Rotation Equivariant Orientation Estimation for Omnidirectional Localization

Chao Zhang[1], Ignas Budvytis[1,2], Stephan Liwicki[1], and Roberto Cipolla[1,2]

[1] Cambridge Research Lab, Toshiba Europe Ltd, Cambridge, UK
{chao.zhang, stephan.liwicki}@crl.toshiba.co.uk
[2] Department of Engineering, University of Cambridge, Cambridge, UK
{ib255, rc10001}@cam.ac.uk

Abstract. Deep learning based 6-degree-of-freedom (6-DoF) direct camera pose estimation is highly efficient at test time and can achieve accurate results in challenging, weakly textured environments. Typically, however, it requires large amounts of training images, spanning many orientations and positions of the environment making it impractical for medium size or large environments. In this work we present a direct 6-DoF camera pose estimation method which alleviates the need for orientation augmentation at train time while still supporting any SO(3) rotation at test time. This property is achieved by the following three step procedure. Firstly, omni-directional training images are rotated to a common orientation. Secondly, a fully rotation equivariant DNN encoder is applied and its output is used to obtain: (i) a rotation invariant prediction of the camera position and (ii) a rotation equivariant prediction of the probability distribution over camera orientations. Finally, at test time, the camera position is predicted robustly due to an in-built rotation invariance, while the camera orientation is recovered from the relative shift of the peak in the probability distribution of camera orientations. We demonstrate our approach on synthetic and real-image datasets, where we significantly outperform standard DNN-based pose regression, (i) in terms of accuracy when a single training orientation is used, and (ii) in training efficiency when orientation augmentation is employed. To the best of our knowledge, our proposed rotation equivariant DNN for localization is the first direct pose estimation method able to predict orientation without explicit rotation augmentation at train time.

1 Introduction

Visual localization aims at finding the position and orientation of the input camera sensor with respect to a known environment, using image alone. Its significance in many practical applications, including autonomous driving [1], robotics [2] and augmented reality [3, 4], inspires numerous publications over the years [5–19]. Nevertheless, robust localization in complex environments remains a challenge to-date [20–23].

Classical localization using feature correspondences dates back decades ago [5–7], and some of these methods remain competitive today especially in mid- and

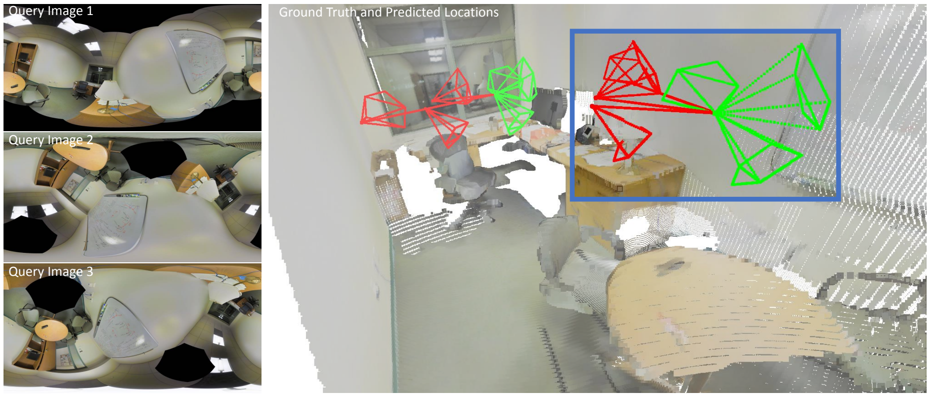


Fig. 1. Given equirectangular input of arbitrary orientation (left images), our method predicts 6-DoF camera poses (green - ground truth, red - predictions). Note that our method does not require any rotation augmentation at train time and hence significantly increases convergence speed. Left column, shows query images from Stanford 2D3DS [51] dataset under arbitrary orientations and corresponding camera pose predictions are highlighted on the right. Our method is capable of rotation invariant position prediction and efficient orientation estimation.

large-scale environments [24, 21]. Nevertheless, recent advances in deep learning particularly improve upon localization with challenging lighting, appearance changes and in run-time performance [19, 10, 14]. In our work, we target deep localization with a particular focus on generalization for invariance to camera orientation.

Localization with deep neural networks (DNNs) has been tackled using image retrieval [8], relative pose regression [9–11], scene coordinate regression [12–15] and direct camera pose estimation [16–19] approaches. Image retrieval approaches formulate localization as a problem of finding the image most similar to the query image. Relative pose regression methods use image feature correspondences between query and retrieved image to further refine the pose. Scene coordinate regression methods perform an efficient image to point cloud feature matching to find camera pose, while direct pose estimation approaches typically regress the position and orientation of the camera directly, in an end-to-end setup. We deal with the former type as we present a new approach to pose regression under challenging rotations at test time, but our method is applicable to other frameworks.

PoseNet [16] presents an early approach to direct pose estimation using DNNs, and it has been a popular framework since [17, 19]. Recently, however, [23] highlighted the limitations of direct pose regression methods, and compared their performance to networks performing the image retrieval task. In particular, both type of methods struggle to generalise to unseen, novel viewpoints, hence collection of large amounts of images (or utilisation of rendered views) is required if good test performance is to be expected. In [25] image retrieval is improved by enhancing original viewpoints with additional synthetically generated views. Similarly, warped RGB images with depth data are exploited to improve deep pose regression in [18], while [19] applies novel view generation for DNNs by leveraging the sparsity of SIFT features [26].

Contribution In our work, we take a different approach to improving the deep pose regression framework, as we leverage rotation equivariance to improve view generalization. In particular, inspired by rotation equivariant deep learning on spheres [27–29], we formulate localization for spherical, omni-directional input (see Figure 1). We estimate camera position via regression from the feature response of the rotation invariant decoder, while camera orientation is extracted from the relative orientation in rotation equivariant feature response. Our contributions are as follows:

1. We present the first rotation invariant, deep camera position regression network.
2. We introduce rotation equivariant decoder and a sample efficient classification loss to generate camera rotation estimation in full SO(3) from only **one** rotation observation in training data, without rotation augmentation.
3. We evaluate our method on synthetic and real datasets.

The rest of this work is divided as follows. Section 2 discusses relevant work in localisation. Section 3 provides details of our proposed localization method. Sections 4 and 5 describe the experiment setup and corresponding results.

2 Related Work

In this section, we discuss relevant works on deep learning based localization, and, in particular, direct pose regression methods. We also review equivariant feature learning in the context of pose estimation and spherical deep learning. The interested readers are referred to [20–23] for a more detailed review of localization approaches.

2.1 Localization using Deep Neural Networks

Localization methods which use deep learning have received much interest in recent years [8–19]. Typical approaches tackle the task *via* place recognition, relative pose regression, scene coordinate regression or direct camera pose estimation.

Place Recognition methods formulate the localization task as image retrieval problem where 6-DoF camera pose estimation is not required. Examples include NetVLAD [8] which generates an image descriptor that is aggregated from local descriptors taken from convolutional responses at pixel level. PlaNet [30] formulates the place recognition task as a classification problem using quantized camera coordinates.

Relative Pose Regression methods use image retrieval followed by relative pose estimation between query and retrieved images to refine the pose prediction. In [9], a Siamese network is employed to find the pose transformation between two images. Additionally, end-to-end implementations for the retrieval and refinement are presented in [10] and [11].

Scene Coordinate Regression approaches predict 2D-to-3D point correspondences *via* per pixel regression of scene coordinates. They obtain a 6-DoF camera pose prediction by absolute pose estimation. Differentiable RANSAC optimization is presented in [12] where a DNN is employed for hypothesis scoring. In [13] an angle-based re-projection loss is optimized, while [14] produces a differentiable score from RANSAC inlier counts. A scene coordinate regression with semantic labels is presented in [15].

Direct Pose Estimation provides pose predictions directly using convolutional DNNs. PoseNet [16] regresses to camera pose from image signals alone using a simple deep learning framework. However, absolute pose regression has poor generalization to unseen viewpoints and thus requires well sampled training data [23] if good test performance is expected. An LSTM module is employed by [17] to reduce overfitting in the final fully connected layers as structured feature correlation is introduced. In [18] and [19] original dataset views for training are enhanced with novel view generation, for RGB-D and RGB input respectively.

In our work we consider direct pose estimation due to its train and test time speed and simplicity. We overcome the problem of overfitting by training from densely sampled locations using artificially rendered images of the scene of interest. Our method does not require different samples of orientations and demonstrates orders of magnitude faster convergence at training, and significant improved performance at test time.

2.2 Omni-directional Localization and Equivariant Features

Omni-directional sensor input increases the field of view for the localization task and more importantly rotations are easily handled by simply moving the pixels on the image sphere. Distortions due to camera pose, otherwise introduced by the planar image projection are reduced [31]. Therefore, spherical images present a very attractive input to camera pose estimation. Early works introduce rotation invariant omni-directional localization using color histograms [32–34] or Eigenspace models [35, 36]. Later, SIFT features [26] were adapted to spherical input for wide angle localization tasks [37].

In recent years, many classical feature matching tasks have been revisited with deep learning. For example, SIFT [26] is reformulated using spacial transformer networks [38] to introduce scale and rotation equivariance in DNNs [39, 40], all be it only approximately. In [41], DNNs exploiting harmonic filters are used for guaranteed rotation equivariant features. Neither, however, are trivially applicable to spherical input.

Research on spherical CNN computations include [42] which projects convolutional filters onto the tangent plane of the sphere, [43–46] who apply convolutions on an unfolded icosahedron mesh, and [47] who employ kernels on a HEALPix spheres. Most do not support rotation invariance, while non-trivial equivariance is not supported by any.Rotation invariance is also only approximate due to required mesh alignments. Fundamental rotation equivariance for spherical CNNs is first presented in [27] and [28]. We base our method on [28] which leverages convolutions in a spherical fourier representation to ensure equivariance. In [29], a simplification is introduce to [28] using the Clebsch-Gordan decomposition.

We apply omni-directional localization since it allows for (i) rotation invariant camera position estimation, and based on the convolutions presented by [28] (ii) efficient, rotation equivariant orientation estimation from feature responses. We emphasize, to the best of our knowledge, we propose the first approach which exploits rotation equivariance for the localization task.

3 Leveraging Rotation Equivariance for Localization

Our method, illustrated in Figure 2, consists of three modules: (i) the rotation equivariant spherical encoder, (ii) the rotation invariant decoder for camera position regression, and (iii) the rotationally equivariant orientation classifier which shifts its prediction according to the camera rotation. Below, we describe each part in detail.

3.1 Equivariant Spherical Encoder

The first module consists of a rotation equivariant feature encoder inspired by the spherical convolutions introduced in [28] and the general architecture of ResNet-18 [48]. Specifically, we adapt the work of [28] to perform rotationally equivariant feature extraction for the localization task. Our intention is twofold. Firstly, theoretically sound rotation equivariant feature response can be integrated [28] to provide rotation invariant feature response which is a useful property for robustly predicting camera positions while being agnostic to orientation. Secondly, rotation equivariance allows us to formulate a framework in which only a single orientation needs to be observed during training.

We build our encoder in the following way. First, we apply S^2Conv [28] on the spherical input images to create an output feature map that is indexed by rotations in SO(3). After that, we replicate the typical ResNet-18 architecture [48] using three SO(3)Conv [28] based ResNet blocks (see Figure 2). Here, the S^2Conv convolution corresponds to a convolution of a 2D pattern with a 2D surface on the spherical manifold, where the spherical manifold is shifted using SO(3) rotations, analogous to filter translations in standard convolution operations on planar images. Since the output shifts with input orientation changes, rotation equivariance is achieved, similar to translation equivariance in standard CNN. Note, the output is now indexed by SO(3). The SO(3)Conv operates on feature maps in SO(3), and, similarly to S^2Conv convolutions, it applies a 2D filter on the spherical manifold under rotations in SO(3) (also see [28] for more detail). We emphasize, each output is fully rotation equivariant, as it changes

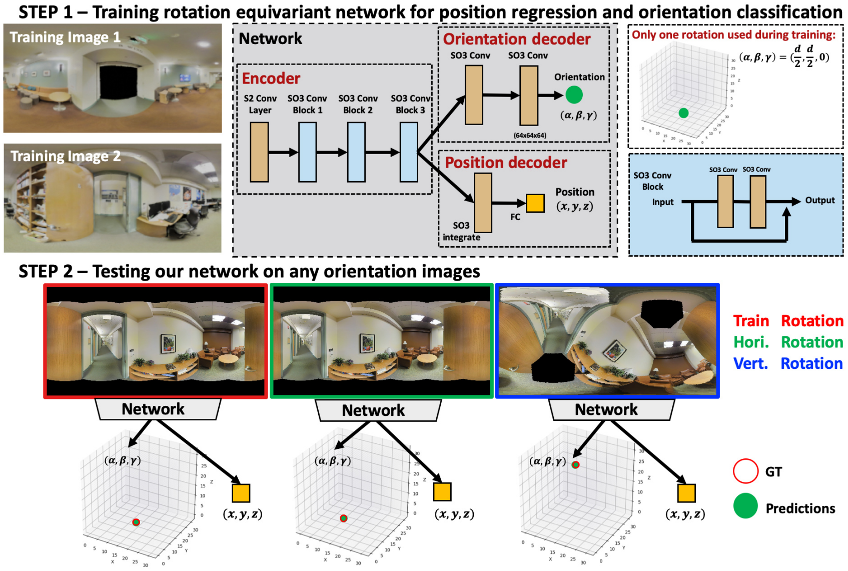


Fig. 2. Training and testing frameworks are shown. For training, we use artificially rendered images of fixed orientation in the global mesh. Our feature encoder is a spherical CNN inspired by the ResNet-18 architecture [48], consisting of a $S2$Conv layer and multiple ResBlocks composed of SO(3)Conv layers. By design feature maps are fully rotation equivariant (also see Figure 3). The output features are fed into a rotation invariant decoder for position regression, while orientation prediction leverages further equivariant SO(3)Conv layers ending with a single channel SO(3) cube feature response. In training we classify a particular cell of this cube for the fixed training orientation. At test time we support arbitrary orientations, as the classification peak rotates accordingly within the SO(3) cube response.

according to input orientation changes. The final layer's output is represented by a 3D cube, where feature responses are indexed using the XYZ Euler angles representation of rotations.

In our implementation, each ResBlock is made of SO(3)Conv-BN-ReLU-SO(3)Conv-BN. At the end of each block, the input is added to the result before ReLU. In comparison to 2D feature maps, a 3D feature map representation requires significantly larger GPU memory resources. To reduce memory requirements, we resize input images to $64 \times 64px$ in our experiments. Following [28], bandwidths relating to the spatial dimension are $32, 16, 8, 8, 8$, and the number of features are $3, 32, 64, 128, 128$. The final output feature map is of size $128 \times 16 \times 16 \times 16$. The resulting 3D feature map encoder forms the input to our position and orientation decoder heads. Figure 3 shows example filter responses.

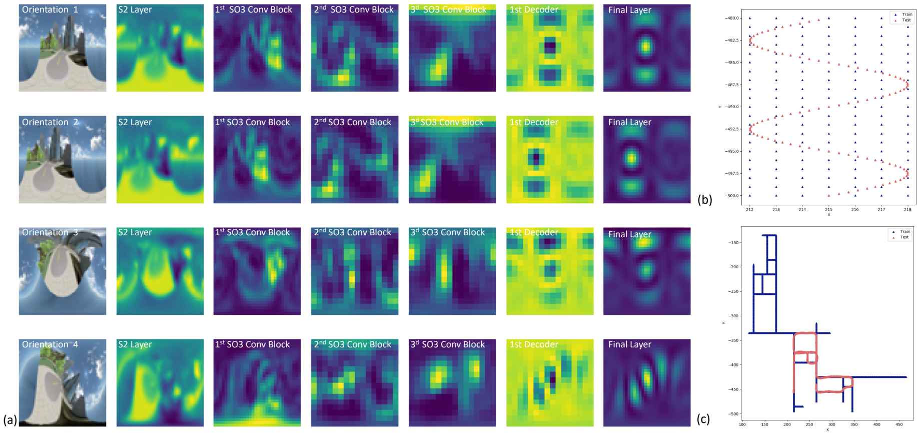


Fig. 3. Analysis of equivariance. (a) Random channels of feature map responses at initial layer, after ResBlocks, and orientation decoder for varying input orientations are shown (we average the output of the features indexed by the roll rotation of the XYZ Euler angles for visualization). Note, feature map responses change following the input image camera orientation. In the final layer for orientation, a strong single peak is formed. We show train locations (blue) and test locations (red) of the SceneCity Grid (b) and SceneCity Small (c) dataset from our experiments in Section 4.

3.2 Invariant Position Regressor

Our pose decoder includes two output heads. The position head is used to regress the 3D camera position $(x, y, z) \in \mathbb{R}^3$. To achieve this, we leverage the rotation-equivariance of our encoder, as we integrate over SO(3) space to produce a rotation-invariant position prediction. The final fully connected layer is used to regress the position vector in $\mathbb{R}^3$. Note, this is similar to standard PoseNet [16] where spatial aggregation is followed by one or more fully connected layers to perform pose regression. In contrast to the special aggregation in PoseNet, our aggregation results in a theoretically fully rotationally invariant feature vector and thus reduces the learning requirements for our network significantly.

Mean squared loss is used for training the position regression. Note that, instead of predicting the position directly, we use PCA whitening to normalize the GT coordinates. We find this helps the general position task, as input values are consistent across training sets even if training coordinates vary largely in scales for different datasets.

3.3 Equivariant Orientation Classifier

To the best of our knowledge, the orientation decoder represents the first rotation equivariant CNN for orientation prediction. In particular, we formulate the orientation prediction as an equivariant classification task. This is motivated by the fact that S^2Conv and $SO(3)$Conv convolutions preserve orientation information throughout layers, and hence the 3D feature response in the XYZ

Euler angle-based cube feature map represents the change in the orientation of inputs. We capitalize on this by forcing train time images to have the same orientation throughout the dataset, without loss of generalization[3], and provide the classification to have a single channel SO(3)Conv layer with softmax and cross entropy loss. We can then recover the orientation of images at test time by finding the relative shift of the softmax layer output in the SO(3) cube of XYZ Euler angles. Examples of feature responses obtained at different layers are visualized in Figure 3. Notice, as we rotate the image along azimuth and elevation, the feature response moves accordingly. The orientation decoder is implemented as SO(3)Conv-SO(3)Conv-Softmax. Note that, in our implementation, the XYZ Euler angles relate to azimuth ($\alpha \in [-\pi, \pi)$), elevation ($\beta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$) and roll ($\gamma \in [-\pi, \pi)$). Since our output is quantized by the classification task, the number of possible rotations is controlled by the output resolution of the cube. Therefore, there is a trade-off between rotational accuracy and efficiency. In our experiments, we show 64^3 to be a promising choice.

4 Experimental Setup

This section provides a brief description of datasets used, network training details and evaluation protocol.

4.1 Datasets

Two datasets with known camera location and orientation are used to evaluate and compare our method with two direct 6-DoF pose regression methods, PoseNet[16] and SphereNet[42].

SceneCity [49]: This dataset contains equirectangular images rendered from two artificial cities, one big and one small. The small city, used in our experiments is first applied to localization in [50]. The environment contains 102 buildings and 156 road segments. Additionally to images, the dataset provides a 3D textured mesh which can be used to render additional images with desired camera locations and orientations. In our evaluations, we use the dataset in two ways: (i) We take a small street segment of the map (about $6m \times 20m$) and render 147 training images and 100 test images. This dataset is denoted SceneCity Grid, shown in Figure 3(b). Training images are densely sampled with equal spacing. And the test images are sampled along a sin curve. (ii) We use the original Small SceneCity locations from [50] for training and testing. The training set consists of 1146 locations, while the test set has 300 locations as shown in Figure 3(c).

[3] Since input is omni-directional image, camera orientation can be adjusted with minimal loss.

Stanford 2D3DS [51] This dataset consists of 1413 equirectangular images captured in an indoor environment. The dataset covers 6 areas of approximately 6000 m^2. It has been widely used in spherical semantic segmentation and depth estimation tasks [43, 52, 46]. The dataset is accompanied by 3D point clouds with textures and ground truth camera poses of provided images. In this work we present the effectiveness of our approach on two scenarios. In both cases, we train the model on synthetically rendered images. They differ in the testing stage. The first is to test on synthetic images, while the latter involves testing on images captured in real scenes. Note, the second task is especially challenging due to the simulation to real gap [53]. In summary, we use all 1413 real images as well as their rendered counterparts as test data. For training data, we render images with random origins within a radius of 30cm around the test locations. In total, 7065 synthetic training images are generated. Following the protocol in [53], we use aggressive non-geometric augmentation (Gaussian blur, additive Gaussian noise, contrast change, image-wise and channel-wise brightness change) of Blender rendered images in order to increase localization performance on real images (see supplementary material for details).

4.2 Training Setup and Evaluation Protocol

Our localization network in Section 3 is implemented in PyTorch [54]. Equirectangular images are resized to $64 \times 64px$ as input to the network in all experiments unless stated otherwise. We use an Adam optimizer with polynomial learning rate scheduler with initial learning rate set to 10^{-4}. We train the network with batch size 20 and up to 3k epochs. We use an $\ell 2$-norm for position regression loss and a cross entropy loss for orientation. We weight the position loss at ×100, as we find the rotation task converges faster. Our network is trained from scratch, as no pretraining is performed.

We evaluate our method based on average (avg) and median (med) Euclidean distance on position and angular divergence. Our method is compared with standard implementations of direct pose regression using planar and spherical convolutions. Full omnidirectional images are provided as input. In particular, PoseNet [16] is implemented with a ResNet-18 [48] backbone followed by two fully connected layers for pose regression. The ResNet-18 features are pretrained on ImageNet [55]. We employ the convolutions in SphereNet [42] to implement a version of PoseNet for spherical input. Here, unlike PoseNet, the convolutions are designed to work on spherical data by applying distortion corrected grid kernels on equirectangular images. In essence, this method applies 2D convolutions on the tangent planes of the sphere. A pretrained spherical VGG-16 [56] backbone is used as the encoder of SphereNet since filters are easily applied to the spherical convolutions. Following [16], the position regression loss is based on $\ell 2$-norm while orientation loss is based on $\ell 2$-norm using quaternion.

Method	Aug. Type	Epochs	Original Rot.				Rand. y-axis Rot.				Random SO(3)			
			Pos.(m)		Rot.(°)		Pos.(m)		Rot.(°)		Pos.(m)		Rot.(°)	
			avg	med	avg	med	avg	med	avg	med	avg	med	avg	med
PoseNet	none	30k	0.20	0.20	0.00	0.00	4.85	4.66	90.0	90.0	**5.78**	5.43	**128**	128
	y-axis	30k	0.29	0.29	1.22	0.99	0.31	0.30	1.81	1.62	**5.73**	5.19	**120**	118
	SO(3)	30k	0.36	0.34	3.94	3.52	0.39	0.39	5.48	5.37	**0.35**	0.35	**5.62**	5.30
SphereNet	none	30k	0.19	0.17	0.00	0.00	5.33	5.31	90.0	90.0	**7.48**	6.47	**128**	128
	y-axis	30k	0.41	0.33	4.41	4.05	6.05	3.56	44.3	44.9	**8.84**	6.48	**122**	120
	SO(3)	30k	0.35	0.32	5.25	4.89	0.42	0.39	7.18	5.86	**0.38**	0.36	**6.98**	6.68
Ours(64^3)	none	3k	0.11	0.09	4.20	4.20	0.12	0.10	4.20	4.20	**0.17**	0.16	**5.05**	5.05
	SO(3)	3k	0.22	0.17	2.54	2.51	0.19	0.17	4.21	4.20	**0.19**	0.16	**2.54**	2.32
Ours(32^3)	none	3k	0.12	0.11	8.34	8.34	0.15	0.14	13.4	14.0	**0.18**	0.17	**13.2**	13.6
Ours(128^3)	none	3k	0.15	0.15	2.10	2.10	0.22	0.21	3.12	3.18	**0.31**	0.28	**2.78**	3.02
PoseNet*	SO(3)	30k	0.63	0.55	3.44	3.26	0.51	0.56	4.51	3.53	**0.72**	0.67	**7.88**	7.57
Ours*(64^3)	none	3k	0.16	0.17	4.20	4.20	0.36	0.31	4.69	4.90	**0.37**	0.31	**4.73**	5.14

Table 1. Ablation study on SceneCity Grid. Our method is compared to PoseNet and SphereNet under varying rotation augmentation on training data, and tested on different positions and (i) original training rotations, (ii) random y-axis rotations, and (iii) random SO(3) rotations. The average (avg) and median (med) position and orientation errors are reported. The most challenging experiments include random orientations in testing (**bold**). We highlight the version of PoseNet, SphereNet and our method used in remaining experiments. Additionally, we show results for varying orientation decoder resolution (shown in parentheses), and PoseNet and our method trained without PCA whitening (denoted by PoseNet* and Ours*(64^3) respectively).

5 Results

In this section we present the following results: (i) an ablation study on augmentation, orientation decoder resolution and PCA whitening, (ii) experiments on larger synthetic datasets and (iii) experiments on real data.

5.1 Ablation Studies on SceneCity Grid

Our ablation uses the SceneCity Grid dataset, which consists of densely sampled training and testing locations to provide optimal data for pose regression networks, and will not suffer from interpolation or extrapolation issues [23]. Results are shown in Table 1.

Rotation Augmentation: It is known that PoseNet [16] and its variants are prone to overfitting to training data [23]. In this section we investigate how geometric training augmentation based on rotations affects the performance. In particular, we investigate testing on images from new positions but with original training rotations, horizontally rotated (rotations around y-axis), and randomly rotated by any SO(3) rotation. Unsurprisingly, both SphereNet and PoseNet demonstrate relatively good performance for the matching pairs of train and test data rotations, achieving better than $0.5m$ accuracy for position. Nevertheless, they overfit to training data and poorly generalize to unseen orientations, decreasing performance to above $5m$ position errors. Note, only with full rotational augmentation, localization with arbitrary camera orientations is successful. We also emphasize, more training epochs (30k versus 3k in our method) are required to make this kind of methods competitive. In contrast, our method demonstrates good position and orientation predictions in all scenarios archiving below $0.2m$ position error with about 5° error on orientations. Such results are even reached for the most challenging case with one rotation during training and any orientation at testing. Thus, our method successfully generalizes one training orientation to arbitrary test orientations.

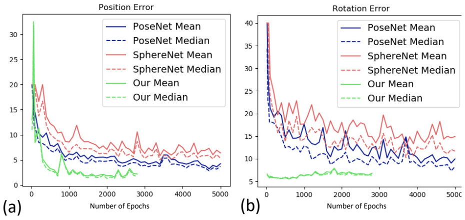
Orientation Decoder Resolution: Our decoder for orientation prediction is parameterized by the size of the output feature cube. Hence, its predictions are inherently quantized. In a second ablation study we evaluate our method with output cube of size 32^3, 64^3 and 128^3. Here, higher resolution improves the orientation accuracy at the expense of slightly reduced position accuracy: 13.2° error with $0.18m$ error for 32^3, 5.05° error with $0.17m$ error for 64^3 and 2.78° error with $0.31m$ error for 128^3. This is due to the fact that the difficulty of the classification task is increased, and thus reduces capacity for improved position loss. Finally, we note that the usage of full rotational augmentation reduces the effect of quantization (from 5.05° to 2.54° error at 64^3), but at the cost of training efficiency. Hence we conclude that a resolution of 64^3 without rotation augmentation is most suitable for our method.

PCA Whitening: Finally we investigate the effect of PCA whitening to conclude our ablation. PCA whitening of position coordinates improves the position prediction for both, PoseNet and our method, by about twice the accuracy (to $0.72m$ and $0.37m$ respectively). It normalizes the position coordinates to a common range which makes training easier.

5.2 Testing on Larger Synthetic Environments

In this section, we evaluate our method on two larger environments: SceneCity Small and Stanford-2D3DS, as described in Section 4.1.

SceneCity Small: Similarly to SceneCity Grid, SceneCity Small is adjusted to have fixed camera orientations for all training poses. During test time, random orientations are used. For PoseNet and SphereNet, full SO(3) rotation augmentation is applied, while our method only sees a single training rotation. Figure 4 shows the performance curve over the evaluated epoch. Our method performs best, and converges much quicker than PoseNet and SphereNet. We emphasize, our decoder for rotation with cross entropy loss converges especially fast. Overall, our method achieves $2.22m$ error for position, while PoseNet and SphereNet have $4.39m$ and $6.07m$ error respectively. Our rotation error is competitive with 9.41° error. Finally we note, PoseNet with only horizontal orientation augmentation fails.



Method	Aug. Type	Pos. avg	Pos. med	Rot. avg	Rot. med
PoseNet	y-axis	**73.2**	65.0	**125**	123
	SO(3)	**4.39**	3.99	**9.06**	8.47
SphereNet	SO(3)	**6.07**	5.20	**11.6**	10.8
Ours	none	**2.22**	1.71	**9.41**	9.13

Fig. 4. Evaluation on SceneCity Small for our method compared to PoseNet and SphereNet. (a) Position and (b) orientation errors are shown over epochs. Note, our method achieves high performance ($< 2m$) within 2k epochs, while rotation error of $< 10°$ is reached after less than 600 epochs (close to theoretical limit due to quantization which is about 5°). In the table we compare the methods quantitatively on the test set of SceneCity Small with random orientations. Additionally, PoseNet with y-axis augmentation is presented.

Method	Aug Type	Synthetic Images								Real Images							
		Orig. Rotation				Rand. Rotation				Orig.				Rand. Rotation			
		Pos.		Rot.		Pos.		Rot.		Pos.		Rot.		Pos.		Rot.	
		avg	med	avg	med	avg	med	avg	med	avg	med	avg	med	avg	med	avg	med
PoseNet	y-axis	1.76	1.58	9.70	5.70	**20.3**	18.4	**121**	117	2.75	1.95	15.4	7.45	**20.2**	18.5	**123**	119
	SO(3)	1.92	1.59	25.6	21.3	**2.10**	1.75	**27.7**	23.2	4.59	2.41	40.6	28.0	**6.25**	3.40	**51.3**	36.5
SphereNet	SO(3)	1.89	1.53	24.7	20.0	**2.35**	1.78	**32.7**	26.5	3.86	2.29	38.5	26.7	**4.81**	2.92	**50.5**	38.3
Ours	none	0.98	0.84	10.9	8.47	**1.79**	1.54	**13.3**	12.6	3.07	1.64	18.2	9.15	**3.57**	2.45	**25.6**	13.1

Table 2. Quantitative results on Standford 2D3DS, comparing our method to PoseNet and SphereNet. A set of columns on the left side of the table, contain testing results on synthetic images, while the columns on the right side contain equivalent results on real images. Our network significantly outperforms competing methods on the random orientation test data for both synthetic and real images.

Stanford 2D3DS: Synthetic results of our method in comparison with PoseNet and SphereNet are shown on the left set of columns of Table 2. For random test rotations, our method with single training orientation outperforms PoseNet and SphereNet on rotation estimation, achieving 13.3°. Notice also, we improve upon position error since our aggregation in the rotation invariant position head simplifies the learning task for the fully connected layer of the regression – here PoseNet and SphereNet perform with 2.10m and 2.35m respectively while our method produces only 1.79m error. Finally, we emphasize, our method is limited by the classification quantization in the orientation decoder as we use a 64^3 output resolution. Qualitative results are shown in Figure 6.

5.3 Results on a Real Image Testing Set

We use the real images for evaluation on Stanford 2D3DS. Note, training data is synthetically rendered using Blender as described in Section 4.1. Again, we compare our method with no rotation augmentation with PoseNet and SphereNet

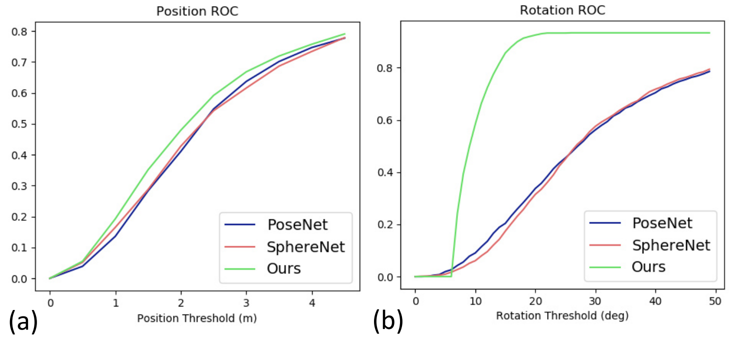


Fig. 5. ROC curve of position and rotation errors. We plot the percentage of data points with predictions below a given threshold for either position (a) or orientation (b) for our method, PoseNet and SphereNet, trained on synthetic images and tested on real images from Stanford 2D3DS. While our method is comparable for position prediction, we significantly improve upon orientation prediction. Here more than 50% of the data points are predicted within 10° error. Competing methods only achieve such an accuracy for predictions within 30°.

with full SO(3) rotation augmentation in training. We test on two versions of test data: one with original orientations and one with random orientations.

The final columns of Table 2 show the results. The performance of PoseNet is slightly improved with horizontal augmentation since the dataset is biased towards horizontally consistent data. Here, PoseNet achieves 2.75m while we reach 3.07m accuracy. Nevertheless, this version of PoseNet overfits and does not generalize to random rotations. Overall, our method performs best, at 3.57m position error and 25.6° orientation error. Qualitative results are shown in Figure 6.

We draw the receiver operating characteristic (ROC) curve in Figure 5, which calculates the percentage of test images within a specific position error or rotation error threshold. The results are generated by testing on real images with original camera location and rotation. Comparing position accuracy, our method obtains competitive results to other methods that need full augmentation. In terms of orientation, our method outperforms other methods with a large margin, having 80% orientations predicted within 15°. This demonstrates our gain of formulating orientation estimation as a classification problem with rotation equivariant response.

Finally we note, in general the performance on real images drops compared to synthetic images in Section 5.2. For example, the performance of our method reduces from 1.79m to 3.57m and 13.3° to 25.6° when moving from synthetic to real data. Similar performance drops are observed by all methods. Although intensive data augmentations is used (Section 4.1), there is a significant performance gap between synthetic and real data. We attribute this issue to direct pose regression being sensitive to the difference of training and testing data. A possible remedy for reducing such a domain gap is to apply image domain translation

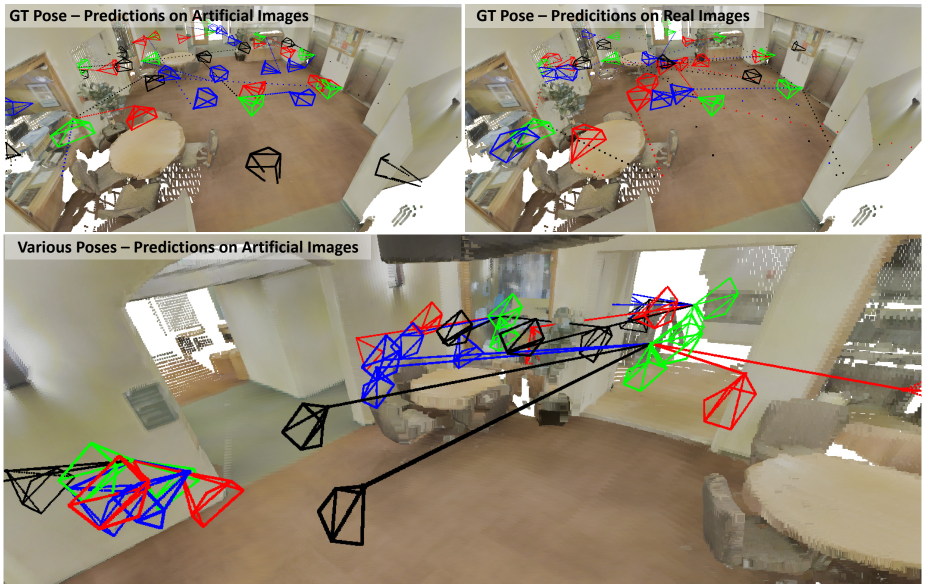

GT Pose – Predictions on Artificial Images
GT Pose – Predicitions on Real Images
Various Poses – Predictions on Artificial Images

Fig. 6. Qualitative predictions of our method (red), PoseNet (blue) and SphereNet (black) on Stanford-2D3DS, quantitative results of which are reported in Table 2. Green camera poses correspond to the ground truth. Pose prediction results on synthetic images at original camera locations is shown top-left, while evaluation on real images is shown top-right. Results of synthetic tests with random orientations are shown at the bottom. Overall, our method predicts poses closer to ground truth. For real images, pose prediction suffers, but our method still provides good camera orientations.

[57] during the test stage before feeding input to the network. Another approach could be to consider auxiliary tasks such as scene coordinate regression or depth estimation to improve the generalization ability. We leave such investigation to future work.

6 Conclusion

In this work we proposed a novel network for rotation equivariant camera pose estimation. This work is motivated by spherical equivariant convolutions, and the need of scalable 6-DoF camera pose estimation networks which can be efficiently trained. Our method learns arbitrary camera orientations from only a single orientation in training, significantly improving training efficiency in terms of epochs needed. In our evaluation, we demonstrate our approach on synthetic and real image input, where we significantly outperform standard DNN-based pose regression. Finally, we emphasize, to the best of our knowledge, our proposed rotation equivariant DNN for omnidirectional localization is the first direct pose estimation method able to predict orientation without explicit rotation augmentation at train time.

References

1. Häne, C., Heng, L., Lee, G.H., Fraundorfer, F., Furgale, P., Sattler, T., Pollefeys, M.: 3d visual perception for self-driving cars using a multi-camera system: Calibration, mapping, localization, and obstacle detection. Image and Vision Computing **68** (2017) 14–27
2. Lim, H., Sinha, S.N., Cohen, M.F., Uyttendaele, M., Kim, H.J.: Real-time monocular image-based 6-dof localization. The International Journal of Robotics Research **34** (2015) 476–492
3. Castle, R., Klein, G., Murray, D.W.: Video-rate localization in multiple maps for wearable augmented reality. In: 2008 12th IEEE International Symposium on Wearable Computers, IEEE (2008) 15–22
4. Middelberg, S., Sattler, T., Untzelmann, O., Kobbelt, L.: Scalable 6-dof localization on mobile devices. In: European conference on computer vision, Springer (2014) 268–283
5. Schindler, G., Brown, M., Szeliski, R.: City-scale location recognition. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE (2007) 1–7
6. Li, Y., Snavely, N., Huttenlocher, D.P.: Location recognition using prioritized feature matching. In: European conference on computer vision, Springer (2010) 791–804
7. Zhang, W., Kosecka, J.: Image based localization in urban environments. In: Third international symposium on 3D data processing, visualization, and transmission (3DPVT'06), IEEE (2006) 33–40
8. Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: Netvlad: Cnn architecture for weakly supervised place recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 5297–5307
9. Melekhov, I., Ylioinas, J., Kannala, J., Rahtu, E.: Relative camera pose estimation using convolutional neural networks. In: International Conference on Advanced Concepts for Intelligent Vision Systems, Springer (2017) 675–687
10. Balntas, V., Li, S., Prisacariu, V.: Relocnet: Continuous metric learning relocalisation using neural nets. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 751–767
11. Nakashima, R., Seki, A.: Sir-net: Scene-independent end-to-end trainable visual relocalizer. In: 2019 International Conference on 3D Vision (3DV), IEEE (2019) 472–481
12. Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., Rother, C.: Dsac-differentiable ransac for camera localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 6684–6692
13. Li, X., Ylioinas, J., Verbeek, J., Kannala, J.: Scene coordinate regression with angle-based reprojection loss for camera relocalization. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 0–0
14. Brachmann, E., Rother, C.: Learning less is more-6d camera localization via 3d surface regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 4654–4662
15. Budvytis, I., Teichmann, M., Vojir, T., Cipolla, R.: Large scale joint semantic re-localisation and scene understanding via globally unique instance coordinate regression. In: 30th British Machine Vision Conference 2019. (2019)
16. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE international conference on computer vision. (2015) 2938–2946

17. Walch, F., Hazirbas, C., Leal-Taixe, L., Sattler, T., Hilsenbeck, S., Cremers, D.: Image-based localization using lstms for structured feature correlation. In: Proceedings of the IEEE International Conference on Computer Vision. (2017) 627–637
18. Handa, A., Patraucean, V., Badrinarayanan, V., Stent, S., Cipolla, R.: Understanding real world indoor scenes with synthetic data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 4077–4085
19. Purkait, P., Zhao, C., Zach, C.: Synthetic view generation for absolute pose regression and image synthesis. In: BMVC. (2018) 69
20. Piasco, N., Sidibé, D., Demonceaux, C., Gouet-Brunet, V.: A survey on visual-based localization: On the benefit of heterogeneous data. Pattern Recognition **74** (2018) 90–109
21. Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., Safari, D., Okutomi, M., Pollefeys, M., Sivic, J., et al.: Benchmarking 6dof outdoor visual localization in changing conditions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 8601–8610
22. Garcia-Fidalgo, E., Ortiz, A.: Vision-based topological mapping and localization methods: A survey. Robotics and Autonomous Systems **64** (2015) 1–20
23. Sattler, T., Zhou, Q., Pollefeys, M., Leal-Taixe, L.: Understanding the limitations of cnn-based absolute camera pose regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 3302–3312
24. Sattler, T., Torii, A., Sivic, J., Pollefeys, M., Taira, H., Okutomi, M., Pajdla, T.: Are large-scale 3d models really necessary for accurate visual localization? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 1637–1646
25. Irschara, A., Zach, C., Frahm, J.M., Bischof, H.: From structure-from-motion point clouds to fast location recognition. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE (2009) 2599–2606
26. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International journal of computer vision **60** (2004) 91–110
27. Esteves, C., Allen-Blanchette, C., Makadia, A., Daniilidis, K.: Learning so (3) equivariant representations with spherical cnns. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 52–68
28. Cohen, T.S., Geiger, M., Köhler, J., Welling, M.: Spherical cnns. In: International Conference on Learning Representations. (2018)
29. Kondor, R., Lin, Z., Trivedi, S.: Clebsch–gordan nets: a fully fourier space spherical convolutional neural network. In: Advances in Neural Information Processing Systems. (2018) 10117–10126
30. Weyand, T., Kostrikov, I., Philbin, J.: Planet-photo geolocation with convolutional neural networks. In: European Conference on Computer Vision, Springer (2016) 37–55
31. Zhang, C., He, S., Liwicki, S.: A spherical approach to planar semantic segmentation. In: British Machine Vision Conference. (2020)
32. Ulrich, I., Nourbakhsh, I.: Appearance-based place recognition for topological localization. In: Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065). Volume 2., Ieee (2000) 1023–1029
33. Blaer, P., Allen, P.: Topological mobile robot localization using fast vision techniques. In: Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292). Volume 1., IEEE (2002) 1031–1036

34. Gonzalez-Barbosa, J.J., Lacroix, S.: Rover localization in natural environments by indexing panoramic images. In: Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292). Volume 2., IEEE (2002) 1365–1370
35. Kröse, B.J., Vlassis, N., Bunschoten, R., Motomura, Y.: A probabilistic model for appearance-based robot localization. Image and Vision Computing **19** (2001) 381–391
36. Winters, N., Gaspar, J., Lacey, G., Santos-Victor, J.: Omni-directional vision for robot navigation. In: Proceedings IEEE Workshop on Omnidirectional Vision (Cat. No. PR00704), IEEE (2000) 21–28
37. Hansen, P., Corke, P., Boles, W., Daniilidis, K.: Scale invariant feature matching with wide angle images. In: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE (2007) 1689–1694
38. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Advances in neural information processing systems. (2015) 2017–2025
39. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: Lift: Learned invariant feature transform. In: European Conference on Computer Vision, Springer (2016) 467–483
40. Zhang, X., Yu, F.X., Karaman, S., Chang, S.F.: Learning discriminative and transformation covariant local feature detectors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 6818–6826
41. Worrall, D.E., Garbin, S.J., Turmukhambetov, D., Brostow, G.J.: Harmonic networks: Deep translation and rotation equivariance. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 5028–5037
42. Coors, B., Condurache, A.P., Geiger, A.: Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 518–533
43. Jiang, C., Huang, J., Kashinath, K., Marcus, P., Niessner, M., et al.: Spherical cnns on unstructured grids. In: ICLR. (2019)
44. Lee, Y., Jeong, J., Yun, J., Cho, W., Yoon, K.J.: Spherephd: Applying cnns on a spherical polyhedron representation of 360deg images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 9181–9189
45. Cohen, T., Weiler, M., Kicanaoglu, B., Welling, M.: Gauge equivariant convolutional networks and the icosahedral CNN. In: Proceedings of the 36th International Conference on Machine Learning. (2019) 1321–1330
46. Zhang, C., Liwicki, S., Smith, W., Cipolla, R.: Orientation-aware semantic segmentation on icosahedron spheres. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 3533–3541
47. Krachmalnicoff, N., Tomasi, M.: Convolutional neural networks on the healpix sphere: a pixel-based algorithm and its application to cmb data analysis. Astronomy & Astrophysics **628** (2019) A129
48. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
49. Zhang, Z., Rebecq, H., Forster, C., Scaramuzza, D.: Benefit of large field-of-view cameras for visual odometry. In: 2016 IEEE International Conference on Robotics and Automation (ICRA), IEEE (2016) 801–808
50. Budvytis, I., Sauer, P., Cipolla, R.: Semantic localisation via globally unique instance segmentation. (2019)
51. Armeni, I., Sax, S., Zamir, A.R., Savarese, S.: Joint 2d-3d-semantic data for indoor scene understanding. arXiv preprint arXiv:1702.01105 (2017)

52. Zioulis, N., Karakottas, A., Zarpalas, D., Daras, P.: Omnidepth: Dense depth estimation for indoors spherical panoramas. In: Proceedings of the European Conference on Computer Vision (ECCV). (2018) 448–465
53. Li, J., Budvytis, I., Cipolla, R.: Indoor re-localisation using synthetic data. , Department of Engineering, University of Cambridge, Technical report: ENG-TR.003,ISSN 2633-68369. (2020)
54. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. In: Advances in neural information processing systems. (2019) 8026–8037
55. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09. (2009)
56. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
57. Murez, Z., Kolouri, S., Kriegman, D., Ramamoorthi, R., Kim, K.: Image to image translation for domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 4500–4509