OmniZoomer: Learning to Move and Zoom in on Sphere at High-Resolution

Zidong Cao1 Hao Ai1,2* Yan-Pei Cao2 Ying Shan2 Xiaohu Qie2 Lin Wang1,3†
1 AI Thrust, HKUST(GZ) 2 ARC Lab, Tencent PCG 3 Dept. of CSE, HKUST
caozidong1996@gmail.com, hai033@connect.hkust-gz.edu.cn, caoyanpei@gmail.com
yingshan@tencent.com, tigerqie@tencent.com, linwang@ust.hk

Figure 1: From top to bottom: equirectangular projection (ERP) of omnidirectional images, perspective projection from ERP images with a specific field-of-view (FoV), w.r.t. the blue regions, and results from LAU-Net, OmniZoomer and ground truth. From left to right: increasing zoom levels. Our approach can move to the object of interest, and freely zoom in and zoom out on omnidirectional images, which can recover clear and preserved textural details with increasing zoom levels.

Abstract

Omnidirectional images (ODIs) have become increasingly popular, as their large field-of-view (FoV) can offer viewers the chance to freely choose the view directions in immersive environments such as virtual reality. The Möbius transformation is typically employed to further provide the opportunity for movement and zoom on ODIs, but applying it to the image level often results in blurry effect and aliasing problem. In this paper, we propose a novel deep learning-based approach, called OmniZoomer, to incorporate the Möbius transformation into the network for movement and zoom on ODIs. By learning various transformed feature maps under different conditions, the network is enhanced to handle the increasing edge curvatures, which alleviates the blurry effect. Moreover, to address the aliasing problem, we propose two key components. Firstly, to compensate for the lack of pixels for describing curves, we enhance the feature maps in the high-resolution (HR) space and calculate the transformed index map with a spatial index generation module. Secondly, considering that ODIs are inherently represented in the spherical space, we propose a spherical resampling module that combines the index map and HR feature maps to transform the feature maps for better spherical correlation. The transformed feature maps are decoded to output a zoomed ODI. Experiments show that our method can produce HR and high-quality ODIs with the flexibility to move and zoom in to the object of interest. Project page is available at http://vlislab22.github.io/OmniZoomer/.

1. Introduction

Omnidirectional images (ODIs) have garnered significant attention as a means to maximize the amount of content and context captured within a single image, and
there is a growing demand for utilizing such visual content within devices, e.g., mobile apps and head-mounted displays (HMDs) for virtual reality (VR) [38]. To provide an interactive experience, these devices enable users to control the view direction. However, most 360° cameras have a fixed focal length and do not support optical zoom, which causes the apparent size of objects in ODIs fixed. This limits the immersive experience when users expect to move and zoom in to an object of interest to see more details.

Generally, there exist three solutions to zoom in on the equirectangular projection (ERP) format ODIs or their perspective patches. The first is to zoom in on ERP images uniformly. However, as ERP images have non-uniform pixel density in different latitudes [8], uniform zoom can severely distort the object shapes. The second is to zoom in on perspective patches projected from ODIs. As the perspective patches of ODI have uniform pixel density [10], distortion problem can be solved. However, due to the limited FoV, these patches only concentrate on local regions and ignore the relationship between each other during transformations. Thirdly, Möbius transformation has recently been employed to provide movement and zoom freedom on ODIs [34, 15, 27]. It is the only conformal bijective transformation on the sphere that preserves angles. However, applying Möbius transformation on the image level often leads to blurry and aliasing problems due to two reasons. Firstly, zoom-in makes a portion of the ODIs enlarged, making the enlarged region blurry and pixelated. Moreover, if 360° cameras are placed vertically, the ODIs suffer from distortion mainly in high-latitude regions and remain lots of straight lines in equator regions. After transformations, the appearance of the vertically captured ODIs varies greatly, resulting in more curves in both high-latitude and equator regions (See Fig. 2). Describing these curves with the same amount of pixels that originally represent straight lines becomes challenging.

To obtain high-quality ODIs after movement and zoom, in this paper, we propose a novel deep learning-based approach, dubbed OmniZoomer, to incorporate the Möbius transformation into the network for freely moving and zooming in on ODIs, as shown in Fig. 1. By learning transformed feature maps in various conditions, the network is enhanced to handle the increasing curves caused by movement and zoom, as well as the inherent spherical distortion in ODIs. In this case, the blurry effect can be solved to some extent, but the aliasing problem still exists, such as edge discontinuity and shape distortion (See Fig. 9(c)).

To further address the aliasing problem, we propose two key components. Firstly, to compensate for the lack of pixels for describing curves, we propose to enhance the extracted feature maps to high-resolution (HR) space before the transformation. The HR feature maps contain more fine-grained textural details, and are sufficient to represent the increasing curvatures and maintain the object shapes precisely. We then propose a spatial index generation module (Sec. 3.2) to calculate the transformed index map based on the HR feature maps and Möbius transformation matrix, which can be conducted on the HR feature space. Although applying Möbius transformation on HR images with existing super-resolution (SR) methods [24, 40, 44] can serve the same purpose, this solution is sub-optimal because the models might not handle the increasing curves (See Tab. 1 and Fig. 6). In addition, some image warping methods [36, 22] can learn the warping process in the network but are constrained to estimate spatial-varying grids on the 2D plane, rather than the sphere. There are also some SR models designed for ODIs [8, 43]. However, they are limited to vertically captured ODIs or predetermined data structures.

Subsequently, we propose a spherical resampling module that combines the HR feature maps and transformed index maps for feature map transformation. The spherical resampling is inspired by the inherent spherical representation of the ODIs and the spherical conformity of Möbius transformation. It resamples based on the spherical geodesic of two points on the sphere, which better relates the original HR feature maps and transformed ones. With HR feature representation and the spherical resampling module, OmniZoomer alleviates the blurry effect and aliasing problem substantially, enabling moving and zooming in to an object of interest on ODIs with preserved shapes and continuous curves. Finally, these feature maps are processed with a decoder to output a zoomed ODI. After movement and zoom, OmniZoomer can generate more precise visual results with clear textural and structural details (See Fig. 2).

As collecting real-world ODI pairs under Möbius transformation is difficult, we propose a dataset based on ODISR dataset [8], dubbed ODIM dataset, containing synthesized ODIs with various Möbius transformations. We evaluate the effectiveness of OmniZoomer on the ODIM dataset under various Möbius transformations and up-sampling factors. The experimental results show that OmniZoomer outperforms existing methods quantitatively and qualitatively.
The main contributions of this paper can be summarized as follows: (I) We propose a novel deep learning-based approach, called OmniZoomer, to incorporate the Möbius transformation into the deep network. (II) We enhance the feature maps to HR space and calculate the HR index map with a spatial index generation module. We also propose a spherical resampling module for better spherical correlation. (III) We establish ODIM dataset for supervised training. Compared with existing methods, OmniZoomer achieves the state-of-the-art performance under various Möbius transformations and up-sampling factors.

2. Related Work

Application of Möbius Transformation. One of the main immersive experience in 360° devices is the control by the viewers. Although current devices can provide the opportunity to control view directions and field-of-views (FoVs) [5], the zoom quality needs to be improved to see more details [33]. Möbius transformation has been applied on ODIs, including straight line rectification [32, 15, 14], stereo pairs rectification [16], and rotation and zoom [34]. However, these methods operate on the image level, whose performance relies heavily on the quality of the raw ODI.

Recently, in [41], Möbius transformation has been employed in deep learning for feature augmentation. Especially, [41] fuses the features, which are applied multiple transformations to predict the raw ODI and address the spherical distortion. However, [41] does not generate the HR and high-quality transformed ODIs. Moreover, Möbius transformation has also been applied for data augmentation [45], activation function [25, 31], pose estimation [3], and convolutions [26]. Although Möbius convolution [26] shows a strong capability of spherical equivalence, it requires the spherical harmonic transform in each convolution block, resulting in low computational efficiency. In this work, we propose a learning-based approach to improve the textural and structural details of ODIs when moving and zooming in to an object of interest.

Image Warping. It is widely utilized in various tasks, e.g., optical flow estimation [6] and video SR [23]. Generally, it is conducted by calculating transformed spatial indices, and resampling information from the input images based on the transformed indices [18]. Considering jagging and blurry effects of image warping [36], SRWarp interprets the image warping as a spatially-varying SR problem and proposes an adaptive warping layer to estimate the rotation during warping. SRWarp also shows that simply concatenating existing SR models [24, 40, 44, 17] with warping operation is sub-optimal. Furthermore, LTEW [22] estimates the varied shape and integrates the priors into an implicit representation in the Fourier space. Differently, we focus on transforming and resampling on the sphere with a curved surface. Our proposed spherical resampling module outperforms these warping methods significantly (See Tab. 3).

ODI Super-Resolution. Traditional ODI SR methods primarily utilize a sequence of low-resolution (LR) ODIs to stitch an HR ODI [1, 2, 4, 20, 28]. Recently, [12] and [29] propose learning-based SR methods that incorporate the distortion maps to tackle spherical distortions. [30] employs adversarial learning for ODI SR, but only treats ODIs as 2D planar images. Observing that different latitudes have non-uniform pixel densities, LAU-Net [8] crops ODIs into different latitude bands and dynamically up-samples these...
bands. However, after transformations, e.g., movement and zoom, the bands can not be simply cropped along latitudes. SphereSR [43] proposes to super-resolve an LR ODI to an HR ODI with arbitrary projection types. Nevertheless, the predetermined spherical data structure can not adapt to the transformed ODIs. Unlike these methods, we address a new task of incorporating the Möbius transformation into the network to move and zoom in to the object of interest on ODIs with high-quality textural and structural details.

3. Methodology

Overview. As shown in Fig. 3, we propose a novel end-to-end pipeline, dubbed OmniZoomer, which allows for free movement of the “eyes” to objects of interest and zooming in directly on the sphere with preserved shapes and high-quality textural details. Firstly, we extract HR feature maps \( F_{\text{IN}} \in \mathbb{R}^{H \times W \times C} \) from the input ODI \( I_{\text{IN}} \in \mathbb{R}^{h \times w \times 3} \) through an encoder and an up-sampling block (Sec. 3.1). With \( F_{\text{UP}} \)'s index map \( X \in \mathbb{R}^{H \times W \times 2} \) as the input, we propose the spatial index generation module (Sec. 3.2) to apply the Möbius transformation [19] with arbitrary parameters on \( X \) for the transformed spatial index map \( Y \in \mathbb{R}^{H \times W \times 2} \). Note that the channel numbers of \( X \) and \( Y \) indicate the longitude and latitude, respectively. Subsequently, we introduce a spherical resampling module (Sec. 3.3) that generates the transformed HR feature maps \( F_M \in \mathbb{R}^{H \times W \times C} \) by resampling the pixels on the sphere guided by the transformed index map \( Y \). Finally, we decode the feature maps to output a zoomed-in ODI where the region of interest is a clear close-up shot. The decoder consists of three Res-Blocks [24] and a convolution layer. We take the same parameters used in the spatial index generation module to transform the HR ground truth ODIs, and employ the \( L1 \) loss as the supervision loss. We now provide detailed descriptions of these components.

3.1. Feature Extraction

Given an ODI \( I_{\text{IN}} \in \mathbb{R}^{h \times w \times 3} \) with the ERP format, we first apply an encoder consisting of several convolution layers to extract the feature maps \( F_{\text{IN}} \in \mathbb{R}^{H \times W \times C} \). Accordingly, we design an upsampling block with several pixel-shuffle layers [35] to generate the HR feature maps \( F_{\text{UP}} \in \mathbb{R}^{H \times W \times C} \), where \( H = s \times h \), \( W = s \times w \), \( s \) is the scale factor and \( C \) is the channel number. Especially, we apply the Möbius transformation on the HR feature maps based on two considerations: 1) The blurry effect on image level. By learning various transformations, the extracted feature maps demonstrate an enhanced representation capability in handling increasing edge curvatures and solving the blurry effect. 2) The aliasing problem. For instance, in Fig. 4(a), due to the insufficient pixels to describe continuous and clear curves after transformations, the shape of the railing is distorted. Moreover, the aliasing problem is challenging to tackle even if super-resolving the transformed ODIs. In contrast, applying the Möbius transformation to the super-resolved ODI has a significant improvement, as demonstrated in Fig. 4(b).

3.2. Spatial Index Generation

In this section, we apply the Möbius transformation on the spatial index map \( X \) of HR feature maps \( F_{\text{UP}} \) and generate the transformed spatial index map \( Y \) for the subsequent resampling operation. Möbius transformation is known as the only conformal bijective transformation between the sphere and the complex plane. To apply the Möbius transformation on the HR feature maps \( F_{\text{UP}} \), we first use spherical projection (SP) to project the spatial index map \( X \) from spherical coordinates \((\theta, \phi)\) (where \( \theta \) represents the longitude and \( \phi \) represents the latitude) to the Riemann sphere \( \mathbb{S}^2 = \{(x, y, z) \in \mathbb{C}^3 | x^2 + y^2 + z^2 = 1\} \), formulated as:

\[
\text{SP} : \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos(\phi) \cos(\theta) \\ \cos(\phi) \sin(\theta) \\ \sin(\phi) \end{pmatrix}.
\]

(1)

Then, using stereographic projection (STP) [11], we can project a point \((x, y, z)\) of the Riemann sphere \( \mathbb{S}^2 \) onto the complex plane and obtain the projected point \((x', y')\). Let point \((0, 0, 1)\) be the pole, STP can be formulated as:

\[
\text{STP} : x' = \frac{x}{1-z} \quad y' = \frac{y}{1-z}.
\]

(2)

Subsequently, given the projected point \( p (Z_p = x'+iy') \) on the complex plane, we can conduct the Möbius transformation with the following formulation:

\[
f(Z_p) = \frac{az_p + b}{cz_p + d}.
\]

(3)

where \( a, b, c, \) and \( d \) are complex numbers satisfying \( ad - bc \neq 0 \). Finally, we apply the inverse stereographic projection \( \text{STP}^{-1} \) and inverse spherical projection \( \text{SP}^{-1} \) to
Figure 5: (a) Linear resampling is related to the the partial area \(S_i\), diagonally opposite to the corner pixel \(i\. (b) Spherical resampling considers the angles (\(i.e., \alpha_01, \alpha_23, \Omega\)) between points on the sphere, which are corresponding to the red solid curves.

We re-project the complex plane into the ERP plane:

\[
\text{STP}^{-1} : \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 2x' \\ 2y' \\ 1+x'^2+y'^2 \end{pmatrix}; \quad (4)
\]

\[
\text{SP}^{-1} : \begin{pmatrix} \theta \\ \phi \end{pmatrix} = \begin{pmatrix} \arctan(y/x) \\ \arcsin(z) \end{pmatrix}. \]

In summary, as shown in Fig. 3, we first project the input index map \(X\) to the complex plane using \(\text{SP}\) (Eq. 1) and \(\text{STP}\) (Eq. 2), and then conduct the Möbius transformation with Eq. 3, and generate the transformed index map \(Y\) through the inverse \(\text{STP}\) (Eq. 4) and inverse \(\text{SP}\) (Eq. 4). After transformation, both the indices (represented with matrix and gradient color) and grid shapes (represented with black lines) in \(Y\) have a noticeable change compared with \(X\).

### 3.3. Spherical Resampling

As spatial indices recorded in \(Y\) are not equidistant, it is necessary to design a resampling method to calculate the feature values for the transformed feature maps \(F_M\) based on \(Y\). Generally, the resampling process can be divided into three steps. The first step is to determine the neighboring pixel set \(N_q\) of the query pixel \(q\), \(i.e.,\) the four corner pixels \(\{p_i \in N_q, i = 0, 1, 2, 3\}\), as illustrated in Fig. 5(a). The second step is to calculate the weight \(w_{q,p}\) for each neighboring pixel \(p_i \in N_q\), \(i.e.,\) the partial area \(S_i\). The third step is to calculate the weighted average of neighboring feature values: 

\[
F(q) = \sum_{p_i \in N_q} w_{q,p} F(i). \]

Previous image warping methods, \(e.g.,\) SRWarp [36] and LTEW [22], consider the rotation of the local varied grid with Jacobian matrix. In this case, the resampling bases are recalculated according to the grid rotation, and the resampling weight \(w_{q,p}\) is re-projected to the new bases, either explicitly [36] or implicitly [22]. Although these methods can be used directly for Möbius transformation on ODIs, they cannot deal with the spherical representation of ODIs due to two key reasons: 1) As shown in Fig. 5(a), the partial area (marked with blue region) in 2D plane is stretched non-uniformly when projected to the spherical surface due to spherical distortion; 2) Although the resampling bases can be corrected with rotation, the resampling process is still limited to 2D plane, which is sub-optimal for describing the relationship between two points on the sphere.

Inspired by the inherent spherical representation of ODIs and the spherical conformality of Möbius transformation, we propose the spherical resampling module to generate the transformed feature maps \(F_M\). The spherical resampling module directly resamples on the curved sphere based on the spherical geodesic of two points on the sphere. Given a query pixel \(q\) with the spatial index \((\theta_q, \phi_q)\) from the index map \(Y\), we choose its four corner pixels \(\{p_i, i = 0, 1, 2, 3\}\) as the neighboring pixels, which are located on the feature maps \(F_{UP}\) (as shown in the left of Fig. 5(b)). The indices of the neighboring pixels satisfy the following conditions: \(\theta_0 = \theta_3, \theta_1 = \theta_2, \phi_0 = \phi_1, \phi_2 = \phi_3\). To obtain the feature value of the query pixel \(q\), we employ the spherical linear interpolation (Slerp) [13], which is a constant-speed motion along the spherical geodesic of two points on the sphere, formulated as follows:

\[
\text{Slerp}(a, b) = \frac{\sin(1-t)\beta}{\sin \beta} a + \frac{\sin t\beta}{\sin \beta} b, \quad (5)
\]

where \(\beta\) is the angle subtended by \(a\) and \(b\), and \(t\) is the resampling weight. Note that \(t\) is easy to determine if \(a\) and \(b\) are located on the same longitude. Therefore, we calculate the feature value of pixel \(q\) with two steps. Firstly, we resample \(p_0, p_1, p_2, p_3\) to \(p_0, p_1\), and \(p_2, p_3\), respectively, as shown in the right of Fig. 5(b). Taking the resampling of \(p_{0,1}\) as example, the formulation can be described as:

\[
F(p_{01}) = \frac{\sin(1-t_{01})\alpha_{01}}{\sin \alpha_{01}} F(p_0) + \frac{\sin t_{01}\alpha_{01}}{\sin \alpha_{01}} F(p_1), \quad (6)
\]

where \(\alpha_{01}\) is the angle subtended by \(p_0\) and \(p_1\), and the weight \(t_{01}\) is decided by the location of \(p_0\) on the curve \(p_0p_1\). Notably, \(t_{01}\) should ensure \(p_0\) to have the same longitude with the query pixel \(q\). Similarly, \(\alpha_{23}\) is the angle subtended by \(p_2\) and \(p_3\), and \(t_{23}\) also has the same longitude with the query pixel \(q\) by calculating the weight \(t_{23}\). After that, we follow the Slerp (Eq. 5) to calculate the feature value \(F_q\) as follows:

\[
F(q) = \frac{\sin(1-t_q)\Omega}{\sin \Omega} F(p_{01}) + \frac{\sin t_q\Omega}{\sin \Omega} F(p_{23}), \quad (7)
\]

where \(\Omega\) is the angle subtended by \(p_{01}\) and \(p_{23}\), and \(t_q\) is decided by the location of \(q\) on the curve \(p_{01}p_{23}\). Due to
Table 1: Quantitative comparison of Möbius transformation results on ODIs. (+Transform) denotes that we first employ a scale-specific SR model for image SR and then conduct image-level Möbius transformation on the SR image. We report on ODI-SR dataset and SUN360 dataset with up-sampling factors $\times 8$ and $\times 16$. **Bold** indicates the best results.

<table>
<thead>
<tr>
<th>Method</th>
<th>Scale</th>
<th>ODI-SR $\times 8$</th>
<th>SUN 360 $\times 8$</th>
<th>ODI-SR $\times 16$</th>
<th>SUN 360 $\times 16$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bicubic</td>
<td>26.77</td>
<td>0.7725</td>
<td>25.87</td>
<td>0.7103</td>
<td>24.79</td>
</tr>
<tr>
<td>EDSR-baseline (+Transform) [24]</td>
<td>27.42</td>
<td>0.7930</td>
<td>26.97</td>
<td>0.7468</td>
<td>25.39</td>
</tr>
<tr>
<td>Ours-EDSR-baseline</td>
<td>27.48</td>
<td>0.7949</td>
<td>27.15</td>
<td>0.7526</td>
<td>25.47</td>
</tr>
<tr>
<td>RRDB (+Transform) [40]</td>
<td>27.45</td>
<td>0.7946</td>
<td>27.10</td>
<td>0.7515</td>
<td>25.42</td>
</tr>
<tr>
<td>RCAN (+Transform) [44]</td>
<td>27.46</td>
<td>0.7906</td>
<td>27.04</td>
<td>0.7443</td>
<td>25.45</td>
</tr>
<tr>
<td>ETDS (+Transform) [7]</td>
<td>27.38</td>
<td>0.7912</td>
<td>26.84</td>
<td>0.7418</td>
<td>25.42</td>
</tr>
<tr>
<td>Omni-SR (+Transform) [39]</td>
<td>27.45</td>
<td>0.7920</td>
<td>26.99</td>
<td>0.7463</td>
<td>25.45</td>
</tr>
<tr>
<td>LAU-Net (+Transform) [8]</td>
<td>27.25</td>
<td>0.7813</td>
<td>26.77</td>
<td>0.7363</td>
<td>25.23</td>
</tr>
<tr>
<td>SRWarp [36]</td>
<td>27.43</td>
<td>0.7911</td>
<td>27.12</td>
<td>0.7495</td>
<td>25.40</td>
</tr>
<tr>
<td>LTEW [22]</td>
<td>27.32</td>
<td>0.7899</td>
<td>26.85</td>
<td>0.7420</td>
<td>25.39</td>
</tr>
<tr>
<td>Ours-RCAN</td>
<td>27.53</td>
<td>0.7970</td>
<td>27.34</td>
<td>0.7592</td>
<td>25.50</td>
</tr>
</tbody>
</table>

Figure 6: Visual comparisons of Möbius transformation results with $\times 8$ up-sampling factor on ODI-SR dataset.

the page limit, more formulations about the parameter $t_{01}$, $t_{23}$ and $t_q$ can be found in the supplementary material. Actually, our spherical resampling module calculates the angular relationship of query pixels and their corresponding corner pixels, which can better describe the resampling on the spherical surface with curvatures. Meanwhile, there is no need to estimate the transformed grid shape like [22], because Möbius transformation is conformal on the sphere that preserves the angles subtended by two curves.

4. Experiment

4.1. Dataset and Implementation Details

Datasets. No datasets for ODIs under Möbius transformations exist and collecting real-world ODI pairs with corresponding Möbius transformation matrices is difficult. Thus, we propose ODI-Möbius (ODIM) dataset to train our OmniZoomer and compared methods in a supervised manner. Our dataset is based on the ODI-SR dataset [9] with 1191 images in the train set, 100 images in the validation set, and 100 images in the test set. During training, we applied various Möbius transformations via setting random parameters $\{a, b, c, d\}$ of Eq. 3. Each Möbius transformation includes all horizontal rotation, vertical rotation, and zoom, as we aim to move and zoom in on ODIs. More details can be found in the Suppl material. During validating and testing, we assign a fixed Möbius transformation matrix for each ODI. Besides, we further test on SUN360 [42] dataset with 100 images.

Implementation details. The resolution of the HR ERP images is $1024 \times 2048$, and the up-sampling factors we choose are $\times 8$ and $\times 16$. We use L1 loss, which is optimized by Adam optimizer [21], with an initial learning rate of 1e-4. The batch size is 2 when using EDSR-baseline [24] as back-
<table>
<thead>
<tr>
<th>Method</th>
<th>Scale</th>
<th>ODI-SR</th>
<th>SUN 360</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>×8</td>
<td>×16</td>
<td>×8</td>
</tr>
<tr>
<td>Bicubic</td>
<td>19.64</td>
<td>0.5908</td>
<td>19.72</td>
</tr>
<tr>
<td>EDSR [24]</td>
<td>23.97</td>
<td>0.6483</td>
<td>23.79</td>
</tr>
<tr>
<td>RCAN [44]</td>
<td>24.26</td>
<td>0.6554</td>
<td>23.88</td>
</tr>
<tr>
<td>360-SS [30]</td>
<td>24.14</td>
<td>0.6539</td>
<td>24.19</td>
</tr>
<tr>
<td>SphereSR [43]</td>
<td>24.37</td>
<td>0.6777</td>
<td>24.17</td>
</tr>
<tr>
<td>LAU-Net [8]</td>
<td>24.36</td>
<td>0.6602</td>
<td>24.24</td>
</tr>
<tr>
<td>LAU-Net+ [9]</td>
<td><strong>24.63</strong></td>
<td><strong>0.6615</strong></td>
<td><strong>24.37</strong></td>
</tr>
<tr>
<td>Ours-EDSR-baseline</td>
<td>24.48</td>
<td>0.6756</td>
<td>24.31</td>
</tr>
<tr>
<td>Ours-RCAN</td>
<td><strong>24.53</strong></td>
<td><strong>0.6797</strong></td>
<td><strong>24.41</strong></td>
</tr>
</tbody>
</table>

Table 2: Quantitative comparison of ODI SR task. The numbers are excerpted from [9] except for [43], due to its reported results are obtained by utilizing 800 training images in the ODI-SR dataset. We report ×8, ×16 SR results on the ODI-SR and SUN360 datasets. Bold indicates the best results, and blue indicates the second-best results.

Figure 7: Visual comparisons of different methods for Möbius transformation with ×8 up-sampling factor on SUN360 dataset.

bone, while the batch size is 1 when using RCAN [44] as backbone. Especially, considering the spherical imagery of ODIs, we use specific WS-PSNR [37] and WS-SSIM [46] metrics for evaluation.

### 4.2. Quantitative and Qualitative Evaluation

**Move and Zoom in:** As OmniZoomer is the first learning-based method, there are no prior-arts that can be directly compared. For fair and sufficient evaluation, we design two types of comparative experiment. First, we combine the existing image SR models for 2D planar images and ODIs [24, 40, 44, 7, 39, 8] with image-level Möbius transformations, whose resampling process is achieved by nearest interpolation. This way, we compare OmniZoomer with these approaches under various Möbius transformations. The SR models designed for 2D planar images are retrained with their provided hyperparameters. Second, we further compare our OmniZoomer with existing image warping methods [36, 22] and incorporate Möbius transformations into their learning process.

Tab. 1 provides a quantitative comparison of different methods for various Möbius transformations with up-sampling factors ×8 and ×16. We use a lightweight backbone EDSR-baseline [24] and a deep backbone RCAN [44]. OmniZoomer with EDSR-baseline as backbone outperforms several 2D SR models with image-level transformations, e.g., EDSR-baseline [24], RRDB [40] and RCAN [44], in all metrics. It reveals the effectiveness of our OmniZoomer incorporating Möbius transformation into the feature representation. Compared with the ODI-specific SR method LAU-Net [8], our OmniZoomer also achieves better performance. Note that LAU-Net shows lower performance than SR models designed for 2D planar images, e.g., EDSR-baseline. We ascribe it to that LAU-Net is limited...
to only consider the ODIs captured with vertically placed 360° cameras, which have no movement and zoom. By applying deeper backbone RCAN, OmniZoomer outperforms existing methods in all metrics, all up-sampling factors, and test sets. For example, compared with LAU-Net [8], OmniZoomer has a 0.57dB improvement of WS-PSNR on SUN360 dataset with ×8 up-sampling factor. As shown in Fig. 6, our OmniZoomer predicts clearer wood strips with high-quality textural details with ×8 up-sampling factor, which are missing in other methods’ predictions. It shows the effectiveness of our HR feature representation and spherical resampling. Similarly, in Fig. 7, OmniZoomer reconstructs more complete structures of the building and preserves the shape of buildings after transformations.

**Direct SR:** Although our work shares a different purpose with the image SR methods, it can also perform SR when the Möbius transformation matrix is applied as the identity matrix. In this case, OmniZoomer is degraded to a conventional SR model, except for the spherical resampling module. Tab. 2 shows the quantitative results of OmniZoomer with two backbones, exhibiting that OmniZoomer with RCAN as backbone obtains 3 (total 8) best metrics, while OmniZoomer with EDSR-baseline as backbone obtains 2 (total 8) second best metrics. **OmniZoomer has a strong capability to handle the increasing curves and the inherent distortions on ODIs.**

### 4.3. Ablation Studies

**Spherical resampling module.** Tab. 3 illustrates that our spherical resampling module achieves the best performance compared with both applying traditional resampling algorithm (e.g., Bicubic), and estimated base rotation in the image warping method [36]. For example, our spherical resampling module obtains 0.07dB WS-PSNR gain compared with utilizing Jacobian matrix for base rotation. Also, by adding a multi-layer perceptron (MLP) to make the 2D rotation estimation learnable, the performance gain is limited (0.01dB). It is mainly because that [36] estimates the 2D rotation, which is not applicable for 3D rotation on the sphere surface. Compared with these planar resampling methods, spherical resampling module has obvious improvement benefiting from fitting the sphere surface with curvatures.
This can be verified qualitatively in Fig. 8, where spherical resampling recovers more continuous edges of the windmill. Furthermore, by adding ResBlocks [24] into the decoder, the transformed feature maps can be further refined. It brings 0.02dB gain in WS-PSNR metric.

**Different positions for Möbius transformation.** There are totally four possible positions to conduct Möbius transformation, i.e., the input image level, the input feature level, the HR feature level, and the HR image level (output of the network). Tab. 4 demonstrates that conducting Möbius transformation in the input image level and input feature level is not applicable, due to the severely destroyed structures, which are difficult to reconstruct in the HR space. Also, conducting Möbius transformation on the HR image level leads to sub-optimal results as the network has no knowledge to handle increasing edge curvatures in various transformations, e.g., movement and zoom. From Fig. 9, we can see that conducting Möbius transformation on the HR feature level recovers the clearest pipelines on the ceiling.

**Integrating feature up-sampling and Möbius transformation.** By integrating them into a whole, we find that the performance drops by about 0.07dB WS-PSNR. The reason is about the aliasing problem in the input feature level.

**Number of parameters.** Our OmniZoomer with EDSR-baseline as the backbone is compact and performs better than LAU-Net in Möbius transformation tasks. Also, with 0.1M extra parameters, OmniZoomer-RCAN achieves a significant performance gain than RCAN [40]. More details about computational costs and time consuming of each module can be found in the Suppl material.

### 5. Conclusion

In this paper, we proposed to incorporate the Möbius transformation into the network for freely moving and zooming in on ODIs. We found that ODIs under Möbius transformations suffer from blurry effect and aliasing problems due to zoomed-in regions and increasing edge curvatures. Based on the problems, we found that learning Möbius transformations on the HR feature level and resampling on the sphere surface enhance the network to predict clear curves and preserved shapes. We demonstrated that this deep learning-based network outperforms existing methods under various Möbius transformations. Therefore, OmniZoomer can produce HR and high-quality ODIs with the flexibility to move and zoom in to the object of interest.

**Limitation and Future Work:** This work can estimate HR and high-quality ODIs under various Möbius transformations. However, the parameters of the Möbius transformation need to be determined by users, according to the movement and zoom level. In this case, users might try for several times to determine the optimal transformation, which influences the interactive experiences. In the future work, we hope to learn to how to select an optimal transformation by only assigning the interested objects. This might include the techniques about omnidirectional object detection and scene understanding.

**Acknowledgement:** This work was supported by the CCF-Tencent Open Fund and the National Natural Science Foundation of China (NSFC) under Grant No. NSFC22FYT45.

### References


[33] Andre Potenya. de matemática pura e aplicada zoom for virtual reality based on m"obius transformations. 2018.


