This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DarSwin: Distortion Aware Radial Swin Transformer

Akshaya Athwale[1], Arman Afrasiyabi[3], Justin Lagüe[1], Ichrak Shili[1],
Ola Ahmad[1,2], Jean-François Lalonde[1]

[1]Université Laval [2]Thales Digital Solutions [3] Yale University

Abstract

Wide-angle lenses are commonly used in perception tasks requiring a large field of view. Unfortunately, these lenses produce significant distortions making conventional models that ignore the distortion effects unable to adapt to wide-angle images. In this paper, we present a novel transformer-based model that automatically adapts to the distortion produced by wide-angle lenses. We leverage the physical characteristics of such lenses, which are analytically defined by the radial distortion profile (assumed to be known), to develop a distortion aware radial swin transformer (DarSwin). In contrast to conventional transformer-based architectures, DarSwin comprises a radial patch partitioning, a distortion-based sampling technique for creating token embeddings, and an angular position encoding for radial patch merging. We validate our method on classification tasks using synthetically distorted ImageNet data and show through extensive experiments that DarSwin can perform zero-shot adaptation to unseen distortions of different wide-angle lenses. Compared to other baselines, DarSwin achieves the best results (in terms of Top-1 accuracy) with significant gains when trained on bounded levels of distortions (very-low, low, medium, and high) and tested on all including out-of-distribution distortions. The code and models are publicly available at https://lvsn.github.io/darswin/

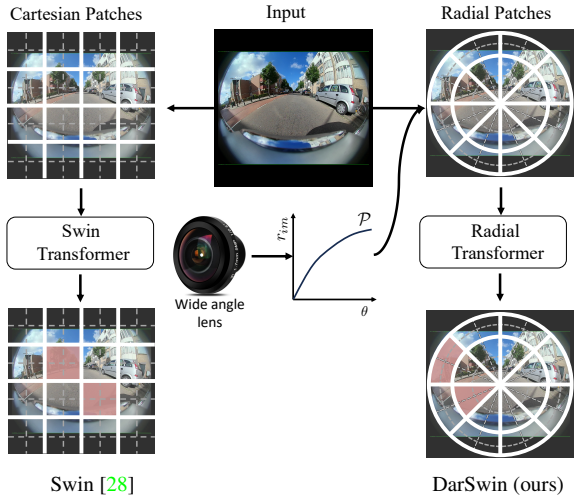


Figure 1: Illustration of (cartesian) Swin [28] (left) and our (radial) DarSwin (right) given a wide-angle image (middle). While Swin [28] computes attention on the predefined windows over square image patches (bottom left orange region), DarSwin performs radial transformations using distortion-aware radial patches and computes the attention on windows defined over radial patches (shown in orange region bottom right), which enables greater generalization capabilities across different lenses.

1. Introduction

Wide field of view (FOV) lenses are becoming increasingly popular because their increased FOV minimizes cost, energy, and computation since fewer cameras are needed to image the entire environment. They are having a positive impact on many applications, including security [22], augmented reality (AR) [38], healthcare and more particularly, autonomous vehicles [9, 49], which require sensing their surrounding 360° environment.

Unfortunately, such wide angle lenses create significant distortions in the image since the perspective projection model no longer applies: straight lines appear curved, and the appearance of the same object changes as a function of its position on the image plane. This distortion breaks the translation equivariance assumption implicit in convolution neural networks (CNNs) and therefore limits their applicability. This problem is further exacerbated by the diversity in lens distortion profiles: as we will demonstrate, a network naively trained for a specific lens tends to overfit to that specific distortion, and does not generalize well when tested on another lens. Just as methods are needed to address the "domain gap" [16] from dataset bias [42, 20], we believe we must also bridge the "distortion gap" to truly make wide angle imaging applicable.

One popular strategy to bridge the distortion gap is to cancel the effect of distortion on the image plane by warping the input image back to a perspective projection model accord-

ing to calibrated lens parameters. Conventional approaches can then be trained and tested on the resulting "un-distorted" images. A wide array of such methods, ranging from classical [5, 35, 52, 30, 19] to deep learning [47, 45], have been proposed. Unfortunately, warping a very wide angle image to a perspective projection tends to create severely stretched images and restricts the maximum field of view since, in the limit, a point at 90° azimuth projects at infinity. Reducing the maximum field of view defeats the purpose of using a wide angle lens in the first place. Other projections are also possible (e.g., cylindrical [33]) but these also tend to create unwanted distortions.

Recently, methods that break free from the "undistort first" strategy aim to reason directly about the images without undistorting them. For example, methods like [34, 1] use deformable convolutions [8, 54] to adapt convolution kernels to the lens distortion. However, the high computational cost of deformable CNNs constrains the kernel adaptation to a few layers inside the network. Other approaches like spherical CNNs [6] or gauge equivariant CNN [7] can adapt to different manifolds but their applicability for lens distortion has not been demonstrated. Finally, vision transformers [11] and their more recent variants [53, 28, 44] could also better bridge the "distortion gap" since they do not assume any prior structure other than permutation equivariance, but their cartesian partitioning of the image plane do not take lens geometry into account (see fig. 1).

In this paper, we present DarSwin, a transformer-based architecture that adapts its structure to the lens distortion profile, which is assumed to be known (i.e. the camera is calibrated). Our method, inspired by the recent Swin transformer architecture [28], leverages a distortion aware sampling scheme for creating token embeddings, employs polar patch partitioning and merging strategies, and relies on angular relative positional encoding. This explicitly embeds knowledge of the lens distortion in the architecture and makes it much more robust to the "distortion gap" created by training and testing on different lenses.

The main contribution of this paper is a novel transformer-based encoder that automatically adapts to the (known) lens distortion profile, which relies on a combination of the following novel distortion aware components: polar patch partitioning, distortion-based sampling scheme for creating token embeddings along with a jittering technique for better generalization, and angular relative position encoding for radial patch merging. We show, through extensive classification experiments, that DarSwin can perform zero-shot adaptation (without pretraining) across different lenses. Indeed, when our method is trained on a restricted set of distortions, we observe that it is much more robust to changes in distortions at test time than all of the compared alternatives, including baseline Swin [28] (applied on both distorted and undistorted images) and deformable attention transformer [44].

2. Related work

Panoramic distortion Panoramic images span the full 360° field of view and are most commonly projected onto a plane using an equirectangular projection creates severe distortions especially near the poles. Many works have explicitly designed approaches to deal with equirectangular distortion, including depth estimation [55], saliency detection [50], segmentation [51], layout estimation [12], and object detection [39] to name a few. However, as with [6], these are specifically designed for spherical distortion and do not generalize to wide-angle lenses.

Image undistortion Applying tasks like classification and object recognition on wide-angle images is relatively recent [36, 37, 33, 46, 49] due to the presence of distortions in the image. In computer vision, the application of wide-angle images ranges from visual perception [25] to autonomous vehicle cameras [49, 24, 26]. In this respect, the initial studies mainly focused on correcting the distortion of the image [45, 10, 47, 52, 21]. Recently, many convolutional-based and attention-based models have been proposed that try to directly reason on wide-angle images without relying on distortion correction models.

Convolution-based approaches CNNs [23, 40, 27] are particularly well-suited for perspective images due to their implicit bias and translational equivariance [4]. Methods like [41, 37, 36] tries to adapt CNNs on fisheye images for tasks such as object detection. However, the distortion caused by wide angle images breaks this symmetry, which reduces the generalization performance of CNNs. Deformable convolutions [8] (later extended in [54]) learn a deformation to be applied to convolution kernels which can provide greater flexibility at the cost of significant additional computation. Closer to our work, [1, 34, 9] use such deformable CNNs to understand the distortion in a fisheye image. In contrast, our DarSwin leverages attention-based mechanisms rather than convolution. Recently, Jang et al. [18] proposed a framework for distortion-aware domain adaptation, where a generator network is trained to transform an image to a different lens profile. In contrast, our method can adapt to a different lens without additional training.

Self-attention-based approaches Vision transformers (ViT) [11] use self-attention mechanisms [43] computed on image patches rather than performing convolutions. Unlike CNNs, a ViT does not have a fixed geometric structure in its architecture: any extra structure is given via positional encoding. More recently, the Swin transformer architecture [28] proposes a multi-scale strategy of window-based attention. Later, deformable attention transformer (DAT) [44], adapts

the concept of deformable CNNs [8, 54] to increase adaptability. Unlike existing transformer architectures, DarSwin explicitly embeds the distortion into its structure using polar sampling, patch partition, window-based self-attention, and angular positional encoding.

3. Image formation

We begin with a brief review of lens distortion models relevant to this work. The pixel coordinates $\mathbf{p}_{\text{im}}$ of a 3D point $\mathbf{p}_{\text{w}} = [x, y, z]^{\mathsf{T}}$ in world coordinates are given by

$$\mathbf{p}_{\text{im}} = [u,\ v]^{\mathsf{T}} = \mathcal{P}(\mathbf{p}_{\text{w}})\,, \tag{1}$$

where $\mathcal{P}$ is a 3D-to-2D projection operator, including conversion from homogeneous coordinates to 2D. Here, without loss of generality, the camera is assumed to be at the world origin (so its rotation and translation are ignored).

To represent wide-angle images, it is common practice to use a projection model that describes the relationship between the radial distance $r_{\text{im}} = \sqrt{u^2 + v^2}$ from the image center and the incident angle $\theta = \arctan(\sqrt{x^2+y^2}/z)$. This relationship takes the generic form

$$r_{\text{im}} = \mathcal{P}(\theta)\,. \tag{2}$$

Within the scope of our interest, we consider three types of projections.

Perspective projection. Under the perspective projection model, the projection operator $\mathcal{P}$ takes the form [19]

$$\mathcal{P}_{\text{pers}}(\theta) \equiv r_{\text{im}} = f\tan(\theta)\,, \tag{3}$$

where f is the focal length (in pixels). The perspective projection is the rectilinear model of pinhole lenses. Wide-angle lenses disobey the law of perspective projection and therefore cause non-linear distortions.

Polynomial projection. In the case of wide-angle lenses, there are several classical projection models [3, 14, 13, 31] giving different formulas for $\mathcal{P}$; see [17] for a detailed analysis of the accuracy of such models. A unified, more general, model is defined as an $n-$degree polynomial and given by

$$\mathcal{P}_{\text{poly}}(\theta) \equiv r_{\text{im}} = a_1\theta + a_2\theta^2 + ... + a_n\theta^n\,. \tag{4}$$

For example, the WoodScape dataset [49] employs a 4-degree ($n = 4$) polynomial for their lens calibration. We adopt this polynomial function to define the lens projection curve used in our method (sec. 4).

Spherical projection. The spherical projection model [2, 29] describes the radial distortion by a *single*, *bounded* parameter $\xi \in [0, 1]$[1]. It projects the world point $\mathbf{p}_{\text{w}}$ to the image as follows

$$[u, v]^{\mathsf{T}} = \left[\frac{xf}{\xi||\mathbf{p}_{\text{w}}|| + z}, \frac{yf}{\xi||\mathbf{p}_{\text{w}}|| + z}\right]^{\mathsf{T}}\,. \tag{5}$$

We employ this model in experiments (sec. 5) because of its ability to represent distortion with a single parameter.

4. Methodology

4.1. Overview

Fig. 2 shows an overview of our proposed distortion aware transformer architecture. It accepts as input a single image and its distortion parameters in the form of a lens projection curve $\mathcal{P}(\theta)$ (see eq. (2)). The image domain is first segmented into patches according to a polar partitioning module (sec. 4.2). Then, a linear embedding is computed from sampled points (sec. 4.3), reshaped into a radial-azimuth projection and fed to the first Swin transformer block. The attention mechanism employs an angular relative positional encoding scheme guided by the lens curve. This is followed by three blocks performing patch merging (sec. 4.6) and additional Swin transformer blocks. More details on each of these steps are provided below.

4.2. Polar partition

The first step of our proposed architecture is to partition the image domain (defined by a 2D plane) into patches. As opposed to Swin which performs the split in cartesian coordinates (fig. 3a), DarSwin employs a polar patch partitioning strategy (fig. 3b). After centering a polar coordinate system on the image center (assumed to be known), we first split according to azimuth angle φ (in the image plane) in N_φ equiangular regions. For the radial dimension, we split the image into N_r radial regions such that the splits are equiangular in θ and obtain the corresponding radii using the lens projection function $\mathcal{P}(\theta)$ (see eq. (2) and fig. 4). The total number of patches is therefore $N_\varphi \times N_r$. In our experiments, we set $N_r = 16$ and $N_\varphi = 64$ for an input image of size 64×64 pixels.

4.3. Linear embedding

The resulting image patches created by this approach have unequal amount of pixels. Therefore, we rely on a distortion aware sampling strategy to obtain the same number of points for each patch. To sample from the images, we define the number of sampling points along the radius and azimuth, as shown in fig. 5, and adapt the pattern according

[1] ξ can be slightly greater than 1 for certain types of catadioptric cameras [48] but this is ignored here.

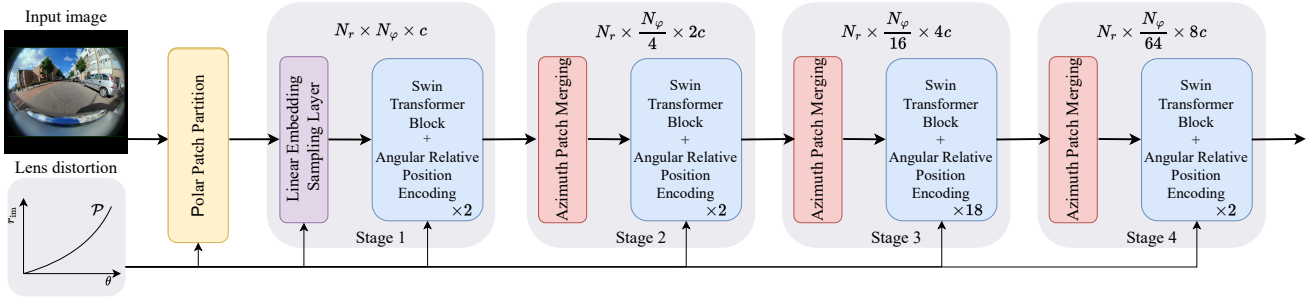
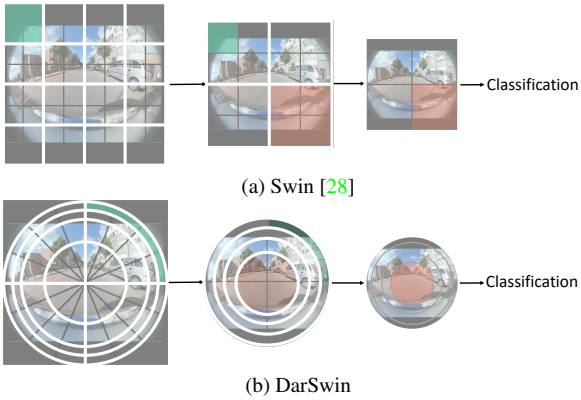


Figure 2: Overview of our distortion aware transformer architecture, DarSwin (Azimuth Merge). It employs hierarchical layers of Swin-S transformer blocks [28] interspersed with patch merging layers. To make it adapt to lens distortion, the patch partition, linear embedding, and patch merging layers all take the lens projection curve $\mathcal{P}$ (c.f. sec. 3) as input.



(a) Swin [28]

(b) DarSwin

Figure 3: Illustration of the difference between (a) Swin [28], which uses Cartesian image patches (shown in grey) and windows (shown in white borders) by making divisions along image axes, Swin performs attention on a Cartesian window and merges 2×2 neighbourhood patches to build a hierarchical structure (shown in the green and orange shaded region) (b) our proposed DarSwin uses polar image patches by making divisions along radius and azimuth. In our case, the images patches (shown in grey) and windows (shown in white borders) are defined along azimuth and merged along azimuth as (shown in the green and orange shaded region)

to each partition. In our experiments, we set 10 sampling points along radius and azimuth for each patch. Points are sampled in an equiangular fashion along the azimuth direction. For the radial dimension, we sample according to the same pattern as the polar partitioning (sec. 4.2); that is, we split in an equiangular fashion according to θ and obtain the corresponding radii using $\mathcal{P}(\theta)$ (eq. (2)) as shown in fig. 5. The input image is sampled using bilinear interpolation, and samples are arranged in a polar format as illustrated in fig. 5. The resulting sample values are then fed into a linear embedding layer to produce token embeddings as input for the

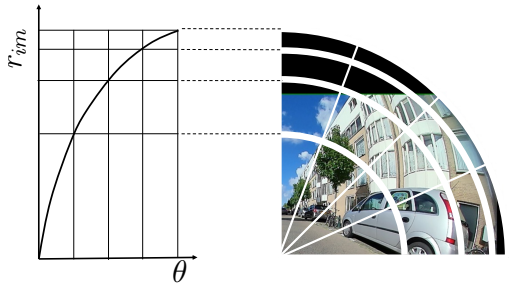


Figure 4: Example of distortion aware polar patch partition. Here, $N_r = 4$ and $N_\varphi = 4$ partitions along radius and azimuth respectively are used to illustrate (right, only the top-right quadrant of the image is shown). While the azimuth partitioning is performed in an equiangular fashion, the radial dimension takes the lens distortion curve (left) into account. The field of view along the incident angle θ is split into N_r equal parts (left) and corresponding radial are obtained from the distortion curve. Hence for different lenses we can have different radii depending on the distortion parameters.

transformer block.

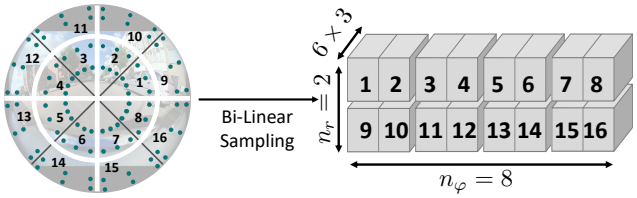


Figure 5: Example of sampling strategy on patch partitions with a total of 16 patches (here, the number of divisions along radius and azimuth are $N_r = 2$ and $N_\varphi = 8$ respectively) with window size $(1, 2)$, i.e. $(M_r = 1, M_\varphi = 2)$. We use bilinear interpolation to sample RGB values from the image (6 blue dots per patch) and arrange them in polar coordinates.

4.4. Window-based self attention

The Swin transformer architecture [28] uses window-based self-attention, where attention is computed on non-overlapping windows of $M \times M$ patches. Here, we maintain that strategy, but the polar nature of our patches allows for an additional design choice. Namely, the number of patches along azimuth M_φ and radius M_r can be different. As shown in tab. 1, we define two variants: M patches along both azimuth and radius, $M_r = M_\varphi = M$ (DarSwin-RA); or $M_\varphi = M^2$ patches along azimuth and $M_r = 1$ patch along radius (DarSwin-A). Note that we break this rule when the input resolution across radius or azimuth becomes lower than M to maintain the number of patches in each window equals to M^2 at each Stage (see tab. 1). We experimentally found (see sec. 5.4) that computing attention across M^2 patches along the azimuth (DarSwin-A) yielded improved performance.

Similarly, shifted window self-attention (used in [28] to introduce connections across windows at each stage of the network) is done by displacing the windows by M patches along the azimuth.

4.5. Angular relative positional encoding

The Swin transformer relative positional encoding [28] includes a position bias $B \in \mathbb{R}^{M^2 \times M^2}$, where M^2 is the number of patches in a window, added to each head when computing similarity:

$$\text{Att}(Q, K, V) = \text{Softmax}(QK^T/\sqrt{d} + B)V\,, \qquad (6)$$

where $Q, K, V \in \mathbb{R}^{M^2 \times d}$ are the queries, keys, and values matrices respectively; and d is the query/key dimension.

Since DarSwin follows a radial partitioning, we employ an angular relative positional encoding to capture the relative position between tokens with respect to incident θ and azimuth φ angles. We divide the position bias B in eq. (6) into two parts: B_θ and B_φ, the incident-angular and azimuthal relative position bias respectively. Given the i-th token angular coordinates

$$\theta_i = \frac{\theta_{\max}(i - 0.5)}{N_r} \text{ and } \varphi_i = \frac{2\pi(i - 0.5)}{N_\varphi}\,, \qquad (7)$$

where $\theta_{\max}$ is the half field of view. The relative angular positions $(\Delta\theta, \Delta\varphi)$ between tokens i, j are given by

$$\begin{aligned} \Delta\theta &= \theta_i - \theta_j\,, \text{where } i, j \in [1, ..., N_\theta]\,, \text{and} \\ \Delta\varphi &= \varphi_i - \varphi_j\,, \text{where } i, j \in [1, ..., N_\varphi]\,. \end{aligned} \qquad (8)$$

The two tensors B_φ and B_θ are defined as

$$B_\theta = a_{\Delta\theta} \sin(\Delta\theta) + b_{\Delta\theta} \cos(\Delta\theta)\,, \qquad (9)$$

$$B_\varphi = a_{\Delta\varphi} \sin(\Delta\varphi) + b_{\Delta\varphi} \cos(\Delta\varphi)\,. \qquad (10)$$

Table 1: Model architecture specification and variants. In both variants, we adapt the small architecture of Swin Transformer [28], which has four stages of patch merging. **Input Resolution** : Resolution of feature map at every stage. **Window Size** : Window size (M_r, M_φ), M_r : Number of patches along radial axis in a window, M_φ : Number of patches along azimuth axis in a window. **DarSwin-RA**: Architecture variant with four patches along both azimuth and radial axis and merging 2×2 neighborhood patches along the radius and azimuth. **DarSwin-A**: architecture variant with 16 patches along the azimuth axis and one patch along the radial axis in a window and merging 1×4 neighborhood patches along the radius and azimuth.

DarSwin-RA		DarSwin-A	
Input Resolution	Window Size (M_r, M_φ)	Input Resolution	Window Size (M_r, M_φ)
Stage-1 (16, 64)	(4, 4)	Stage-1 (16, 64)	(1, 16)
Stage-2 (8, 32)	(4, 4)	Stage-2 (16, 16)	(1, 16)
Stage-3 (4, 16)	(4, 4)	Stage-3 (16, 4)	(4, 4)
Stage-4 (2, 8)	(2, 8)	Stage-4 (16, 1)	(16, 1)

Here, a_* and b_* are trainable coefficients. Since the relative positions in a window along angular and azimuth axes ranges from $[-M_\theta + 1, M_\theta - 1]$ and $[-M_\varphi + 1, M_\varphi - 1]$ respectively, where M_θ and M_φ are number of patches in a window (see sec. 4.4, here $M_\theta = M_r$). We parameterize two bias matrices $\hat{B}_\varphi \in \mathbb{R}^{(2M_\varphi - 1) \times 2}$ and $\hat{B}_r \in \mathrm{R}^{(2M_\theta - 1) \times 2}$. Hence a_* and b_* are taken from $\hat{B}_\theta$ and $\hat{B}_\varphi$.

Finally, the two tensors $B_\theta, B_\varphi \in \mathbb{R}^{M_r^2 \times M_\varphi^2}$ are built on all pairs of tokens. The final attention equation is thus

$$\text{Att}(Q, K, V) = \text{Softmax}(QK^T/\sqrt{d} + B_\theta + B_\varphi)V\,. \qquad (11)$$

4.6. Polar patch merging

Similar to window-based self attention (sec. 4.4), the polar nature of our architecture enables many possibilities when merging patches. For example, we could merge 2×2 neighboring patches (DarSwin-RA) or 1×4 merge along azimuth (DarSwin-A) as shown in tab. 1. We found the azimuth merging strategy to outperform the others in our experiments (see sec. 5.4).

5. Classification experiments

To evaluate the efficacy of our proposed DarSwin encoder, we perform a series of experiments on image classification. Since there exists no classification dataset for wide angle images, we instead create synthetically distorted images using 200 randomly chosen classes from the ImageNet1k dataset [23].

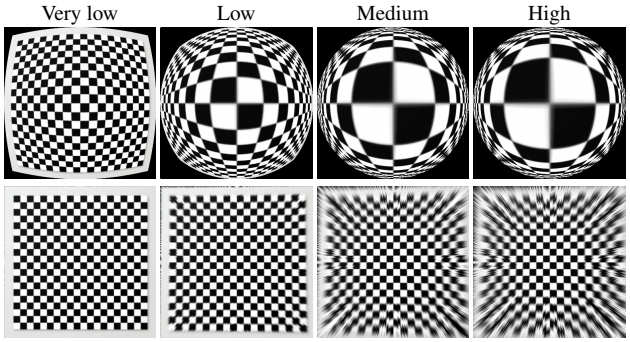


Figure 6: Visualization of a checkerboard pattern distorted according to each distortion level at its original resolution of (224, 224) and downsampled to (64, 64) in our four training sets. From left to right: very low, low, medium, and high. The second row represents the respective undistorted images.

5.1. Dataset

To evaluate our approach on a wide range of conditions, we employ the unified spherical projection model (c.f. sec. 3) to synthetically distort the (perspective) ImageNet1k images. For this, we warp the images at the original pixel resolution (224×224) then downsample to (64×64) for all experiments.

Training sets We generate four different training sets with-different levels of distortion, defined by the distortion parameter ξ: "very low" ($\xi \in [0.0, 0.05]$), "low" ($\xi \in [0.2, 0.35]$), "medium" ($\xi \in [0.5, 0.7]$), and "high" ($\xi \in [0.85, 1.0]$). Fig. 6 shows examples of a checkerboard image distorted at each level. Training images are distorted on-the-fly during training with the distortion level sampled uniformly from the interval mentioned above. Each training set contains 260,000 images and 10,000 validation images, over 200 classes.

Test set Test sets of 30,000 images over 200 classes are generated using the same procedure. Here, ξ is determined once for each image and kept fixed. And we test for all ξ values between $[0, 1]$ to evaluate generalization to different lens distortions.

5.2. Baselines and training details

We compare our approach with the following baselines: Swin-S [28], Deformable Attention Transformer (DAT-S) [44], and Swin-S on input undistorted images dubbed "Swin(undis)". As with our DarSwin, this last baseline has knowledge of the distortion parameters whereas the first two do not. Note that we do not include comparisons to methods which estimate distortion [47, 45, 21, 52]. Indeed, the spherical projection model (sec. 3) is bijective: the undistortion function has a closed form and is exact. Therefore, the "Swin (undis)" method serves as an upper bound to all self-calibration methods because it is, in essence, being given the "ground truth" undistortion.

All three baselines employ 32 divisions along the image width and height. For DarSwin, we use $N_r = 16$ and $N_\varphi = 64$ divisions along the radius and azimuth respectively, which yields the same total number of 1024 patches for all methods. All three baselines are trained with a window size (4, 4) on our synthetically distorted training sets.

All methods use the AdamW optimizer on a batch size of 128 using a cosine decay learning rate scheduler and 20 epochs of linear warm-up. We use an initial learning rate of 0.001 and a weight decay of 0.05. We include all of the augmentation and regularization strategies of [28], except for random crop and geometric transformation (like shearing and translation). Our model requires 0.03M additional parameters over the Swin baseline which contains 48M, representing a 0.061% increase.

5.3. Zero-shot lens distortion generalization

We are interested in evaluating whether our distortion-aware method can better generalize to other unseen lenses at test time. Hence, we train each approach on a level of distortion (c.f. sec. 5.1) and evaluate them on all distortion values $\xi \in [0, 1]$. For this, entire test set is distorted using a single ξ value, and we repeat this process for every $\xi \in [0, 1]$ to simulate different lens distortion.

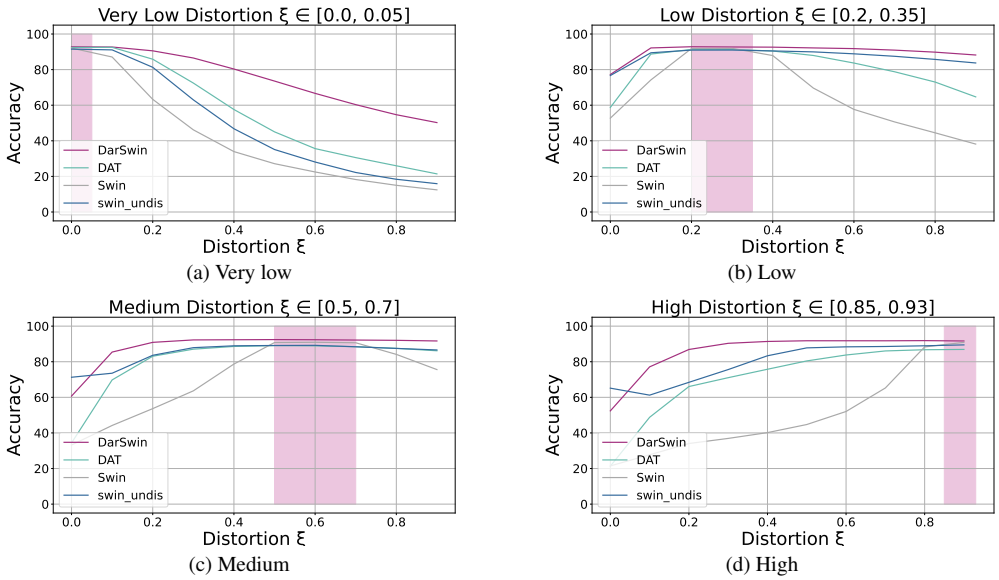
Results are reported in fig. 7, where the distribution of training distortions is drawn in pink. We observe that DarSwin performs on par with the baselines when test distortions overlap with training, but shows much greater generalization capabilities outside of the training domain. Furthermore, tab. 2 shows the top-1 accuracy for the test set distorted with $\xi = 0.4$, which is in none of the training intervals. Again, we note that DarSwin yields better generalization accuracy (without fine-tuning) than all baselines, even Swin (undis) which also has access to the ground truth distortion function (see sec. 5.2).

Table 2: Comparisons of top-1 accuracy of the models trained on four levels of distortion (pink regions in fig. 7) and tested on distortion level $\xi = 0.4$. Each row is color-coded as best and second best.

Methods	Very Low	Low	Medium	High
DarSwin	80.33	92.61	92.35	91.39
Swin	33.94	87.90	78.7	40.1
Swin (undis)	47.48	91.52	91.07	87.34
DAT	57.5	90.4	88.5	75.7

5.4. Ablations

We ablate some important design elements (tab. 3) and training strategies (tab. 4). All ablations are performed using



(a) Very low (b) Low (c) Medium (d) High

Figure 7: Top-1 classification accuracy (mean) as a function of test distortion for our method (DarSwin-A) and previous state of the art: DAT [44], Swin [28], and Swin (undis) (see text). All methods are trained on a restricted set of lens distortion curves (indicated by the pink shaded regions): (a) Very low, (b) low, (c) medium and (d) high distortion. We observe zero-shot adaptation to lens distortion of each approach by testing across all $\xi \in [0, 1]$.

DarSwin-A trained on the "low" distortion ($\xi \in [0.2, 0.35]$) and tested on all the distortion levels.

Positional encoding (PE) We compare our angular relative position encoding (sec. 4.5) with two versions of the Fourier-based polar positional encoding from [32] in the first part of tab. 3. First, polar PE encodes patches using their relative distortion-aware radial lengths r and azimuth values φ. Second, angular PE encodes patches using the relative value of incident angle θ and azimuth φ. While angular PE yields better performance, we observe that our angular relative PE improves the performance even further.

Number of sampling points along radius S_r We observe in tab. 3 that DarSwin with $S_r = 10$ sampling points along the radius outperforms the models with $S_r = 5$ and $S_r = 2$ sampling points along the radius by just over (1-6)%.

Window formations and merging Ablations of different window formation and merging strategies : along azimuth (DarSwin-A) or along radius+azimuth (DarSwin-RA) respectively are reported in tab. 3. We observe that merging along the azimuth outperforms the merging along randius+azimuth by approximately (1-4)% on each distortion level.

Jittering. As discussed in sec. 4.3, when patches are sampled for the linear embedding layer, an augmentation strategy is used to jitter the sample points in the patch. According to tab. 4, this jittering augmentation improves the performance by (1–2)%.

Table 3: Ablation study on different design elements of DarSwin, including different positional encoding, number of sampling points along the radius n_r, and window formation and merging strategy: DarSwin-A (azimuth) or DarSwin-RA (radius+azimuth).

	Very low	Low	Medium	High
Angular relative PE	83.43%	92.8%	91.5%	88.27%
Polar PE [32]	79.6%	92.0%	90.1%	85.4%
Angular PE [32]	81.47%	92.617%	91.11%	87.4%
$S_r = 10$	83.43%	92.8%	91.5%	88.27%
$S_r = 5$	81.81%	91.94%	90.00%	85.19%
$S_r = 2$	78.9%	90.4%	87.9%	82.4%
DarSwin-A	83.43%	92.8%	91.5%	88.27%
DarSwin-RA	79.7%	92.7%	90.3%	84.7%

Distortion aware (DA) sampling. We observe in tab. 4 that distortion aware sampling improves performance by almost (2–6)% compared to uniform sampling in a patch.

Distortion aware (DA) patch partition. We observe in tab. 4 that without lens information, DarSwin cannot generalize to unseen distortion.

Table 4: Ablation study on different types of sampling techniques and augmentation strategies on DarSwin-A. For jittering we compare "no jittering" (all samples are given to the MLP for linear embedding as is) with "with jittering"(jittering is applied on the sample points in a patch). For sampling, we compare "Uniform sampling": points are sampled uniformly inside a patch; to "Distortion aware (DA) sampling": lens information is taken into account to sample points inside a patch. "Distortion aware (DA) partition" DarSwin uses lens information to partition the patch.

	Very low	Low	Medium	High
With jittering	83.43%	92.8%	91.5%	88.27%
No jittering	81.5%	92.6%	91.16%	87.4 %
DA sampling	83.43%	92.8%	91.5%	88.27%
Uniform sampling	82.4%	92.5%	91.2%	86.2%
DA partition	83.43%	92.8%	91.5%	88.27%
w/o DA partition	52.2%	91.3%	86.2%	75.5%

Table 5: Comparison of our method with baselines on generalization across projection model. We record top-1 accuracy on the polynomial projection test set for all methods trained on four distortion levels of the spherical projection model. Each row is color-coded as best and second best.

Training Levels	Very Low	Low	Medium	High
Swin(undis)	85.9%	87.1%	71.3%	60.3%
DAT	65.4%	85.9%	78.9%	63.8%
Swin	33.5%	70.2%	42.3%	25.6%
Ours	82.7%	84.8%	78.4%	65.5%

5.5. Generalization over projection models

In tab. 5, we aim to check for robustness for domain shift at inference due to the use of a different distortion model (c.f. sec. 3). In particular, we use the test set of 30,000 images and distort them using the distortion parameters of a 4-degree polynomial projection model. Each image in the test set is assigned four different distortion parameters randomly sampled from a uniform distribution. The undistortion is not exact since the polynomial projection function is not invertible. Hence Swin (undis) performs only better than DarSwin on low distortions, but the performance degrades on medium to high distortion levels. Our model performs better generalization on all distortion levels. DAT fails to generalize on low distortion levels, and Swin (undis) fails to generalize on high distortions.

6. Discussion

This paper presents DarSwin, a new distortion aware vision transformer which adapts its structure to the lens distortion profile of a (calibrated) lens. DarSwin achieves state-of-the-art performance on zero-shot adaptation (without pretraining) on different lenses on classification using synthetically distorted images from the ImageNet1k dataset.

Limitations and future research directions While our method demostrates state-of-the-art performance, it suffers from some limitations. First, the distortion aware sampling strategy is shown to be effective for zero-shot adaptation, the sparsity of sampling points and necessity to interpolate pixel values may affect the performance of the model. While this issue is partially alleviated using our proposed jittering augmentation technique, other strategies may also be possible. Second, our model assumes knowledge of the lens distortion profile, hence it is appropriate only for the calibrated case. We hope to extend our work to uncalibrated lenses, for example by taking inspiration from [15, 47, 10]. Finally, while our experiments demonstrate promising performance on classification experiments, we wish to expand to per-pixel tasks, such as semantic classification and depth estimation. Here, making pixel decoders distortion aware is an exciting direction for future work.

Acknowledgments This research was supported by NSERC grant ALLRP-567654, Thales, an NSERC USRA to J. Lagüe, and the Digital Research Alliance Canada. We thank Yohan Poirier-Ginter, Frédéric Fortier-Chouinard, Adam Tupper and Justine Giroux for proofreading.

References

[1] Ola Ahmad and Freddy Lecue. FisheyeHDK: Hyperbolic deformable kernel learning for ultra-wide field-of-view image recognition. In *AAAI*, 2022. 2

[2] João P. Barreto. A unifying geometric representation for central projection systems. *Comput. Vis. Img. Underst.*, 2006. 3

[3] Conrad Beck. Apparatus to photograph the whole sky. *J. Scientific Inst.*, 2(4):135–139, 1925. 3

[4] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Velickovic. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *CoRR*, abs/2104.13478, 2021. 2

[5] Pierre-Andre Brousseau and Sebastien Roy. Calibration of axial fisheye cameras through generic virtual central models. In *Int. Conf. Comput. Vis.*, 2019. 2

[6] Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. In *Int. Conf. Learn. Represent.*, 2018. 2

[7] Taco S. Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional

networks and the icosahedral CNN. In *Int. Conf. on Mach. Learning*, 2019. 2

[8] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Int. Conf. Comput. Vis.*, 2017. 2, 3

[9] Liuyuan Deng, Ming Yang, Hao Li, Tianyi Li, hu Bing, and Chunxiang Wang. Restricted deformable convolution-based road scene semantic segmentation using surround view cameras. *IEEE Trans. Int. Trans. Syst.*, 08 2019. 1, 2

[10] Frédéric Devernay and Olivier Faugeras. Straight lines have to be straight automatic calibration and removal of distortion from scenes of structured environments. *Mach. Vis. Appl.*, 13, 08 2001. 2, 8

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. Learn. Represent.*, 2020. 2

[12] Clara Fernandez-Labrador, Alejandro Perez-Yus, Gonzalo Lopez-Nicolas, and Jose J Guerrero. Layouts from panoramic images with geometry and deep learning. *IEEE Rob. Autom. Letters*, 3(4):3153–3160, 2018. 2

[13] Margaret M. Fleck. Perspective projection: The wrong imaging model. *IEEE Trans. Reliability*, 1995. 3

[14] Robin Hill. A lens for whole sky photographs. *Quart. J. Royal Meteo. Soc.*, 50(211):227–235, 1924. 3

[15] Yannick Hold-Geoffroy, Kalyan Sunkavalli, Jonathan Eisenmann, Matthew Fisher, Emiliano Gambaretto, Sunil Hadap, and Jean-François Lalonde. A perceptual measure for deep single image camera calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 8

[16] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *Eur. Conf. Comput. Vis.*, 2018. 1

[17] Ciarán Hughes, Patrick Denny, Edward Jones, and Martin Glavin. Accuracy of fish-eye lens models. *Applied optics*, 49(17):3338–3347, 2010. 3

[18] Sujin Jang, Joohan Na, and Dokwan Oh. Dada: Distortion-aware domain adaptation for unsupervised semantic segmentation. In *Adv. Neural Inform. Process. Syst.*, 2022. 2

[19] Juho Kannala and Sami S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):1335–1340, 2006. 2, 3

[20] Aditya Khosla, Tinghui Zhou, Tomasz Malisiewicz, Alexei A Efros, and Antonio Torralba. Undoing the damage of dataset bias. In *Eur. Conf. Comput. Vis.*, 2012. 1

[21] Byunghyun Kim, Dohyun Lee, Kyeongyuk Min, Jongwha Chong, and Inwhee Joe. Global convolutional neural networks with self-attention for fisheye image rectification. *IEEE Access*, 10:129580–129587, 2022. 2, 6

[22] Hyungtae Kim, Eunjung Chae, Gwanghyun Jo, and Joonki Paik. Fisheye lens-based surveillance camera for wide field-of-view monitoring. In *IEEE Int. Conf. Cons. Elec.*, 2015. 1

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Adv. Neural Inform. Process. Syst.*, 2012. 2, 5

[24] Varun Ravi Kumar. Surround-view cameras based holistic visual perception for automated driving. *arXiv preprint arXiv:2206.05542*, 2022. 2

[25] Varun Ravi Kumar, Senthil Yogamani, Hazem Rashed, Ganesh Sitsu, Christian Witt, Isabelle Leang, Stefan Milz, and Patrick Mäder. Omnidet: Surround view cameras based multi-task visual perception network for autonomous driving. *IEEE Robot. Autom. Letters*, 6(2):2830–2837, 2021. 2

[26] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022. 2

[27] Shuying Liu and Weihong Deng. Very deep convolutional neural network based image classification using small training sample size. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, 2015. 2

[28] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Int. Conf. Comput. Vis.*, 2021. 1, 2, 4, 5, 6, 7

[29] Christopher Mei and Patrick Rives. Single view point omnidirectional camera calibration from planar grids. In *Int. Conf. Robot. Aut.*, 2007. 3

[30] R. Melo, M. Antunes, J. P. Barreto, G. Falcão, and N. Gonçalves. Unsupervised intrinsic calibration from a single frame using a "plumb-line" approach. In *Int. Conf. Comput. Vis.*, 2013. 2

[31] Kenro Miyamoto. Fish eye lens. *J. Opt. Soc. Am.*, 54(8):1060–1061, Aug 1964. 3

[32] Ke Ning, Lingxi Xie, Fei Wu, and Qi Tian. Polar relative positional encoding for video-language segmentation. In *Int. Joint Conf. Art. Intel.*, 2020. 7

[33] Elad Plaut, Erez Ben Yaacov, and Bat El Shlomo. 3d object detection from a single fisheye image without a single fisheye training image. In *IEEE Conf. Comput. Vis. Pattern Recog. Worksh.*, 2021. 2

[34] Clément Playout, Ola Ahmad, Freddy Lécué, and Farida Cheriet. Adaptable deformable convolutions for semantic segmentation of fisheye images in autonomous driving systems. *CoRR*, abs/2102.10191, 2021. 2

[35] S. Ramalingam and Peter Sturm. A unifying model for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(7):1309–1319, 2017. 2

[36] Hazem Rashed, Eslam Mohamed, Ganesh Sistu, Varun Ravi Kumar, Ciaran Eising, Ahmad El-Sallab, and Senthil Yogamani. Generalized object detection on fisheye cameras for autonomous driving: Dataset, representations and baseline. In *Wint. Conf. Appl. Comput. Vis.*, 2021. 2

[37] Hazem Rashed, Eslam Mohamed, Ganesh Sistu, Varun Ravi Kumar, Ciaran Eising, Ahmad Sallab, and Senthil Yogamani. Fisheyeyolo: Object detection on fisheye cameras for autonomous driving. In *Adv. Neural Inform. Process. Syst.*, 12 2020. 2

[38] Dieter Schmalstieg and Tobias Höllerer. Augmented reality: Principles and practice. In *IEEE Virt. Reality*, 2017. 1

[39] Yu-Chuan Su and Kristen Grauman. Kernel transformer networks for compact spherical convolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2

[40] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014. 2

[41] Álvaro Sáez, Luis M. Bergasa, Eduardo Romeral, Elena López, Rafael Barea, and Rafael Sanz. Cnn-based fisheye image real-time semantic segmentation. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018. 2

[42] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2011. 1

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Adv. Neural Inform. Process. Syst.*, 2017. 2

[44] Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Vision transformer with deformable attention. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2, 6, 7

[45] Z. Xue, N. Xue, G. Xia, and W. Shen. Learning to calibrate straight lines for fisheye image rectification. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2, 6

[46] Yaozu Ye, Kailun Yang, Kaite Xiang, Juan Wang, and Kaiwei Wang. Universal semantic segmentation for fisheye urban driving images. *IEEE Int. Conf. Syst. Man Cyber.*, pages 648–655, 2020. 2

[47] Xiaoqing Yin, Xinchao Wang, Jun Yu, Maojun Zhang, Pascal Fua, and Dacheng Tao. Fisheyerecnet: A multi-context collaborative deep network for fisheye image rectification. In *Eur. Conf. Comput. Vis.*, 2018. 2, 6, 8

[48] Xianghua Ying and Zhanyi Hu. Can we consider central catadioptric cameras and fisheye cameras within a unified imaging model. In *Eur. Conf. Comput. Vis.*, 2004. 3

[49] Senthil Yogamani, Ciaran Hughes, Jonathan Horgan, Ganesh Sistu, Sumanth Chennupati, Michal Uricar, Stefan Milz, Martin Simon, Karl Amende, Christian Witt, Hazem Rashed, Sanjaya Nayak, Saquib Mansoor, Padraig Varley, Xavier Perrotton, Derek Odea, and Patrick Pérez. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *Int. Conf. Comput. Vis.*, 2019. 1, 2, 3

[50] Heeseung Yun, Sehun Lee, and Gunhee Kim. Panoramic vision transformer for saliency detection in 360° videos. In *Eur. Conf. Comput. Vis.*, 2022. 2

[51] Jiaming Zhang, Kailun Yang, Chaoxiang Ma, Simon Reiß, Kunyu Peng, and Rainer Stiefelhagen. Bending reality: Distortion-aware transformers for adapting to panoramic semantic segmentation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 2

[52] Mi Zhang, Jian Yao, Menghan Xia, Kai Li, Yi Zhang, and Yaping Liu. Line-based multi-label energy optimization for fisheye image rectification and calibration. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2015. 2, 6

[53] Daquan Zhou, Bingyi Kang, Xiaojie Jin, Linjie Yang, Xiaochen Lian, Qibin Hou, and Jiashi Feng. Deepvit: Towards deeper vision transformer. *arXiv preprint arXiv:2103.11886*, 2021. 2

[54] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2, 3

[55] Nikolaos Zioulis, Antonis Karakottas, Dimitrios Zarpalas, and Petros Daras. Omnidepth: Dense depth estimation for indoors spherical panoramas. In *Eur. Conf. Comput. Vis.*, 2018. 2