acquired simultaneously with a polarization image sensor. Given the four linear polarizer images our reflected Stokes vector $\mathbf{s_t}$ and transmitted Stokes vector $\mathbf{s_r}$ in the image plane can be computed as
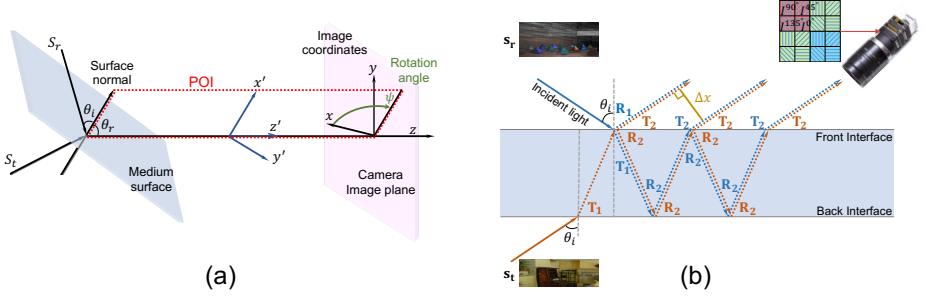
Fig. 1: Light reflection and transmission model: Incident angle θ_i, reflection angle θ_r, transmission angle θ_t, and POI. (a) Object surface and camera image plane are in different coordinate (b) Multiple light paths for reflected the (blue) and transmitted (red) scenes. The top-right is a color polarization camera with a micro polarizer at four different angles and a Bayer filter layout.

$$
\mathbf{s}_t = \begin{bmatrix} s_{0t} \\ s_{1t} \\ s_{2t} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(I_t^{0^\circ} + I_t^{45^\circ} + I_t^{90^\circ} + I_t^{135^\circ}) \\ I_t^{0^\circ} - I_t^{90^\circ} \\ I_t^{45^\circ} - I_t^{135^\circ} \end{bmatrix}, \tag{5}
$$

$$
\mathbf{s}_r = \begin{bmatrix} s_{0r} \\ s_{1r} \\ s_{2r} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(I_r^{0^\circ} + I_r^{45^\circ} + I_r^{90^\circ} + I_r^{135^\circ}) \\ I_r^{0^\circ} - I_r^{90^\circ} \\ I_r^{45^\circ} - I_r^{135^\circ} \end{bmatrix}. \tag{6}
$$

Mueller Matrix. describes the change of polarization states as we propagate the light through the transparent surface, which operates on the Stokes vectors. To transform the vector between the local coordinate frame of the transparent surface and the coordinate frame of the camera sensor, we introduce a rotation Mueller matrix:

$$
\mathbf{C}(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 2\psi & -\sin 2\psi \\ 0 & \sin 2\psi & \cos 2\psi \end{bmatrix}, \tag{7}
$$

which maps a Stokes vector $\hat{\mathbf{s}}$ in local coordinates to a Stokes vector $\mathbf{s}$ in global coordinates. $\mathbf{C}^{-1}(\psi) = \mathbf{C}(-\psi)$ is used for the reverse mapping.

There are two pairs of Mueller matrices for the individual reflection and transmission operations along the light path (Fig. 1(a),(b)): $\mathbf{R}_1$ and $\mathbf{T}_1$ describe the reflection and transmission as the light travels from the outside (optically thinner medium) to the transparent object (optically thicker medium). When the light travels from the inside (optically thicker) to the outside (optically thinner), reflection and transmission are respectively described by $\mathbf{R}_2$ and $\mathbf{T}_2$ [1]. A

detailed description of these components are,

$$\mathbf{R}_1 = \begin{bmatrix} \frac{R_\parallel^1 + R_\perp^1}{2} & \frac{R_\parallel^1 - R_\perp^1}{2} & 0 & 0 \\ \frac{R_\parallel^1 - R_\perp^1}{2} & \frac{R_\parallel^1 + R_\perp^1}{2} & 0 & 0 \\ 0 & 0 & \sqrt{R_\parallel^1 R_\perp^1} & 0 \\ 0 & 0 & 0 & \sqrt{R_\parallel^1 R_\perp^1} \end{bmatrix},\qquad(8)$$

$$\mathbf{T}_1 = \begin{bmatrix} \frac{T_\parallel^1 + T_\perp^1}{2} & \frac{T_\parallel^1 - T_\perp^1}{2} & 0 & 0 \\ \frac{T_\parallel^1 - T_\perp^1}{2} & \frac{T_\parallel^1 + T_\perp^1}{2} & 0 & 0 \\ 0 & 0 & \sqrt{T_\parallel^1 T_\perp^1} & 0 \\ 0 & 0 & 0 & \sqrt{T_\parallel^1 T_\perp^1} \end{bmatrix},\qquad(9)$$

$$\mathbf{R}_2 = \begin{bmatrix} \frac{R_\parallel^2 + R_\perp^2}{2} & \frac{R_\parallel^2 - R_\perp^2}{2} & 0 & 0 \\ \frac{R_\parallel^2 - R_\perp^2}{2} & \frac{R_\parallel^2 + R_\perp^2}{2} & 0 & 0 \\ 0 & 0 & \sqrt{R_\parallel^2 R_\perp^2} & 0 \\ 0 & 0 & 0 & \sqrt{R_\parallel^2 R_\perp^2} \end{bmatrix},\qquad(10)$$

$$\mathbf{T}_2 = \begin{bmatrix} \frac{T_\parallel^2 + T_\perp^2}{2} & \frac{T_\parallel^2 - T_\perp^2}{2} & 0 & 0 \\ \frac{T_\parallel^2 - T_\perp^2}{2} & \frac{T_\parallel^2 + T_\perp^2}{2} & 0 & 0 \\ 0 & 0 & \sqrt{T_\parallel^2 T_\perp^2} & 0 \\ 0 & 0 & 0 & \sqrt{T_\parallel^2 T_\perp^2} \end{bmatrix}.\qquad(11)$$

2.3   Multi-bounce Reflection and Transmission

In this section, we will show a multi-bounce reflection between two double-surface mediums in detail. The reflected scene consists of direct reflection on the surface, and multiple light reflection that transmits out of medium with a polarization state change and spatial shift. The transmitted scene has a similar behavior but only contains multiple reflections between two medium surfaces. We will show how to trace the polarization state by the Stokes vector as well as the Muller matrix step by step.

As illustrated in Fig. 2(a) and Fig. 2(d), in the local coordinate frame of the transparent object, the first directed reflected scene and transmitted scene can be formulated as,

$$\hat{\mathbf{s}}_r^1(x) = \mathbf{R}_1 \hat{\mathbf{s}}_r^0(x),\qquad(12)$$

$$\hat{\mathbf{s}}_t^1(x) = \mathbf{T}_2 \mathbf{R}_2 \mathbf{T}_1 \hat{\mathbf{s}}_t^0(x),$$

$$\mathbf{s}^1(x) = \mathbf{C}\left(\psi\right)(\hat{\mathbf{s}}_r^1(x) + \hat{\mathbf{s}}_t^1(x)).$$

(a) Direct Ref.   (b) 2nd Ref.   (c) 3rd Ref.

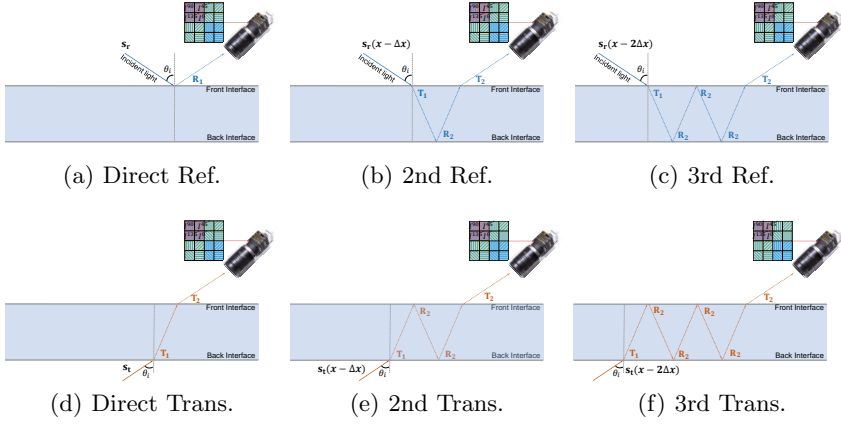(d) Direct Trans.   (e) 2nd Trans.   (f) 3rd Trans.

Fig. 2: The illustration of multi-bounce reflection between double surface.

Transmission and reflection with a second time bounce (Fig. 2(b) and Fig. 2(e)),

$$
\begin{aligned}
\hat{\mathbf{s}}_r^2(x) &= \mathbf{R}_1 \hat{\mathbf{s}}_r^0(x) \\
&\quad + \mathbf{T}_2 \mathbf{R}_2 \mathbf{T}_1 \hat{\mathbf{s}}_r^0(x - \Delta x), \\
\hat{\mathbf{s}}_t^2(x) &= \mathbf{T}_2 \mathbf{R}_2 \mathbf{T}_1 \hat{\mathbf{s}}_t^0(x) \\
&\quad + \mathbf{T}_2 \mathbf{R}_2^2 \mathbf{T}_1 \hat{\mathbf{s}}_t^0(x - \Delta x), \\
\mathbf{s}^2(x) &= \mathbf{C}\left(\psi\right)(\hat{\mathbf{s}}_r^2(x) + \hat{\mathbf{s}}_t^2(x)).
\end{aligned}
\tag{13}
$$

A third time bounce (Fig. 2(c) and Fig. 2(f)),

$$
\begin{aligned}
\hat{\mathbf{s}}_r^3(x) &= \mathbf{R}_1 \hat{\mathbf{s}}_r^0(x) \\
&\quad + \mathbf{T}_2 \mathbf{R}_2 \mathbf{T}_1 \hat{\mathbf{s}}_r^0(x - \Delta x) \\
&\quad + \mathbf{T}_2 \mathbf{R}_2^3 \mathbf{T}_1 \hat{\mathbf{s}}_r^0(x - 2\Delta x), \\
\hat{\mathbf{s}}_t^3(x) &= \mathbf{T}_2 \mathbf{R}_2 \mathbf{T}_1 \hat{\mathbf{s}}_t^0(x) \\
&\quad + \mathbf{T}_2 \mathbf{R}_2^2 \mathbf{T}_1 \hat{\mathbf{s}}_t^0(x - \Delta x) \\
&\quad + \mathbf{T}_2 \mathbf{R}_2^4 \mathbf{T}_1 \hat{\mathbf{s}}_t^0(x - 2\Delta x), \\
\mathbf{s}^3(x) &= \mathbf{C}\left(\psi\right)(\hat{\mathbf{s}}_r^3(x) + \hat{\mathbf{s}}_t^3(x)).
\end{aligned}
\tag{14}
$$

In local coordinate frame of the transparent object, a complete description is

$$
\hat{\mathbf{s}}_r(x) = \mathbf{R}_1 \hat{\mathbf{s}}_r^0(x) + \sum_{i=0}^{\infty} \mathbf{T}_2 \mathbf{R}_2^{2i+1} \mathbf{T}_1 \hat{\mathbf{s}}_r^0(x - i \cdot \Delta x),
\tag{15}
$$

where $\Delta x = 2d \tan \theta_t \sin \theta_t$ is the spatial offset between ghost images (see Fig. 1b), and $\hat{\mathbf{s}}_r^0 = \mathbf{C}(-\psi)\mathbf{s}_r^0$ describes the polarization state of the reflected scene before interacting with the transparent object.

Likewise, the total contributions of the transmitted scene are given by the transmission into the object, followed by an even number of internal reflections (possibly zero), and the transmission out of the object:

$$\hat{\mathbf{s}}_t(x) = \sum_{i=0}^{\infty} \mathbf{T}_2 \mathbf{R}_2^{2i} \mathbf{T}_1 \hat{\mathbf{s}}_t^0(x - i \cdot \Delta x), \tag{16}$$

with $\hat{\mathbf{s}}_t^0 = \mathbf{C}(-\psi)\mathbf{s}_t^0$ being the initial polarization state. The total light imaged by the camera can then be described in global/camera coordinates as

$$\mathbf{s} = \mathbf{C}(\psi)(\hat{\mathbf{s}}_r(x) + \hat{\mathbf{s}}_t(x)). \tag{17}$$

2.4   Pytorch Implementation of Polarization Simulator Engine

In this section, we show the details of the implementation of PSE, and the combination of a popular deep learning framework.

Spatial Shift is not always differentiable for any given parameters. To enable end-to-end training with a physical simulator, we slightly modify the convolution operator to implement spatial translation by given parameters. Spatial translation can be considered as a special convolution that has only one element to be 1 in the kernel and other kernel elements are zero. Other operators in the simulator can directly call PyTorch inline functions or libs. In the code, we simply set 20 pixels as maximal displacement.

```
def shift(self, I):
    tx = np.int(2*self.d*np.tan(self.theta_in)*np.sin(self.theta_in))
    filternp = np.zeros([3, 1, 41, 41])
    filternp[:, :, 20, 20 - tx] = 1
    filter = torch.Tensor(filternp).cuda()
    I_trans = torch.nn.functional.conv2d(I, filter, groups=3, padding
    return I_trans
```

3   Dataset

Fig. 3 shows synthetic ghost image by our polarization simulation engine and captured real images. The zoom-in figure can clearly see the multi-reflection phenomenon, and we can also observe that multiple reflections between the surface will dramatically decrease its intensity, therefore, it almost invisible after the second reflection. Our simulator can accurately reproduce real scene ghost image. Moreover, we give a glimpse of our real scene dataset with ground-truth as well as different reflected scenes (Fig. 9).

3.1   Visual Details Comparison

In Fig. 4 and Fig. 5, we show the details of reflection removal when comparing with original image for real cases. In Fig. 6, we show the details of real scene

ghost images, and our removal results in details. In Fig. 7 and Fig. 8, we show the details of reflection removal for synthetic cases. We simulate the polarizer's effect on diffuse scene that makes scene darker, i.e., total intensity will reduce due to extra polarizer block some light. Our proposed method can rectify the intensity and color of recovered image, to match the color and intensity distribution of ground-truth. For the real cases, we increase polarization camera's exposure time to make the intensity of polarized images to be as similar as normal image, therefore, no extra color rectification is needed.

References

1. Miyazaki, D., Ikeuchi, K.: Inverse polarization raytracing: estimating surface shapes of transparent objects. In: CVPR. (2005)

Fig. 3: Ghost image of synthetic (first row) and real (second row) examples, and our system setup (third row).
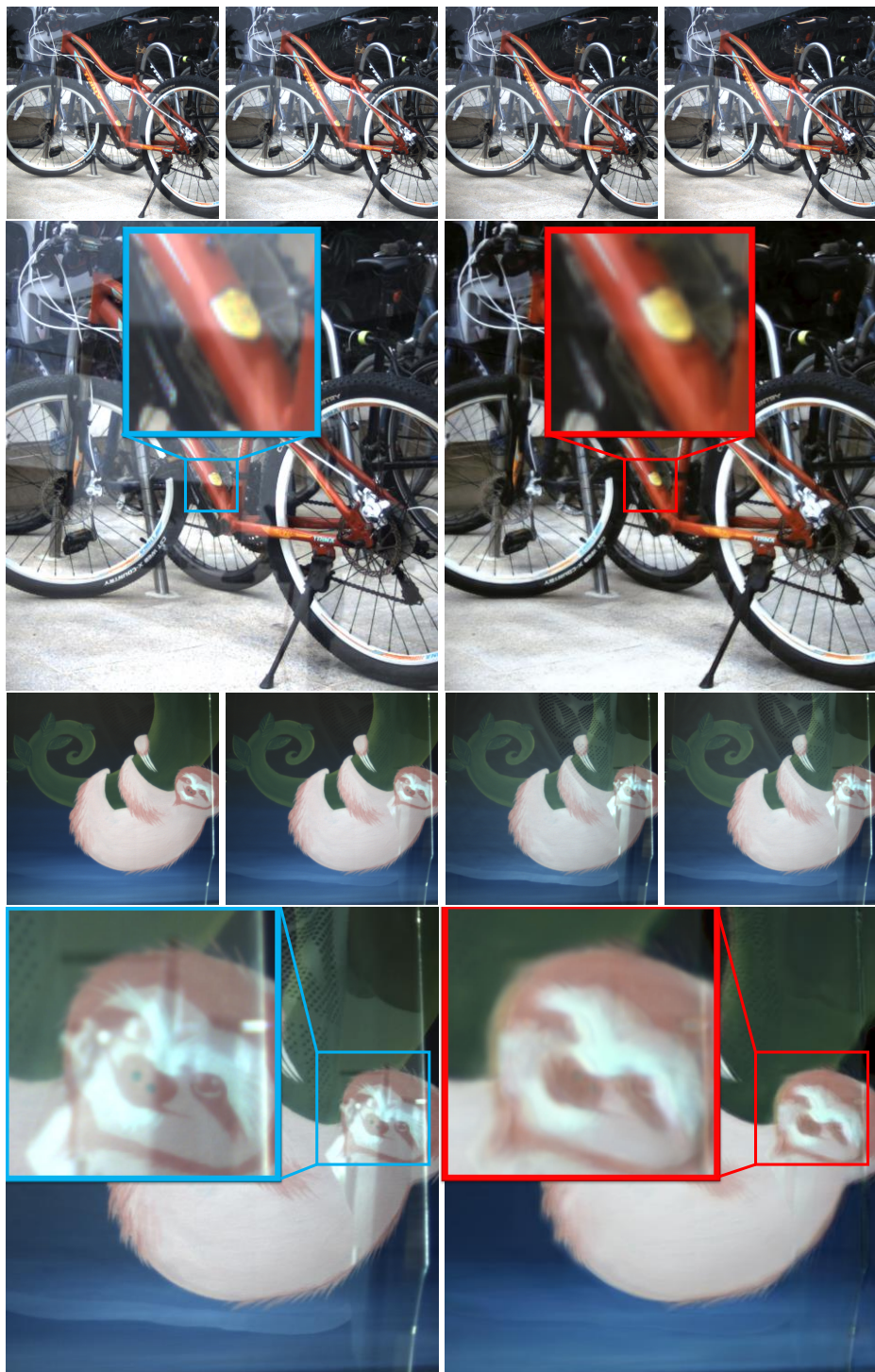
Fig. 4: Bike and monkey. First row: 4 polarized image, second row: detailed comparison of our methods.
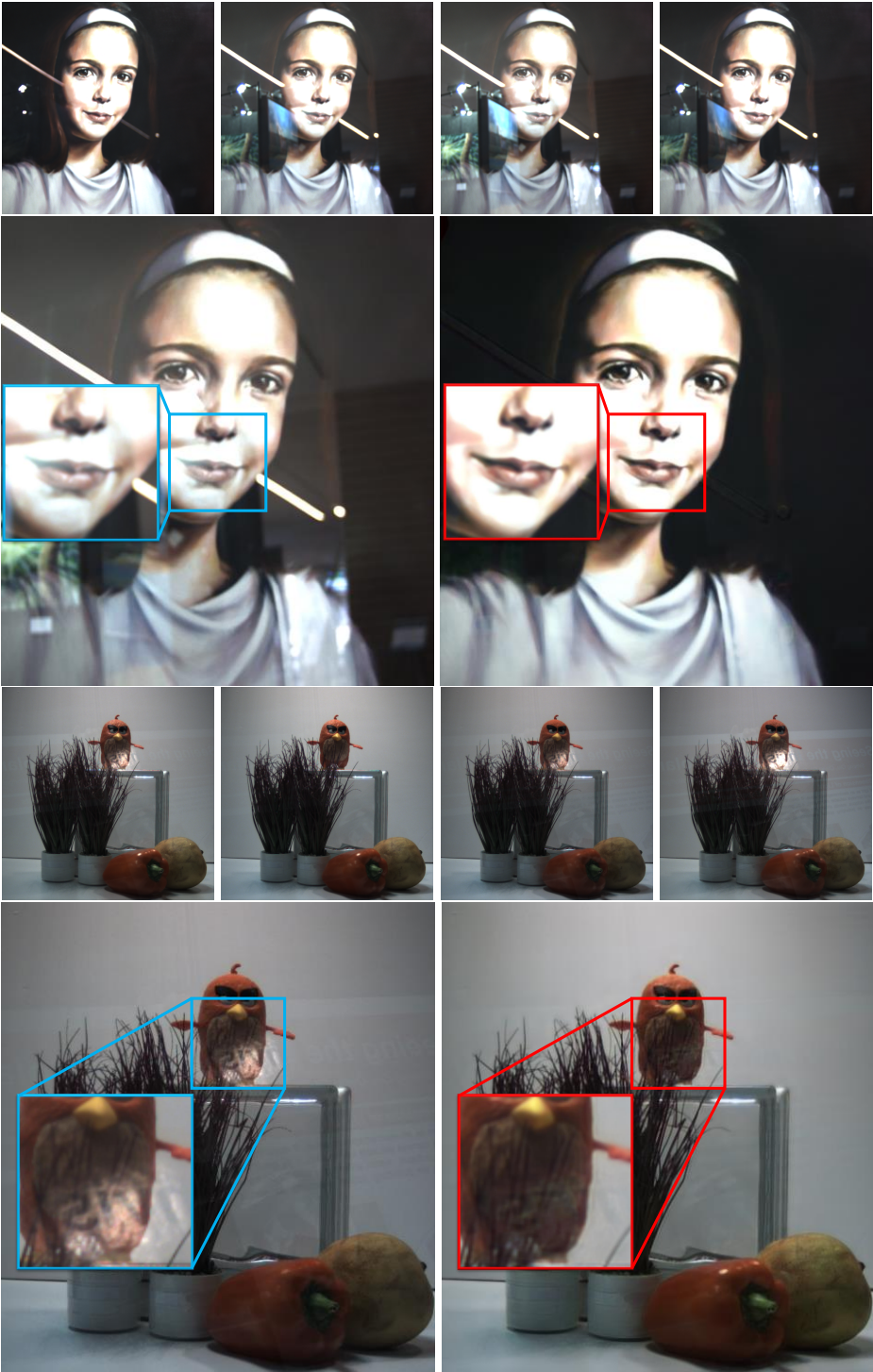
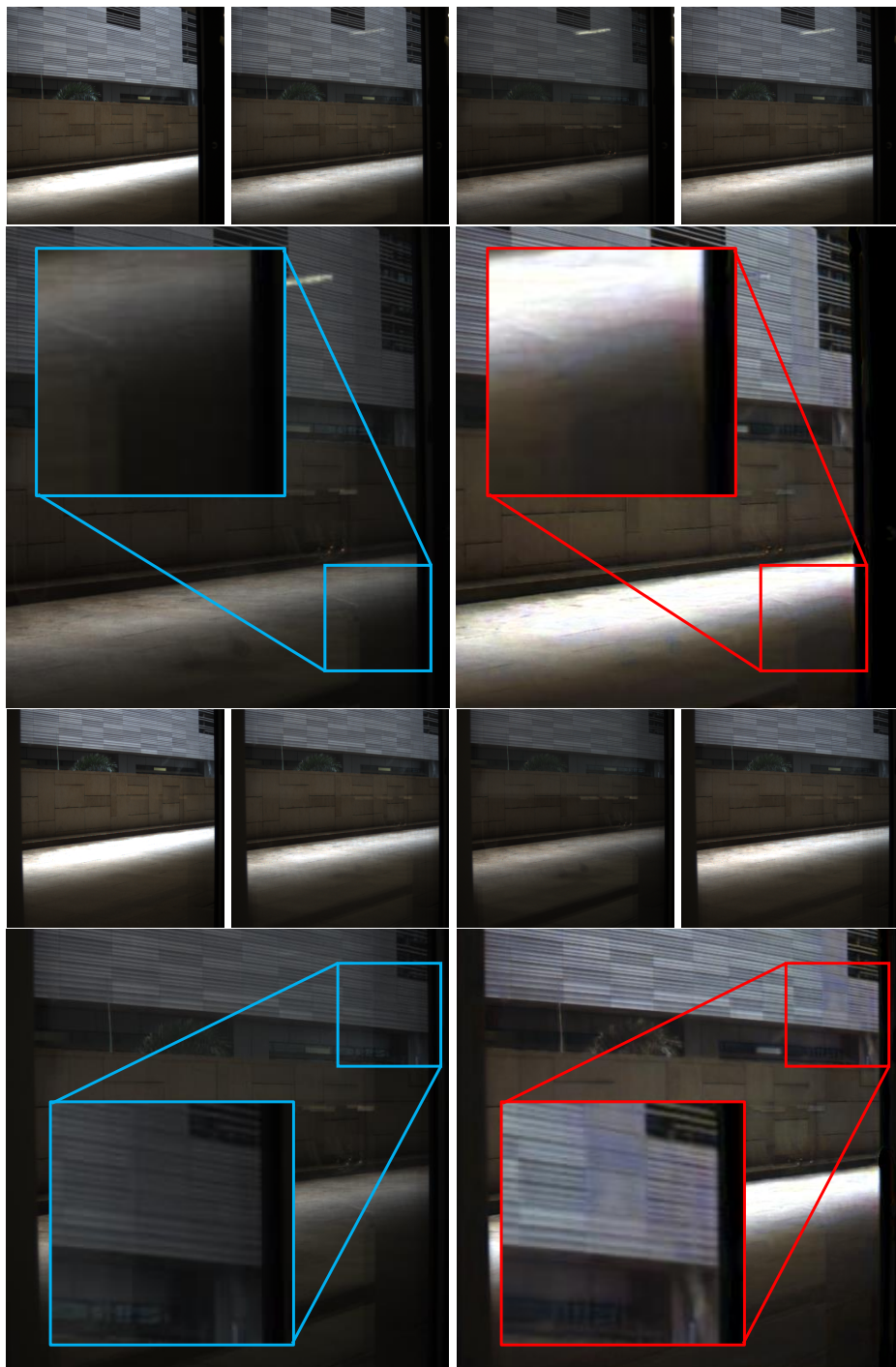Fig. 5: Girl and bird. First row: 4 polarized image, second row: detailed comparison of our methods.

Fig. 6: Ghost effect removal. First row: 4 polarized image, second row: detailed comparison of our methods.

Fig. 7: Synthetic examples. First row: 4 polarized image, second row: detailed comparison of our methods.
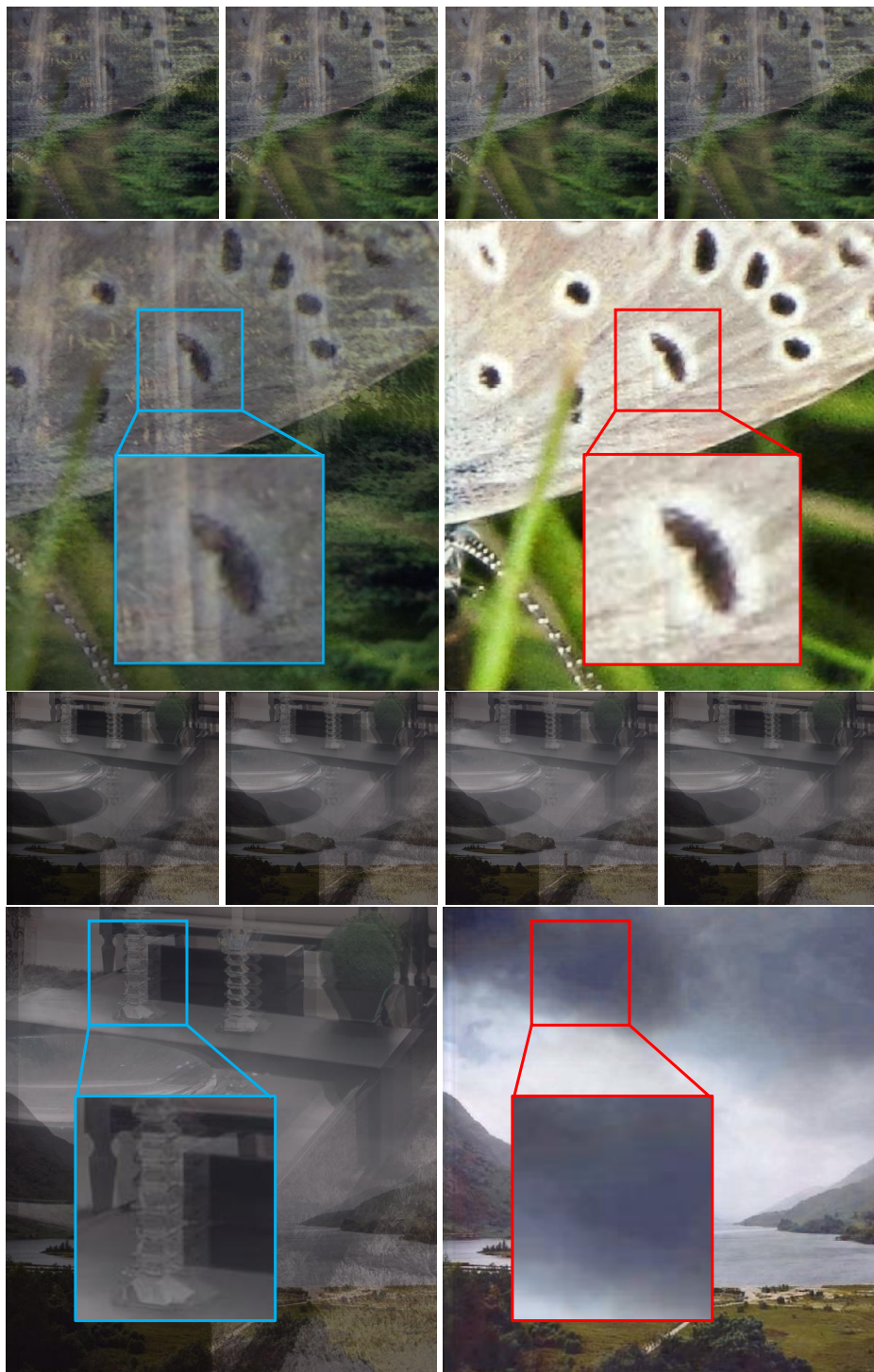
Fig. 8: Synthetic examples. First row: 4 polarized image, second row: detailed comparison of our methods.
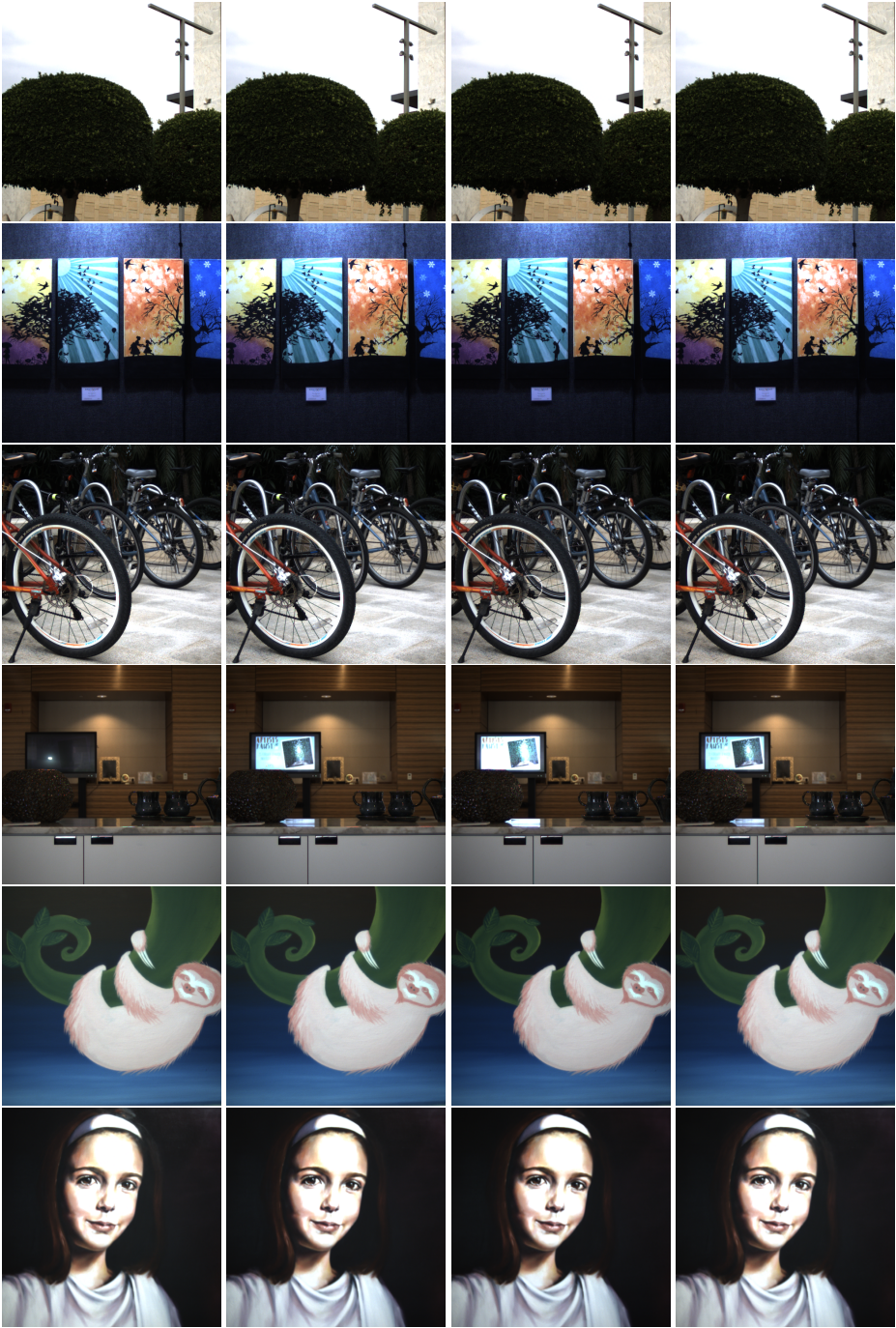
Fig. 9: The example of clear scene without glass. The polarized images are captured by polarization camera with 4 different angles of $\{0°, 45°, 90°, 135°\}$.
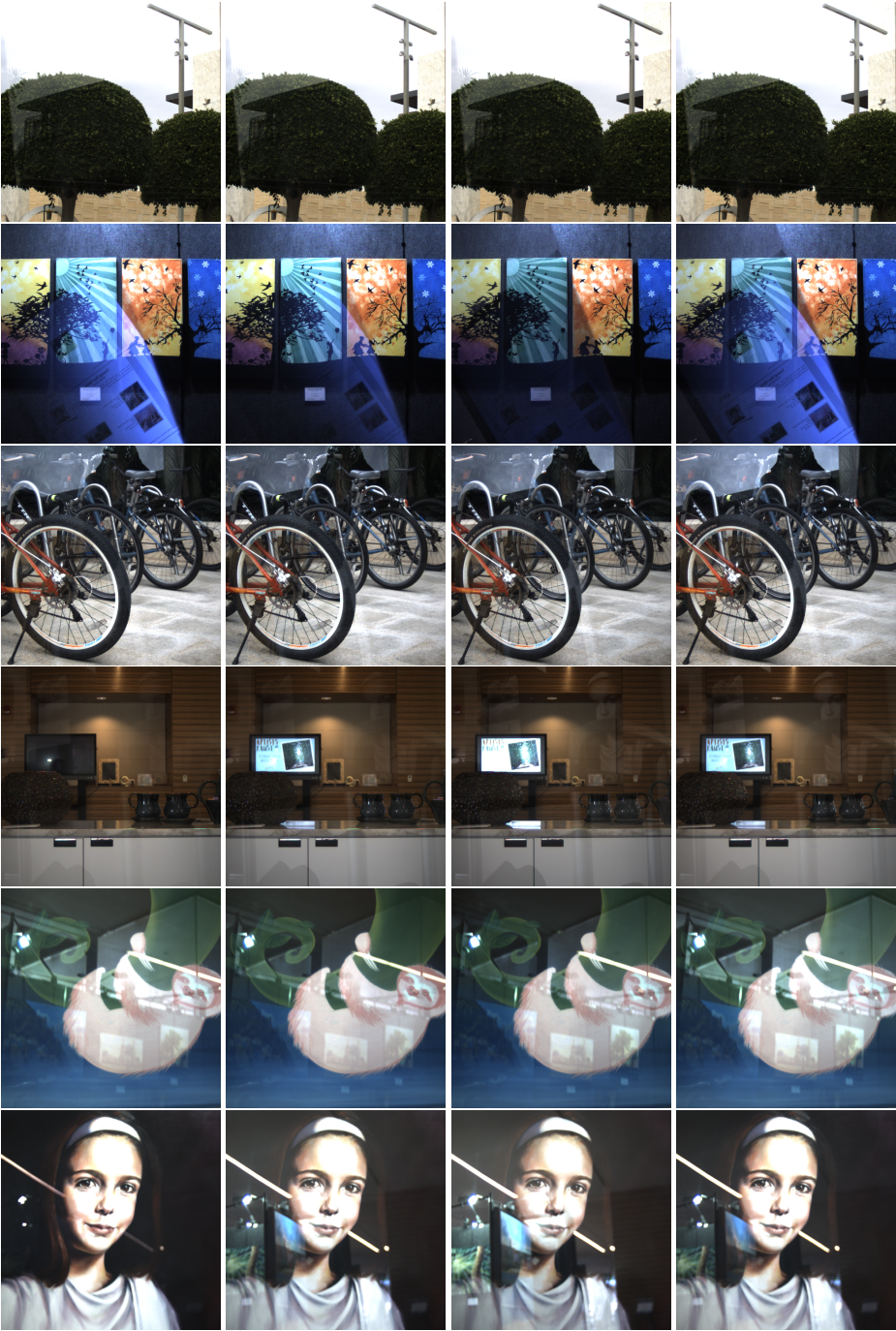
Fig. 10: The example of different glass reflected scenes. The polarized images are captured by polarization camera with 4 different angles of $\{0°, 45°, 90°, 135°\}$.