Tangent Images for Mitigating Spherical Distortion

Marc Eder, Mykhailo Shvets, John Lim, and Jan-Michael Frahm
University of North Carolina at Chapel Hill
Chapel Hill, NC
{meder, mshvets, jlim13, jmf}@cs.unc.edu

Abstract

In this work, we propose "tangent images," a spherical image representation that facilitates transferable and scalable $360°$ *computer vision. Inspired by techniques in cartography and computer graphics, we render a spherical image to a set of distortion-mitigated, locally-planar image grids tangent to a subdivided icosahedron. By varying the resolution of these grids independently of the subdivision level, we can effectively represent high resolution spherical images while still benefiting from the low-distortion icosahedral spherical approximation. We show that training standard convolutional neural networks on tangent images compares favorably to the many specialized spherical convolutional kernels that have been developed, while also scaling efficiently to handle significantly higher spherical resolutions. Furthermore, because our approach does not require specialized kernels, we show that we can transfer networks trained on perspective images to spherical data without fine-tuning and with limited performance drop-off. Finally, we demonstrate that tangent images can be used to improve the quality of sparse feature detection on spherical images, illustrating its usefulness for traditional computer vision tasks like structure-from-motion and SLAM.*

1. Introduction

A number of methods have been proposed to address convolutions on spherical images. These techniques vary in design, encompassing learnable transformations [25, 26], generalizations and modifications of the convolution operation [8, 9, 11, 27], and specialized kernels for spherical representations [7, 16, 29]. In general, these spherical convolutions fall into two classes: those that operate on equirectangular projections and those that operate on a subdivided icosahedral representation of the sphere. The latter has been shown to significantly mitigate spherical distortion, which leads to significant improvements for dense prediction tasks [10, 11, 18]. It also has the useful property that icosahedron's faces and vertices scale roughly by a factor of 4 at each subdivision, permitting a simple analogy to $2\times$ upsampling and downsampling operations in standard convolutional neural networks (CNNs). Because of the performance improvements provided by the subdivided icosahedron representation, we focus expressly on it in this paper.

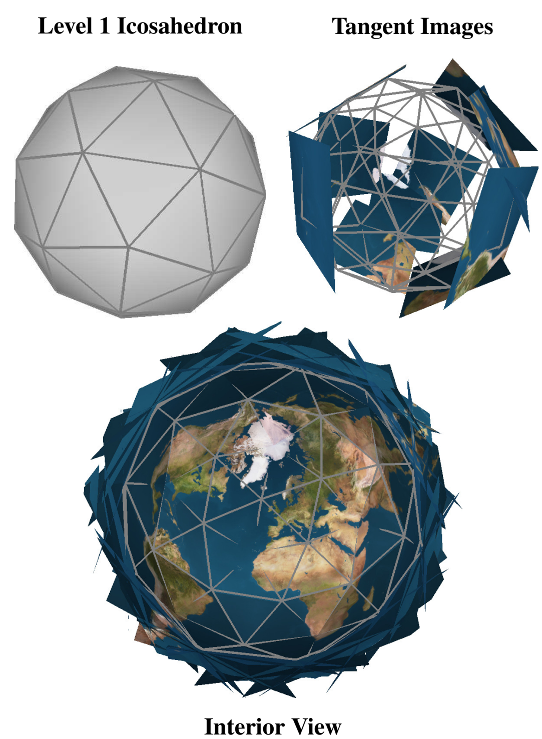


Figure 1: Using tangent images to represent a 4k Earth image [13]. **TL:** A base level 1 icosahedron. **TR:** Selection of tangent images rendered from the Earth image. **B:** Interior view of the tangent image spherical approximation.

Despite a growing body of work on these icosahedral convolutions, there are two significant impediments to further development: (1) the transferability of standard CNNs to spherical data on the icosahedron, and (2) the difficulty in scaling the proposed spherical convolution operations to high resolution spherical images. Prior work has implied [7, 11] or demonstrated [9, 27, 29] the transferability of networks trained on perspective images to different spherical representations. However, those who report results see a noticeable decrease in accuracy compared to CNN performance on perspective images and specialized networks that are trained natively on spherical data, leaving this important and desired behavior an unresolved question. Additionally, the proposed specialized convolutional kernels either require subsequent network tuning [7, 29] or are incompatible with the standard convolution [16].

Nearly all prior work on icosahedral convolutions has been built on the analogy between pixels and faces [7, 18] or pixels and vertices [11, 16, 29]. While elegant on the surface, this parallel has led to difficulties in scaling to higher resolution spherical images. Figure 2 depicts spherical image resolutions evaluated in the prior work. Notice that the highest resolution obtained so far is a level 8 subdivision, which is comparable to a 512×1024 equirectangular image. Superficially, this pixel resolution seems reasonably high, but the angular resolution per pixel is still quite low. A 512×1024 equirectangular image has an angular resolution of $0.352°$. For comparison, a VGA resolution (480×640) perspective image with $45° \times 60°$ field of view (FOV) has an angular resolution of $0.094°$. This is most similar to a 2048×4096 equirectangular image, which has an angular resolution of $0.088°$ and corresponds to a level 10 subdivided icosahedron. As this is a significantly higher resolution than prior work has been capable of demonstrating, this is the resolution on which we test our proposed approach.

In this work, we aim to address both transferability and scalability while leveraging efficient implementations of existing network architectures and operations. To this end, we propose a solution that decouples resolution from subdivision level using oriented, distortion-mitigated images that can be filtered with the standard grid convolution operation. Using these *tangent images*, standard CNN performance is competitive with specialized networks, yet they efficiently scale to high resolution spherical data and open the door to performance-preserving network transfer between perspective and spherical data. Furthermore, use of the standard convolution operation allows us to leverage highly-optimized convolution implementations, such as those from the cuDNN library [5], to train our networks. Additionally, the benefits of tangent images are not restricted to deep learning, as they address distortion through the data representation rather than the data processing tools. This means that our approach can be used for traditional vision applications like structure-from-motion and SLAM as well.

We summarize our contributions as follows:

- We propose the tangent image spherical representation: a set of oriented, low-distortion images rendered tangent to faces of the icosahedron.
- We show that standard CNNs trained on tangent images perform competitively with specialized spherical convolutional kernels while also scaling effectively to high resolution spherical images.
- We demonstrate that tangent images facilitate network transfer between perspective and spherical images with no fine tuning and minimal performance drop-off.
- We illustrate the utility of tangent images for traditional computer vision tasks by using them to improve sparse keypoint matching on spherical images.

2. Related Work

Recently, there have been a number of efforts to close the gap between CNN performance on perspective images and spherical images. These efforts can be naturally divided based on the spherical image representation used.

2.1. Equirectangular images

Equirectangular images are a popular spherical image representation thanks to their simple relation between rectangular and spherical coordinates. However, they demonstrate severe image distortion as a result. A number of methods have been proposed to address this issue. Su and Grauman [25] develop a learnable, adaptive kernel to train a CNN to transfer models trained on perspective images to the equirectangular domain. Su *et al*. [26] extend this idea by developing a kernel that learns to transform a feature map according to local distortion properties. Cohen *et al*. [8, 6] develop spherical convolutions, which provides the rotational equivariance necessary for convolutions on the sphere. This method requires a specialized kernel, however, making it difficult to transfer the insights developed from years of research into traditional CNNs. Works from Coors *et al*. [9] and Tateno *et al*. [27] address equirectangular image distortion by warping the planar convolution kernel in a location-dependent manner. Because the equirectangular representation is so highly distorted, most recent work on this topic, has looked to leverage the distorted-reducing properties of the icosahedral spherical approximation.

2.2. Icosahedral representations

Representing the spherical image as a subdivided icosahedron mitigates spherical distortion, thus improving CNN accuracy compared to techniques that operate on equirectangular images. Eder and Frahm [10] motivate this representation using analysis from the field of cartography. Further research on this representation has primarily focused on the development of novel kernel designs to handle discretization and orientation challenges on the icosahedral manifold. Lee *et al*. [18] convolve on this representation by defining new, orientation-dependent, kernels to sample from triangular faces of the icosahedron. Jiang *et al*. [16] reparameterize the convolutional kernel as a linear combination of differential operators on the surface of an icosahedral mesh. Zhang *et al*. [29] present a method that applies

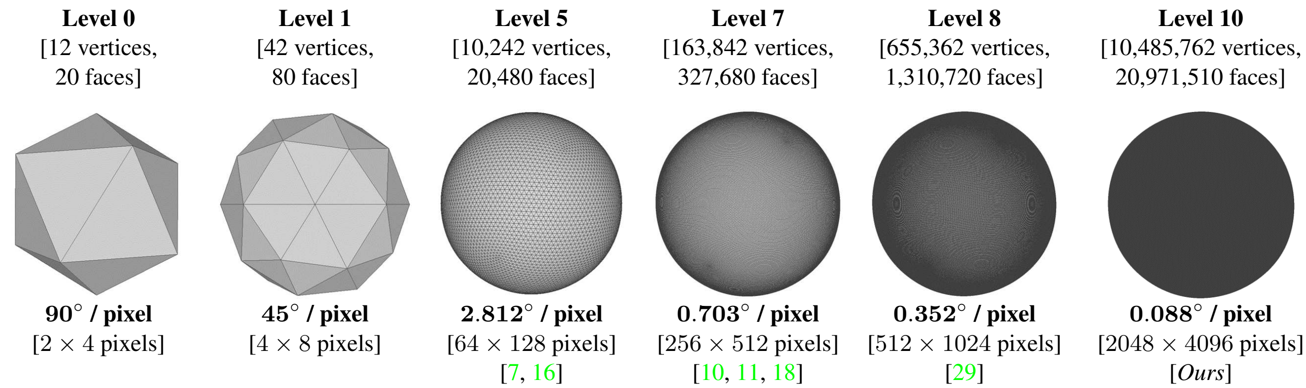


Figure 2: Demonstrating the number of elements, corresponding equirectangular image dimensions, and angular pixel resolution at various icosahedral subdivision levels. The citations beneath each denote the maximum resolution examined in those respective papers. Except for ours, they have all been limited by the pixel-to-face or pixel-to-vertex analogy.

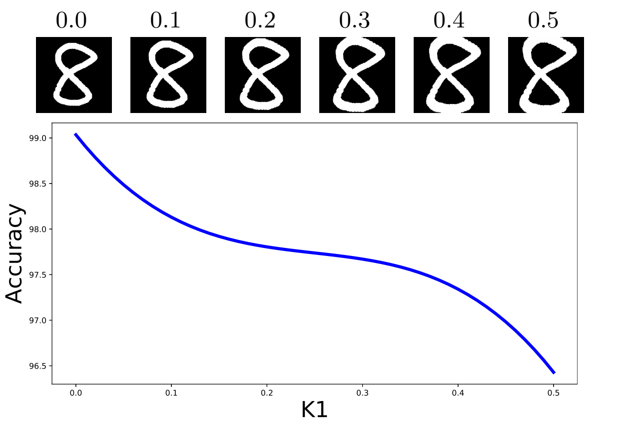


Figure 3: MNIST classification accuracy decreases as pincushion distortion is added to test images by varying the $K1$ parameter of the Brown-Conrady radial distortion model [2]. An example digit is shown at different distortion levels.

a special hexagonal convolution on the icosahedral net. Cohen *et al*. [7] precompute an atlas of charts at different orientations that cover the icosahedral grid and use masked kernels along with an feature-orienting transform to convolve on these planar representations. Eder *et al*. [11] define the "mapped convolution" that allows the custom specification of convolution sampling patterns through a type of graph convolution. In this way, they specify the filters' orientation and sample from the icosahedral surface. Our tangent image representation addresses data orientation by ensuring all tangent images are consistently oriented when rendering and circumvents the discretization issue by rendering to image pixel grids.

3. Mitigating Spherical Distortion

Image distortion is the reason that we cannot simply apply many state-of-the-art CNNs to spherical data. Distortion changes the representation of the image, resulting in local content deformation that violates translational equivariance, the key property of a signal required for convolution functionality. The graph in Figure 3 shows just how little distortion is required to produce a significant drop-off in CNN performance. Distortion in the most popular spherical image representations, equirectangular images and cube maps, is quite significant [10], and hence results in even worse performance. Although we can typically remove most lens distortion in perspective images using tools like the Brown-Conrady distortion model [2], spherical distortion is inescapable. This follows from Gauss's Theorema Egregium, a consequence of which is that a spherical surface is not isometric to a plane. As such, any effort to represent a spherical image as a planar one will result in some degree of distortion. Thus, our objective, and one shared by cartographers for thousands of years, is limited to finding an optimal planar representation of the sphere for our use case.

3.1. The icosahedral sphere

Consider the classical *method of exhaustion* of approximating a circle with inscribed regular polygons. It follows that, in three dimensions, we can approximate a sphere in the same way. Thus, the choice of planar spherical approximation ought to be the convex Platonic solid with the most faces: the icosahedron. The icosahedron has been used by cartographers to represent Earth at least as early as Buckminster Fuller's Dymaxion map [3], which projects the globe onto the icosahedral net. Recent work in computer vision [7, 10, 11, 18, 16, 29] has demonstrated the shape's utility for resolving the distortion problem for CNNs on spherical images as well.

While an improvement over single-plane image projections and its Platonic solid cousin, the cube, the 20-face icosahedron on its own is still limited in its distortion-mitigating properties. It can be improved by repeatedly applying Loop subdivision [21] to subdivide the faces and interpolate the vertices, producing increasingly close spherical approximations with decreasing amounts of local distortion on each face. Figure 4 demonstrates how distortion

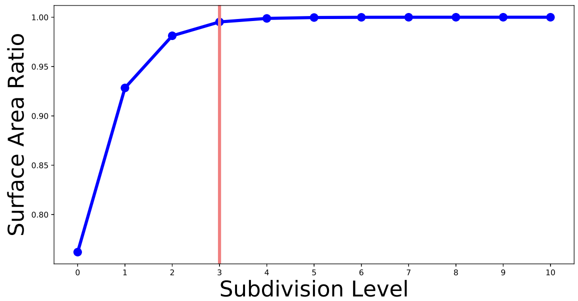


Figure 4: Ratio of the surface area of the subdivided icosahedron to the surface area of a sphere of the same radius at each subdivision level. This global metric demonstrates how closely the subdivision surface approximates a sphere and is drawn from established cartographic metrics [17]. Note the leveling off after the third subdivision level.

decreases at each subdivision level. Not all prior work takes advantage of this extra distortion reduction, though. There has largely been a trade-off between efficiency and representation. The charts used by Cohen *et al*. [7] and the net used by Zhang *et al*. [29] are efficient thanks to their planar image representations, but they are limited to the distortion properties of a level 0 icosahedron. On the other hand, the mapped convolution proposed by Eder *et al*. [11] operates on the mesh itself and thus can benefit from higher level subdivision, but it does not scale well to higher level meshes due to cache coherence problems when computing intermediate features on the mesh. Jiang *et al*. [16] provide efficient performance on the mesh, but do so by approximating convolution with a differential operator, which means existing networks can not be transferred. It is also interesting to note that the current top-performing method for many deep learning tasks, [29], uses the net of the level 0 icosahedron. This suggests that extensive subdivisions may not be necessary for all use cases.

Practical methods for processing spherical images must address the efficient scalability problem, but also should permit the transfer of well-researched, high-performance methods designed for perspective images. They should also provide the opportunity to modulate the level of acceptable distortion depending on the application. To address these constraints, we propose to break the coupling of subdivision level and spherical image resolution by representing a spherical image as a collection of images with tunable resolution and distortion characteristics.

3.2. Tangent images

Subdividing the icosahedron provides diminishing returns rather quickly from a distortion-reduction perspective, as indicated by the red vertical line in Figure 4. Nonetheless, existing methods must continue to subdivide in order to match the spherical image resolution to the number of mesh elements. We untether these considerations by fixing a *base level* of subdivision, b, to define an acceptable de-

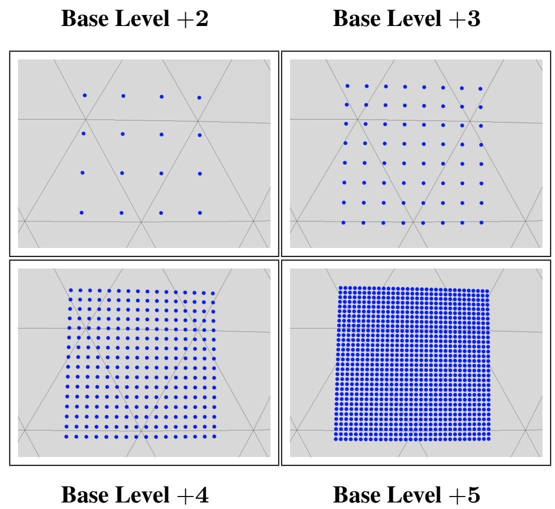


Figure 5: Illustrating how the tangent image resolution increases without changing the underlying subdivision level. The field-of-view of the tangent pixel grid remains unchanged, but its resolution increases by a factor of 2 in each dimension, demonstrated by the blue dots representing pixel samples on the sphere. This scaling maintains the angular pixel resolution of higher level icosahedrons without the need for additional subdivisions.

gree of distortion, and then rendering the spherical image to square, oriented, planar pixel grids tangent to each face at that base level. The resolution of these *tangent images* is subsequently determined by the resolution of the spherical input. Given a subdivision level, s, corresponding to the spherical input resolution, the dimension of the tangent image, d, is given by the relation:

$$d = 2^{s-b} \tag{1}$$

This design preserves the same resolution scaling that would occur through further subdivisions by instead increasing the resolution of the tangent image. This relationship is illustrated in Figure 5.

Our tangent images are motivated by existing techniques in related fields. The approximation of sections of the sphere by low-distortion planar regions is similar to the Universal Transverse Mercator (UTM) geodetic coordinate system, which divides the Earth into a number of nearly-Euclidean zones. Additionally, as tangent images can be thought of as rendering a spherical mesh to a set of quad textures, the high resolution benefits are similar to Ptex [4], a computer graphics technique that enables efficient high-resolution texturing by providing every quad of a 3D mesh with its own texture map. A visualization of the tangent image concept is provided in Figure 1.

Computing tangent images Tangent images are the gnomonic projection of the spherical data onto oriented, square planes centered at each face of a level b subdivided icosahedron. The number of tangent images, N, is determined by the faces of the base level icosahedron: $N = 20(4^b)$, while their spatial extent is a function of the vertex resolution, $R_v(b-1)$, of the level $b-1$ icosahedron and the resolution of the image grid, given by Equation (1). Let (ϕ_f, λ_f) be the barycenter of a triangular face of the icosahedron in spherical coordinates. We then compute the bounds of the plane in spherical coordinates as the inverse gnomonic projection at central latitude and longitude (ϕ_f, λ_f) of the points:

$$\left\{\phi_f \pm \frac{d-1}{2d} R_v(b-1)\right\} \times \left\{\lambda_f \pm \frac{d-1}{2d} R_v(b-1)\right\} \tag{2}$$

The vertex resolution, R_v, of a level b icosahedron, $\mathcal{S}(b)$, is computed as the mean angle between all vertices, v, and their neighbors, $\mathrm{adj}(v)$:

$$R_v(b) = \frac{1}{|\mathcal{S}(b)|} \sum_{v \in \mathcal{S}(b)} \sum_{w \in \mathrm{adj}(v)} \frac{\angle(v,w)}{|\mathrm{adj}(v)|} \tag{3}$$

Using $R_v(b-1)$ ensures that the tangent images completely cover their associated triangular faces. Because vertex resolution roughly halves at each subsequent subdivision level, we define $R_v(-1) = 2R_v(0)$.

Using tangent images Tangent images require rendering from and to the sphere only once each. First, we create the tangent image set by rendering to the planes defined by Equation (2). Then, we apply the desired perspective image algorithm (e.g. a CNN or keypoint detector). Finally, we compute the regions on each plane visible to a spherical camera at the center of the icosahedron and render the algorithm output back to the sphere.

We have released our tangent image rendering code and associated experiments as a PyTorch extension[1].

4. Experiments

Prior research has established a common suite of experiments that have become the test bed for new research on spherical convolutions. This set typically includes some combination of spherical MNIST classification [8, 7, 16, 18, 29], shape classification [8, 12, 16], climate pattern segmentation [7, 16, 29], and semantic segmentation [7, 16, 18, 27, 29]. In order to benchmark against these prior works, we evaluate our method on the shape classification and semantic segmentation tasks. Additionally, we demonstrate our method's fairly seamless transfer of CNNs trained on perspective images to spherical data. Finally, to show the versatility of the tangent image representation, we introduce a new benchmark, sparse keypoint detection on spherical images, and compare our representation to an equirectangular image baseline.

4.1. Classification

We first evaluate our proposed method on the shape classification task. As with prior work, we use the ModelNet40 dataset [28] rendered using the method described by Cohen *et al*. [8]. Because the data densely encompasses the entire sphere, unlike spherical MNIST, which is sparse and projected only on one hemisphere, we believe this task is more indicative of general classification performance.

[1] https://github.com/meder411/Tangent-Images

Method	Filter	Acc.
Cohen *et al*. [8]	Spherical Correlation	85.0%
Esteves *et al*. [12]	Spectral Parameterization	88.9%
Jiang *et al*. [16]	MeshConv	**90.5%**
Ours	2D Convolution	89.1%

Table 1: Classification results on the ModelNet40 dataset [28]. Without any specialized convolution operations, our approach is competitive with the state of the art spherical convolution methods.

Experimental setup We use the network architecture from Jiang *et al*. [16], but we replace the specialized kernels with simple 3×3 2D convolutions. A forward pass involves running the convolutional blocks on each patch separately and subsequently aggregating the patch features with average pooling. We train and test on level 5 resolution data as with the prior work.

Results and analysis Results of our experiments are shown in Table 1. Without any specialized convolutional kernels, we outperform most of the prior work on this task. The best performing method from Jiang *et al*. [16] leverages a specialized convolution approximation on the mesh, which inhibits the ability to fine-tune existing CNN models for the task. Our method can be thought of as using a traditional CNN in a multi-view approach to spherical images. This means that, for global inference tasks like classification, we could select our favorite pre-trained network and transfer it to spherical data. In this case, it is likely that some fine-tuning may be necessary to address the final patch aggregation step in our network design.

4.2. Semantic segmentation

We next consider the task of semantic segmentation in order to demonstrate dense prediction capabilities. To compare to prior work, we perform a baseline evaluation of our method at low icosahedron resolutions (5 and 7), but we also evaluate the performance of our method at a level 10 input resolution in order to demonstrate the usefulness of the tangent image representation for processing high resolution spherical data. No prior work has operated at this resolution. We hope that our work can serve as a benchmark for further research on high resolution spherical images.

Experimental setup We train and test our method on the Stanford 2D3DS dataset [1], as with prior work [8, 7, 16, 29]. We evaluate RGB-D inputs at levels 5, 7, and 10, the maximum resolution provided by the dataset. At level 10 we also evaluate using only RGB inputs to demonstrate the benefit of high resolution capabilities. For the level 5 and 7 experiments, we use the residual UNet-style architecture as in [16, 29], but we again replace the specialized

Stanford2D3DS Dataset					
s	Method	Input	b	mAcc	mIOU
5	Cohen *et al.* [7]	RGB-D	0	55.9	39.4
	Jiang *et al.* [16]	RGB-D	5	54.7	38.3
	Zhang *et al.* [29]	RGB-D	0	**58.6**	**43.3**
	Ours	RGB-D	0	50.2	37.5
7	Tateno *et al.* [27]	RGB	ERP	-	34.6
	Lee *et al.* [18]	RGB	7	26.4	-
	Ours	RGB-D	0	**54.9**	**41.8**
10	Ours	RGB	0	61.0	44.3
	Ours	RGB	1	**65.2**	**45.6**
	Ours	RGB	2	61.5	42.7
	Ours	RGB-D	1	**69.1**	**51.9**

Table 2: Semantic segmentation results. s is the input resolution in terms of equivalent icosahedron level, b is the base subdivison level (ERP denotes equirectangular inputs), mIoU is the mean intersection-over-union metric, and mAcc is the weighted per-class mean prediction accuracy.

kernels with 3×3 convolutions. The higher resolution of the level 10 inputs requires the larger receptive field of a deeper network, so we use a FCN-ResNet 101 [14, 20] model pretrained on COCO [19] for those experiments. For level 5 data, we train on the entire set of tangent images, while for the higher resolution experiments, we randomly sample a subset of tangent images from each spherical input to expedite training. We found this sampling method to be useful without loss of accuracy. We liken it to training on multiple perspective views of a scene.

Results and analysis We report the results of our experiments in Table 2. Results on the Stanford2D3DS dataset are averaged over the 3 folds. Individual class results can be found in the supplementary material. As expected, our method does not perform as well as prior work at the level 5 resolution. Recall that a level 5 resolution spherical image is equivalent to a 16×16 perspective image with $45°$ FOV. Our method takes that already low angular resolution image and separates it into a set of low pixel resolution images. Although it had limited impact on classification, these dual low resolutions are problematic for dense prediction tasks. We expound on the low-resolution limitation further in the supplementary material.

Where our tangent image representation excels is when scaling to high resolution images. What we sacrifice in low-resolution performance, we make up for by efficiently scaling to high resolution inputs. By scaling to the full resolution of the dataset, we are able to report the highest performing results ever on this spherical dataset by a wide margin using only RGB inputs. Adding the extra depth channel, we are able to increase the performance further ($+4.8$ mAcc, $+7.0$ mIOU). At input level 10, we find that base level 1 delivers the best trade-off between the lower FOV at higher base levels and the increased distortion present in lower ones. We elaborate on this trade-off in the supplementary material.

4.3. Network transfer

Our contribution aims to address equivalent network performance regardless of the input data format. That is, for a given network, we strive to achieve *equal performance on both perspective and spherical data*. This objective is motivated by the limited number of spherical image datasets and the difficulty of collecting large scale spherical training data. If we can achieve high transferability of perspective image networks, we reduce the need for large amounts of spherical training data. Because generating tangent images inherently converts a spherical image into a collection of perspective ones, this representation facilitates the desired network transferability without requiring fine-tuning on the spherical data and with limited performance drop-off.

Experimental setup We evaluate the transferability of the tangent image representation in three experiments.

In the first experiment, we evaluate semantic segmentation performance on a *spherical* image test set using a network trained on the corresponding *perspective image* training set. We fine-tune the pre-trained, FCN-ResNet101 model [14, 20] provided by the PyTorch model zoo on the Stanford2D3DS dataset's [1] perspective image training set. We then evaluate semantic segmentation performance on the spherical image test set at a level 8 resolution. This experiment uses RGB inputs only. During the dataset fine-tuning, we make sure to consider the desired angular resolution of the spherical test images. A network trained on perspective images with an angular resolution of $1°$ has learned filters accordingly. Should we apply those filters to an image captured at the identical position, at the same image resolution, but with a narrower FOV, the difference in angular resolution is effectively scale distortion. To match the angular resolution of our spherical evaluation set, we normalize the camera matrices for all perspective images during training such they have the same angular resolution as the test images. Because this is effectively a center-crop of the data, we also randomly shift our new camera center in order capture all parts of the image. Details of this pre-processing are given in the supplementary material. Note that we do not fine-tune on the spherical data.

The second experiment compares the transferability provided by tangent images to prior work that addresses this topic [29]. Using the network architecture from Zhang *et al.* [29], we train a model on the perspective images from the SYNTHIA dataset [24] that correspond to the OmniSYNTHIA dataset's [29] training set. We again utilize the camera normalization procedure mentioned above. We evaluate performance on the OmniSYNTHIA test set at base level 1.

Finally, the third experiment studies the impact of matching angular resolution between training and testing. For this, we apply the ResNet 101 semantic segmentation model from the first experiment to the spherical test set at various resolutions.

Format	Input Res.	AngRes/Pix	mAcc	mIOU
Persp.	128×128	0.352°	55.7	38.9
Spher.	Level 8	0.352°	51.6	36.2

Table 3: Transfer learning using RGB-D data from the Stanford2D3DS dataset. Without fine-tuning, we preserve 93% of the perspective network accuracy when transferring to spherical data represented by tangent images.

s	Method	mAcc	mIOU
6	Zhang *et al.* [29] (*transfer*)	44.8	36.7
	Ours (*transfer*)	**52.8**	**41.3**
	Zhang *et al.* [29] (*native*)	*52.2*	*43.6*
7	Zhang *et al.* [29] (*transfer*)	47.2	**38.0**
	Ours (*transfer*)	**55.3**	35.8
	Zhang *et al.* [29] (*native*)	*57.1*	*48.3*
8	Zhang *et al.* [29] (*transfer*)	52.8	45.3
	Ours (*transfer*)	**65.4**	**49.7**
	Zhang *et al.* [29] (*native*)	*55.1*	*47.1*

Table 4: Comparing our transfer learning results to the prior work from Zhang *et al.* [29] on the OmniSYNTHIA dataset at different input resolutions, s. Note that their reported results are after 10 epochs of fine-tuning, while ours uses none.

Results and analysis Results for the first two experiments are given in Tables 3 and 4, respectively.

In the first experiment, note that both results are attained using a network trained only on perspective data. With tangent images, we are able to preserve 92.6% of the accuracy and 93.1% of the IOU of the perspective evaluation without any subsequent network tuning. This is because the tangent image representation has similar distortion characteristics to perspective images, and we have matched the angular resolution between the two domains.

The results of the second experiment demonstrate that the tangent image approach significantly outperforms the prior state-of-the-art without any specialized kernels or subsequent fine-tuning. Note that Zhang *et al.* [29] report results after 10 epochs of fine-tuning on spherical images, while our approach does not fine-tune on spherical images at all. It is also worth observing that, at higher resolutions, our transfer results are actually competitive with the existing method trained natively on spherical data. Our experiments have been limited by the maximum resolution of available spherical image datasets, but this outcome suggests that network transfer with tangent images may permit even higher resolution spherical image inference.

Finally, the results of the third experiment are plotted in Figure 6. Recall that this model was trained on perspective images normalized to have a per-pixel angular resolution equivalent to that of a level 8 icosahedron. This chart highlights the importance of camera normalization when training on perspective images with the purpose of transferring the network. Observe how performance deteriorates as the angular resolution of the spherical input moves further from the angular resolution of the training data.

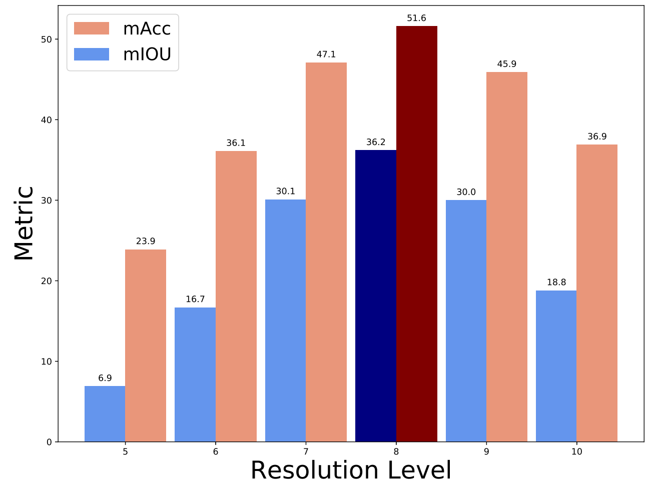
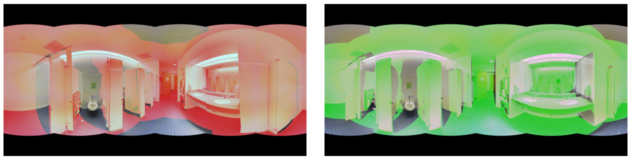


Figure 6: Results are shown for spherical semantic segmentation using a network trained on perspective images that are normalized to have a angular resolution equivalent to a level 8 spherical input. Performance drops off considerably as the angular resolution of the spherical inputs becomes more dissimilar to the training data. Level 8 results are darkened.

4.4. Sparse keypoint correspondences

Recent research on spherical images has focused on deep learning tasks, primarily because many of those works have focused on the convolution operation. As our contribution relates to the representation of spherical data, not specifically convolution, we aim to show that our approach has applications beyond deep learning. To this end, we evaluate the use of tangent images for sparse keypoint detection, a critical step of structure-from-motion, SLAM, and a variety of other traditional computer vision applications.

Data As there is no existing benchmark for this task, we create a dataset using a subset of the spherical images provided by the Stanford2D3DS dataset [1]. To create this dataset, we first cluster the dataset's Area 1 images according to the provided room information. Then, for each location, we compute SIFT features [22] in the equirectangular images and identify which image pairs have FOV overlap using the spherical structure-from-motion pipeline provided by the OpenMVG library [23]. Next, we compute the average volumetric FOV overlap for each overlapping image pair. Because we are dealing with 360° images, there are no image bounds to constrain "visible" regions. Instead, we use the ground truth depth maps and pose information to back-project each image pair into a canonical pose. We then compute the percentage of right image points visible to the left camera using the the left image depth map to remove occluded points, and vice versa. We average the two values to provide an FOV overlap score for the image pair. This overlap is visualized in Figure 7. We define our keypoints dataset as the top 60 image pairs according to this overlap metric. Finally, we split the resulting dataset into an "Easy"



Left Image **Right Image**

Figure 7: FOV overlap visualized between an image pair from our keypoints benchmark derived from the Stanford 2D3DS dataset [1]. The red regions in the left image represent areas visible to the right camera, and the green regions in the right image represent areas visible to the left camera.

set and "Hard" set, again based on FOV overlap. The resulting dataset statistics are shown in Table 5. All images are evaluated at their full, level 10 resolution. We provide the dataset details in the supplementary material to enable further research.

Experimental setup To evaluate our proposed representation, we detect and describe keypoints on the tangent image grids and then render those keypoints back to the spherical image. This rendering step ensures only keypoints visible to a spherical camera at the center of the icosahedron are rendered, as the tangent images have overlapping content. We then use OpenMVG [23] to compute putative correspondences and geometrically-consistent inlier matches.

Results and analysis We evaluate the quality of correspondence matching at 3 different base levels using the equirectangular image format as a baseline. We compute the *putative matching ratio* (PMR), *matching score* (MS), and *precision* (P) metrics defined by Heinly *et al.* [15]. For an image set $\mathcal{S}$ of image pairs, (L, R), with p putative correspondences, f inlier matches, and $n_{\{L,R\}}$ detected keypoints visible to both images, the metrics over the image pairs as defined as follows:

$$\begin{aligned} \text{PMR} &= \frac{1}{2|\mathcal{S}|} \sum_{(L,R)\in\mathcal{S}} \left(\frac{p}{n_L} + \frac{p}{n_R} \right) \\ \text{MS} &= \frac{1}{2|\mathcal{S}|} \sum_{(L,R)\in\mathcal{S}} \left(\frac{f}{n_L} + \frac{f}{n_R} \right) \\ \text{P} &= \frac{1}{|\mathcal{S}|} \sum_{(L,R)\in\mathcal{S}} \frac{f}{p} \end{aligned} \tag{4}$$

In the same way that we compute the FOV overlap, we use the ground truth pose and depth information provided by the dataset to determine which keypoints in the left image should be visible to the right image (n_L) and vice versa (n_R), accounting for occlusion.

Results are given in Table 6. Our use of tangent images has a strong impact on the resulting correspondences, particularly on the hard split. Recall that this split has a lower FOV overlap and fewer inlier matches at the baseline equirectangular representation. Improved performance in this case is thus especially useful. We observe a significant improvement in PMR in both splits. We attribute this improvement to the computation of the SIFT feature vector on our less distorted representation. Like the convolution operation, SIFT descriptors also require translational equivariance in the detection domain. Tangent images restore this property with their low-distortion representation, which enables repeatable descriptors. The better localization of the keypoints affects the inlier matches as well, resulting in a better MS score. We attribute the leveling off in performance beyond level 1 to the reduced FOV of higher level subdivisions, which impedes the detector's ability to find keypoints at larger scales.

Split	# Pairs	Mean FOV Overlap	# Corr.
Hard	30	83.35%	298
Easy	30	89.35%	515

Table 5: Statistics of our keypoints benchmark. # Corr. is the number of inlier matches detected on the equirectangular images in that split. Statistics are averaged over the splits.

Hard				
Metric	**Equirect.**	**L0**	**L1**	**L2**
PMR	22.2%	28.4%	**30.1%**	27.4%
MS	8.2%	11.1%	**11.7%**	10.9%
P	36.9%	39.5%	39.6%	**40.2%**
Easy				
Metric	**Equirect.**	**L0**	**L1**	**L2**
PMR	26.3%	32.4%	**34.6%**	31.9%
MS	13.6%	16.6%	**17.7%**	16.1%
P	46.0%	46.4%	**47.5%**	46.5%

Table 6: Keypoint evaluation metrics. We report the each metric's average over all image pairs per split. L{0,1,2} are the subdivision levels at which we compute the keypoints.

5. Conclusion

We have presented tangent images, a spherical image representation that renders the image onto a oriented pixel grids tangent to a subdivided icosahedron. We have shown that these tangent images do not require specialized convolutional kernels for training CNNs and efficiently scale to represent high resolution data. We have also shown that they facilitate the transfer of networks trained on perspective images to spherical data with limited performance loss. These results further suggest that network transfer using tangent images can open the door to processing even higher resolution spherical images. Lastly, we have demonstrated the utility of tangent images for traditional computer vision tasks in addition to deep learning. Our results indicate that tangent images can be a very useful spherical representation for a wide variety of computer vision applications.

Acknowledgements We would like to thank David Luebke, Pierre Moulon, Li Guan, and Jared Heinly for their consultation in support of this work. This research was funded in part by Zillow.

References

[1] Iro Armeni, Sasha Sax, Amir R Zamir, and Silvio Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017. 5, 6, 7, 8

[2] Duane C Brown. Decentering distortion of lenses. *Photogrammetric Engineering and Remote Sensing*, 1966. 3

[3] Fuller Richard Buckminster. Cartography, Jan. 29 1946. US Patent 2,393,676. 3

[4] Brent Burley and Dylan Lacewell. Ptex: Per-face texture mapping for production rendering. In *Computer Graphics Forum*, volume 27, pages 1155–1164. Wiley Online Library, 2008. 4

[5] Sharan Chetlur, Cliff Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cudnn: Efficient primitives for deep learning. *arXiv preprint arXiv:1410.0759*, 2014. 2

[6] Taco Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Convolutional networks for spherical signals. *arXiv preprint arXiv:1709.04893*, 2017. 2

[7] Taco Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional networks and the icosahedral cnn. In *International Conference on Machine Learning*, pages 1321–1330, 2019. 1, 2, 3, 4, 5, 6

[8] Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations*, 2018. 1, 2, 5

[9] Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In *European Conference on Computer Vision*, pages 525–541. Springer, 2018. 1, 2

[10] Marc Eder and Jan-Michael Frahm. Convolutions on spherical images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–5, 2019. 1, 2, 3

[11] Marc Eder, True Price, Thanh Vu, Akash Bapat, and Jan-Michael Frahm. Mapped convolutions. *arXiv preprint arXiv:1906.11096*, 2019. 1, 2, 3, 4

[12] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68, 2018. 5

[13] James Hastings-Trew. Planet texture maps, 2019. `http://planetpixelemporium.com/earth8081.html`. Accessed 2019-04-16. 1

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 6

[15] Jared Heinly, Enrique Dunn, and Jan-Michael Frahm. Comparative evaluation of binary features. In *European Conference on Computer Vision*, pages 759–773. Springer, 2012. 8

[16] Chiyu Max Jiang, Jingwei Huang, Karthik Kashinath, Prabhat, Philip Marcus, and Matthias Niessner. Spherical CNNs on unstructured grids. In *International Conference on Learning Representations*, 2019. 1, 2, 3, 4, 5, 6

[17] Jon A Kimerling, Kevin Sahr, Denis White, and Lian Song. Comparing geometrical properties of global grids. *Cartography and Geographic Information Science*, 26(4):271–288, 1999. 4

[18] Yeonkun Lee, Jaeseok Jeong, Jongseob Yun, Wonjune Cho, and Kuk-Jin Yoon. Spherephd: Applying cnns on a spherical polyhedron representation of 360deg images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9181–9189, 2019. 1, 2, 3, 5, 6

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 6

[20] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 6

[21] Charles Loop. Smooth subdivision surfaces based on triangles. *Master's thesis, University of Utah, Department of Mathematics*, 1987. 3

[22] David G Lowe et al. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, volume 99, pages 1150–1157, 1999. 7

[23] Pierre Moulon, Pascal Monasse, Romuald Perrot, and Renaud Marlet. Openmvg: Open multiple view geometry. In *International Workshop on Reproducible Research in Pattern Recognition*, pages 60–74. Springer, 2016. 7, 8

[24] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 6

[25] Yu-Chuan Su and Kristen Grauman. Learning spherical convolution for fast features from 360 imagery. In *Advances in Neural Information Processing Systems*, pages 529–539, 2017. 1, 2

[26] Yu-Chuan Su and Kristen Grauman. Kernel transformer networks for compact spherical convolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2

[27] Keisuke Tateno, Nassir Navab, and Federico Tombari. Distortion-aware convolutional filters for dense prediction in panoramic images. In *European Conference on Computer Vision*, pages 732–750. Springer, 2018. 1, 2, 5, 6

[28] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015. 5

[29] Chao Zhang, Stephan Liwicki, William Smith, and Roberto Cipolla. Orientation-aware semantic segmentation on icosahedron spheres. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3533–3541, 2019. 1, 2, 3, 4, 5, 6, 7