Modeling Refraction-Induced Image Distortions in Cameras Behind Transparent Objects

PhD Thesis Summary

Szabolcs Pável

Faculty of Mathematics and Computer Science

Babeş-Bolyai University, Cluj-Napoca

Scientific Supervisor

Prof. Dr. Horia F. Pop

2024

Abstract

Video cameras are frequently used in advanced driver assistance systems. The most common mounting position of front facing cameras is near the rearview mirror, behind the windshield of the car. As light is refracted at the surface of the windshield, this acts as an optical element, and causes highly nonlinear, complex distortions in the images. This is a problem for geometric computer vision algorithms, as they assume a precise camera model, which can map the 3D world to the 2D image plane.

After presenting the fundamental concepts of camera models and deep learning methods, two solutions for the problem of distortions caused by a transparent object in the optical path are presented.

First, a model where we explicitly trace the path of light rays through the refractive elements is proposed. Both the global and local components of the distortions are modeled using radial basis functions, and a calibration algorithm based on checkerboard targets is used to find the optimal model parameters. The method is tested on real images captured by a camera placed behind a glass object, as well as on synthetically generated distorted images.

Second, a deep learning based approach is presented, where a convolutional neural network is used to directly estimate the distortions based on a single image. Similar to the first approach, large scale synthetic and real datasets are generated to train the models. The network is trained using image reconstruction based loss functions, and semantic segmentation and optical flow is included as auxiliary task to improve the results.

Contents

Chapter 1

Introduction

Advanced driver assistance systems became widely adopted in modern cars. These systems provide safety functions such as automatic emergency braking, lane departure warning and lane keep assist, traffic sign recognition, intelligent speed assistance, and many others. These systems rely on multiple sensors to build an accurate representation of the environment around the car. These representations need to be precise and robust, as these systems operate in safety-critical environments. The frequently used sensors include cameras, radar, ultrasonic, and in some cases lidar. Among these sensors, cameras are the most versatile. They provide high resolution information at a low cost, allowing to extract rich semantics about the surroundings. As the human road infrastructure is mostly built around vision, for certain tasks, such as interpreting road markings and traffic signs, cameras are the only solution.

When a scene is captured using a camera, information about the structure of the world is lost. In order to reason about the position of objects around the car, this information has to be recovered. This can be done using multi-camera (stereo) systems, where by observing the same object on multiple views, its position can be recovered using triangulation. When using monocular systems with a single camera, we can use images captured at different timestamps instead, and reconstruct the world using Structure-from-Motion approaches. Lately deep learning techniques were also proposed, which can predict depth information based on single images. Regardless of the camera system and technique used, these methods all require a precise mapping from 3D points in the world to 2D pixels on the image – these mappings are called *camera models*.

The ideal *camera model* depends on the properties of the camera itself. For cameras with small field of views simple models based on perspective projection are sufficient. Wide angle cameras, such as fisheye cameras deviate significantly from the perspective model, the images are geometrically *distorted*, which needs to be accounted for. These models characterize the properties of the cameras *globally* – only a small number of parameters are used to describe the mapping across the entire field of view. Global models can fail for example when cameras are placed behind transparent objects, which refract the incoming light rays. This objects can introduce highly nonlinear behavior in the mapping function, which needs to be modeled accordingly – using *local* models.

The parameters of camera models are estimated during *camera calibration*. Calibration methods can be divided into two main categories. When a camera is first deployed, an *initial (offline) calibration* takes place. This process is done using controlled environments, in the presence of specific calibration targets (e.g. checkerboard patterns) and measurement devices. Initial calibration methods can estimate the parameters to a high precision, but they are time-consuming and expensive. While the camera is used, it can become decalibrated: due to environment effects such as large temperature changes and mechanical stresses (e.g. vibrations) the properties of the lenses drift from their initial values. To correct these issues, an *online self-calibration* can be applied. Self-calibration methods adjust the camera while it is operated, without the need for specific calibration targets in the environment.

1.1 Objectives

The motivation of this thesis comes from *smart cameras* used for advanced driver assistance systems. Automotive smart cameras are compact devices, which include both the optical system and processing units (using System-on-Chip) to implement safety functions. These devices are mounted in the area above the rearview mirror of the car, behind the front car windshield. This comes with unique challenges: the refractive windshield in the close proximity of the camera heavily *distorts* the images. These distortions are highly nonlinear, and standard camera models cannot precisely describe them. The work presented in this thesis focuses on modeling these distortions: both an initial calibration method based on explicitly modeling the light refraction at the windshield, and a self-calibration method based on deep learning is proposed.

The first objective of this thesis is to propose an initial calibration algorithm which satisfies the following requirements:

- The model has to be able to describe both the global and local components of the highly nonlinear windshield distortions. This allows taking into account both the overall shape of the windshield, as well as local irregularities of the surface.

- The model has to be physics based: light rays through the transparent media have to traced, as they are refracted on the surfaces. This approach offers an easy way to incorporate prior knowledge of the general properties of the windshield, and it also provides opportunities to more thoroughly analyze distortions.

- Recent advances of automatic differentiation frameworks have to be leveraged, which provide the feasibility to build and optimize complex models. This provides high flexibility when selecting the models we want to use.

Additionally to the initial calibration method, the second objective of this thesis is to propose a self-calibration method, with the requirements:

- The self-calibration method has to be based on deep learning techniques. Deep learning models can be easily integrated into automotive camera systems, as these offer specific accelerators for neural network workloads.

- Similar to the initial calibration method, the neural network has to use a distortion model which is able to represent both global and local deformations of the image.

- For neural network training a dataset based on real-world measurements of windshield distortions has to be built. The dataset should contain synthetic and real images, and the transferability between simulated and real data should be analyzed. This is necessary, as large scale datasets for camera calibration are not publicly available.

- Image reconstruction based loss functions are preferred over losses relying on ground-truth distortion data. These loss functions open the possibility to extend the method to use datasets without ground truth labels.

- Auxiliary tasks shoudl be integrated into the network architecture, and the performance effects of this multi-task approach should be analyzed. Main auxiliary task candidates are semantic segmentation and optical flow, as these predictions are already available in most automotive neural networks.

1.2 Contributions

The contributions of this thesis are the following:

- New solutions for use-cases where the camera is mounted behind a transparent object, which introduces distortions are proposed. The contribution has impact on advanced driver assistance systems, where the camera is mounted above the rearview mirror of the car. Both an initial calibration (Ch. 4, Ch. 5) and a self-calibration (Ch. 6) method are proposed.
- An initial calibration method for modeling the uneven surface of a transparent object (Ch. 4) is proposed. The global shape of the object is assumed to be known, which is often the case in automotive, while the local components of the distortion are describe using a parametric model. Because the global shape is known, we choose the approach to explicitly model the light refraction, contrary to the more explored approach of generalized camera models.
- To test the proposed method, a distorted image dataset is recorded using a Raspberry Pi camera mounted behind a curved glass object. Additionally, a synthetic dataset is used to provide available ground truth distortions.
- The thesis provides an analysis of windshield distortions based on the proposed generative model. This analysis highlights important properties of the distortions, which opens future research possibilities.
- Additionally, a method to model the global shape of the transparent object is proposed, for use-cases where prior knowledge about the object is not available (Ch. 5). The object is modeled as an ellipsoid, which can cover a large variety of use-cases. Using a similar methodology as for the local model, a synthetic dataset is generated to evaluate the method.
- A deep learning based self-calibration method is proposed (Ch. 6). The chosen distortion model is more complex compared to other methods in the literature, as it includes both global and local components. More precise distortion estimation is also facilitated by the use of auxiliary tasks in the network architecture.
- To train the neural networks, two distorted image datasets are constructed – with simulation and real data – based on real-world windshield distortion measurements.

1.3 List of Publications

The contributions of this Thesis were published in the conference papers below. Papers are categorized according to the *Compute Research and Education (CORE) 2018*[1] rankings (categories A*, A, B, C, D). Category D includes conferences that are not listed in the CORE rankings.

- Category A
  - Szabolcs-Botond Lőrincz, **Szabolcs Pável**, and Lehel Csató. Single View Distortion Correction using Semantic Guidance. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE, July 2019.
- Category B
  - **Szabolcs Pável**, Csanád Sándor, and Lehel Csató. Distortion Estimation Through Explicit Modeling of the Refractive Surface. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Image Processing*, pages 17–28. Springer, September 2019.
- Category D
  - **Szabolcs Pável**. An Ellipsoid Object Model of the Refraction Surface. In *Proceedings of the 11th International Conference on Applied Informatics (ICAI)*, volume 2650, pages 272–279. CEUR Workshop Proceedings, January 2020.

1.4 Thesis Outline

The remainder of this thesis is structured as follows.

In **Chapter 2** we lay down the foundations of camera models and camera calibration techniques. We present the classic global models used for normal and fisheye cameras, which are among the most used camera types. We also introduce the concept of image distortions, and review examples from the literature for how image distortions are modeled. Finally, we focus on more general camera models, which can be applied to model a wide variety of distortions, including the ones introduced by transparent objects in front of the camera. The content of this Chapter is based on (31, 71, 73).

[1] CORE Conference Portal: `https://portal.core.edu.au/conf-ranks/`

In **Chapter 3** we present fundamental concepts from machine learning and deep learning. We present radial basis functions, which we extensively use to model distortions from windshields both in our proposed initial calibration and deep learning based calibration method. Next we present the basic building blocks of deep convolutional neural networks, and present some relevant applications. We close the Chapter with presenting specific loss functions based on image reconstructions, also used by us for training our neural networks. The contents of this Chapter are based on (27, 87).

In **Chapter 4** we present an initial calibration method focusing the local properties of the distortions. We assume the global shape of the transparent object (e.g. windshield) to be known, and model the uneven surface using a radial basis function based model. The model parameters are optimized using images of checkerboard calibration targets. We test our method on synthetic and real data, and analyze the observed image distortions. This Chapter is based on our publication (59).

In **Chapter 5** we present a similar method as in the previous Chapter, but this time we focus on the global shape of the transparent object. The global shape is approximated as an ellipsoid, and we calibrate the parameters based on checkerboard targets. We observe symmetries in the ellipsoid model, and propose a regularization scheme to guide the optimization process. This Chapter is based on our publication (58).

In **Chapter 6** we turn our attention to self-calibration, and propose a deep learning based solution. In the lack of available training data, we build two datasets based on real windshield distortion measurements. We once again use radial basis functions to model the highly nonlinear distortions, and train our networks using image reconstruction losses. Auxiliary tasks are also integrated into the network, where we observe that both semantic segmentation and optical flow improve our results. This Chapter is based on our publication (48).

Finally, **Chapter 7** draws the conclusions of our work.

The content of this thesis is based on a list of 90 references, out of which 17 citations represent the newest advances of the field published in the last 5 years, while others summarize the fundamental work in classic computer vision, machine learning and deep learning.

Chapter 2

Camera Models

A digital camera is an imaging system that uses optical elements like lenses and mirrors to focus light onto a photosensitive sensor, mapping the 3D world to a 2D image plane. This mapping results in the loss of depth information, which 3D computer vision algorithms seek to recover using multiple images from different viewpoints. These images provide geometric constraints that help reconstruct the 3D structure. A precise mathematical model, known as a *camera model*, is essential for describing this optical system and is a key component of 3D computer vision algorithms.

2.1 Classic Camera Models

The choice of camera model depends on the camera used. The *pinhole camera model* is the most widely used and employs a perspective projection to map 3D world coordinates to 2D image points. This model describes an ideal camera and is often paired with a *distortion model* to correct deviations from the ideal case. In complex camera systems like fisheye cameras, specific models are required to account for the nonlinear mapping of pixels.

Camera models are parametric, with parameters that describe the camera system's characteristics, such as focal length, distortion parameters, and camera position in 3D space. *Camera calibration* is the process of finding these optimal parameter values.

2.2 Camera Calibration

Camera calibration methods can be divided into *initial calibration* and *self-calibration.* Initial calibration is performed in controlled environments using calibration targets like checkerboards, while self-calibration uses the geometric properties of the scene and the camera's motion to infer calibration parameters.

Zhang's method (88) is a widely known initial calibration technique using planar checkerboard patterns. The process involves capturing multiple images of the pattern, extracting features, computing homographies, and refining the parameters through nonlinear optimization to minimize the reprojection error.

Self-calibration methods leverage the geometric properties of the scene and look for regular structures or rely on the motion of a camera. The method proposed by Devernay and Faugeras (18), use features like straight lines in the environment. Another approach uses point matches from multiple views to optimize epipolar constraints, as proposed by Claus and Fitzgibbon (15).

2.3 Fisheye Cameras

Fisheye cameras have wide-angle lenses with fields of view exceeding 180 degrees. Traditional models like the pinhole camera model are insufficient due to the significant nonlinear distortions. Various projection models are used for fisheye cameras, including stereographic, equidistant, orthographic, and equisolid projections, each with specific advantages.

Kannala et al. (40) proposed a model using polynomial terms to account for large radial distortions. Rational function-based models, like the division model by Fitzgibbon (22), provide a closed-form inverse and are used for stereo reconstruction. The *Field of View (FOV)* model by Devernay and Faugeras (18) uses a single parameter to describe fisheye lenses and also offers a closed-form inverse.

2.4 Generalized Camera Models

Noncentral camera models, which consider the shift of the optical center, are important for accurately modeling wide-angle lenses. Gennery (24) proposed a model for fisheye lenses with a pupil shift function.

Generalized camera models do not rely on physical properties but treat the optical system as a black-box, providing flexibility at the cost of complex calibration. The *Two-Plane Model* (50) uses interpolation functions to map 2D pixels to 3D coordinates on calibration planes, allowing for both global and local distortions.

Discrete camera models, like the *raxel* model (28), treat each pixel individually, incorporating geometric, radiometric, and optical components. These models require dense observations for calibration.

2.5 Modeling Distortions from Refractive Media

Cameras observing scenes through refractive media, such as underwater or behind windshields, face complex distortions. These can be addressed by explicitly modeling the refracted ray path or using generalized camera models that treat the refractive media as part of the imaging system.

Agrawal et al. (2) studied systems with multiple layers of flat refractive surfaces, describing them as *axial cameras* and providing analytical forward projection equations. Yoon et al. (86) introduced a parametric model for depth estimation using stereo cameras behind transparent objects.

Generalized models like the two-plane model (80) and local models using B-splines (5) have been proposed for automotive use-cases. Kim et al. (14, 42) introduced methods for calibrating cameras with complex refractive distortions, using radial basis functions.

2.6 Conclusions

This chapter presented the theoretical foundations of camera models and calibration. It covered the classic pinhole camera model, geometric distortions, and calibration techniques. For complex systems like fisheye cameras, specific models are necessary to account for large distortions. Generalized and noncentral models offer flexible solutions for various optical systems. Finally, methods for modeling distortions from refractive media were discussed, highlighting both explicit modeling and generalized camera models.

Chapter 3

Machine Learning and Deep Learning

Machine learning, a branch of artificial intelligence, enables computers to learn from data by identifying statistical patterns. It involves training models for tasks like classification, regression, and clustering, using optimization algorithms to minimize prediction errors. Classical machine learning uses models with limited parameters, while deep learning trains neural networks with millions of parameters on large datasets. This Chapter presents the fundamental concepts of both machine learning and deep learning used in our work.

3.1 Radial Basis Functions

Radial Basis Functions (RBFs) (3, 11) are used to interpolate or approximate scattered data. Given a set of data points, the goal is to find a smooth function that satisfies the interpolation condition at these points. RBFs define the interpolation function as the weighted sum of basis functions centered at the data points. The influence of a data point to the interpolated value depends on the distance from the point. Common kernel functions include Gaussian, thin plate spline, multiquadric, and inverse multiquadric functions.

For interpolation, the weights are set to satisfy the interpolation condition, which can be expressed in a matrix form and solved as a linear system. RBFs can also approximate data points by minimizing a cost function, which includes a least squares error term and a regularization term based on the RBF weights. When combined with a polynomial component, the interpolation function gains global representational power. In this case the cost function includes an orthogonality constraint to ensure that global properties are characterized by the polynomial component.

3.2 Gradient Based Optimization of Parametric Models

Optimization techniques are crucial in machine learning for finding the correct model parameters by minimizing a loss function. In deep learning, training involves updating neural network parameters using the gradient of a task-specific loss function, typically with the *Stochastic Gradient Descent* (SGD) algorithm. First-order optimization methods, like SGD, use the first derivative of the loss function and are suitable for deep learning due to the large number of model parameters. Variants of SGD, such as SGD with momentum and Adam optimizer (43), address issues related to initial learning rate settings and provide adaptive learning rates.

Second-order optimization techniques consider the curvature of the objective function via the Hessian matrix, offering faster convergence for problems with fewer parameters. The Newton-Raphson method and its simplified form for nonlinear least squares problems, the Gauss-Newton algorithm, are commonly used. Quasi-Newton methods, such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm and its limited-memory variant L-BFGS, approximate the Hessian matrix and are suitable for problems where computing the full Hessian is impractical.

3.3 Convolutional Architectures

Convolutional Neural Networks (CNNs) are state-of-the-art architectures for computer vision tasks. They process images through layers of convolutions, down- or upsampling, and activation functions. CNNs are efficient due to their ability to model local interactions and translation invariance. Key layers include convolutional layers, activation functions, normalization layers, pooling layers, upsampling layers, and fully connected layers.

Residual networks (ResNets) (33, 34), known for their residual blocks and skip connections, address the vanishing gradient problem, allowing deeper networks to be trained effectively. Different ResNet models, such as ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152, vary in depth and complexity but share a common structure, with an initial convolutional layer followed by stages of residual blocks.

3.4 Computer Vision Tasks

Neural networks have been successfully applied to various computer vision tasks, including semantic segmentation, optical flow estimation and camera calibration. Semantic segmentation classifies each pixel in an image into semantic classes, essential for automotive perception systems. Architectures such as *U-Net* (63) and *DeepLabv3* (13) use encoder-decoder structures and dilated convolutions respectively to maintain high-resolution details while capturing global context.

Optical flow describes the motion of pixels between images and can be sparse or dense. *FlowNet* (20) and its successor *FlowNet 2.0* (37) were among the first deep learning approaches for dense optical flow estimation, using encoder-decoder architectures and cascaded processing to handle large and small displacements.

Deep learning methods for camera calibration can estimate intrinsic and extrinsic parameters from single images. Early methods like *DeepFocal* (82) assumed a simplified pinhole camera model and predicted a limited set of parameters. Later approaches such as *FishEyeRectNet* (85) introduced various improvements, including the use of auxiliary tasks and image reconstruction based loss functions.

3.5 Image Reconstruction Based Training

When direct supervision of camera calibration parameters is challenging, image reconstruction-based loss functions can be used. *Spatial Transformer Networks* (STN) (39) enable differentiable image warping, making it possible to train networks end-to-end using reconstruction losses. STNs consist of a localization network, a grid generator, and a differentiable sampler.

In self-supervised training, such as in *SfmLearner* (90), STNs are used to warp source images to match target views, using predicted depth and ego-motion. The reconstruction loss is based on the difference between the real and generated target views. Combining L1 loss with Structural Similarity Index Measure (SSIM) or multi-scale SSIM (MS-SSIM) (81) provides a more robust learning signal by accounting for human perception of image quality.

3.6 Conclusions

This Chapter covered the essential concepts of machine learning and deep learning relevant to our work. Radial Basis Functions (RBFs) are useful for function interpolation and approximation, and gradient-based optimization techniques are crucial for training machine learning models. Convolutional Neural Networks (CNNs) remain the dominant architecture for computer vision tasks. Neural networks are effective for self-calibration and can be trained using image reconstruction-based loss functions with the help of Spatial Transformer Networks.

Chapter 4

Explicit Modeling of the Refractive Surface

This Chapter presents our publication titled *Distortion Estimation Through Explicit Modeling of the Refractive Surface* (59).

The primary focus of this work is on camera systems used in automotive applications, where cameras are mounted behind windshields or other protective covers. The presence of these refractive materials complicates the geometric modeling of the camera system due to light refraction, which leads to image distortions as light enters or leaves a denser medium, causing directional changes.

When light passes through a refractive material, it changes direction, resulting in image distortions. This phenomenon makes it difficult to use global, central camera models because the refractions make the optical system challenging to characterize. The literature usually addresses this problem using one of two approaches: either by explicitly modeling the refractions or by applying generalized camera models. For automotive applications, the global shape of the windshield is generally known through computer-aided design (CAD) models, though local irregularities can exist. Assuming the global shape is known, we propose a camera model that explicitly accounts for light refractions and includes a local model to handle surface irregularities. This method can be used for the initial calibration of cameras behind transparent objects.

We construct the *forward model* $f_\theta(\boldsymbol{p}) : \Omega \to \mathbb{R}^3$, which maps a pixel from the image to a point in the scene, considering the camera parameters, refractive media, and scene characteristics, collectively denoted as θ. This function is implemented as a raycasting algorithm, allowing image generation given a set of parameters. By inverting the model, we fit the parameters of the refractive media to observed displaced points. We build a Radial Basis Function (RBF) network (6) model of the refractive media's thickness and use Maximum Likelihood (ML) estimation to infer the optimal parameters that caused the distortions.

The contributions of this Chapter are:

- Introduction of a *local model* for uneven refractive surfaces, contrasting with the global models prevalent in literature.
- Focus on scenarios where the global shape of the transparent object is known, presenting an *explicit refraction model.*
- Demonstration that such models significantly reduce calibration error.
- An *analysis* of windshield distortions, providing insights into these optical systems.

4.1 Refractive Surface Model

The refractive media is modeled as a thick cone slice, which is based on the actual shape of the glass object used in our experiments. The inner and outer cones have the same aperture, and the centers are such that the media's thickness is constant. The cone is positioned with its main axis parallel to the camera's y axis and shrinks in the positive y direction.

To account for the uneven surface, a parametric surface in the radial direction is added to the cone. This radial offset is defined using a Radial Basis Function (RBF) network, which is chosen for its universal function approximation capabilities (6). The RBF centers are placed on a regular grid over the input region, and the weights $w_{ij} \in \mathbb{R}$ are tuned to model complex surfaces. The radial offset at a given cone point is the output of the RBF network with Gaussian kernels:

$$\Phi(\boldsymbol{s}') = \sum_{i,j=1}^{N} w_{ij}\phi(\|\boldsymbol{s}_{ij} - \boldsymbol{s}'\|), \quad \text{where } \phi(r) = \exp\left(-\frac{r^2}{2\beta}\right) \tag{4.1}$$

To compute the Cartesian coordinates for a point on the cone, parameterized by a height s_1 and an angle s_2, the RBF offset is first computed and then added to the radius. The surface normals of the outer cone are calculated as the cross product of the partial derivatives of the Cartesian coordinates with respect to the parameters. This cross product depends on the RBF weights, which are used as model parameters during minimization. Changing the RBF weights alters the surface normals, thus changing the direction of the refracted light rays and, consequently, the distortion vectors.

4.2 Raycasting Model

The raycasting model describes the process of associating a pixel from the image with a 3D point on an object, such as a checkerboard pattern. In a distortion-free setup, this can be achieved using the pinhole camera model, which uses a perspective projection and a set of linear operations to describe this relationship. However, in the presence of a refractive surface, this simple geometric description does not hold, and additional steps are needed.

Each ray starts at the camera center, assuming that the intrinsic camera parameters – the focal lengths and the principal point, i.e., the intrinsic camera matrix – are known. All 3D points are expressed in the camera coordinate system. Using the camera intrinsics, any pixel coordinate $\boldsymbol{p}$ can be converted to metric coordinates, which after normalization correspond to the direction vector $\boldsymbol{r_{cam}}$ of the light ray passing through the selected pixel.

The light ray first hits the inner side of the refractive surface, intersecting the cone at $\boldsymbol{x}_i$ and encountering the normal $\boldsymbol{n}_i$. The direction $\boldsymbol{r}_m$ of the refracted light ray inside the media is computed using Snell's law. Knowing the geometry of the refractive body and the new direction of the refracted ray, the location $\boldsymbol{x}_o$ where the ray hits the outer surface is identified, and the second refraction is computed. The direction of the outgoing light ray, $\boldsymbol{r}_o$, is then determined. This second refraction is modulated by the direction of the normal $\boldsymbol{n}_o$, parameterized by the RBF network.

Finally, the outgoing light ray intersects the calibration target, which is defined through a 3D rotation and translation of the board center relative to the camera coordinate system. The intersection point $\boldsymbol{x}_t$ is computed as the intersection of a line and a plane. A local 2D coordinate system on the object plane is defined for easier handling, with its origin at the board center and axes corresponding to the horizontal and vertical directions of the checkerboard grid. The local coordinates of a 3D point $\boldsymbol{x}_t$ are denoted as $\boldsymbol{x}_{cb}$.

4.3 Optimization of the Surface Parameters

The estimation of image distortions involves finding the surface parameters that generated a set of calibration images, using a checkerboard pattern as the calibration target. Gradient descent minimization is employed to determine the optimal RBF weights, while other parameters, including camera intrinsics, the sizes of the inner and outer cones, and the calibration pattern pose and size, are assumed to be known.

For each calibration image, the pixel coordinates and ordering of the checkerboard pattern corners are identified. The corresponding local coordinates of the detected corners on the object plane are given by their distance from the board center. The raycasting function, parameterized by the RBF weights, maps an input pixel to the local coordinates of the corresponding world point on the target checkerboard pattern. The loss function is defined as the $\mathcal{L}_2$ loss between the estimated local coordinates of a corner and the ground truth local coordinates.

4.4 Calibration Results of the Refractive Surface Model

As no public benchmark dataset is available for our problem, we created our own experimental setup. We evaluate the algorithm on two datasets: a noise-free synthetic dataset and a real experimental setup. In the synthetic case, we show that our algorithm is capable of finding the optimal parameters that generated a given image, even with large irregularities on the outer surface. In the second case, we present an experimental setup and demonstrate that the algorithm can reduce reconstruction errors in real-world scenarios.

In the synthetic dataset, we apply the forward image generation model to render synthetic images. The parameters of the camera, refractive surface, and checkerboard pattern are set to similar values as in the real-world experiment. Using an RBF grid of size 4×4, we sample the weights from a Gaussian distribution and generate synthetic images using random positions for the calibration target. The optimization is performed, and the final error is stored as the root mean squared error (RMSE) between the predicted and ground truth checkerboard corners.

In the real dataset, a Raspberry Pi Camera Module v2 captures the checkerboard images. The camera has a 3.68×2.76 mm sensor and registers images at a 3280×2464 pixel resolution. The camera is calibrated using Zhang's method (88), resulting in a 2558.36 pixel focal length and a principal point at $(1666.03, 1273.65)$. After calibration, a cone-shaped glass object is placed in front of the camera. Images are captured, and the optimization process is performed on different sets of images.

The results show significant error reduction in the back-projected pixels, demonstrating the model's effectiveness. The errors are evaluated in terms of RMSE between the ground truth and predicted positions of the checkerboard corners. The initial error, considering no distortion model, is compared to the final error obtained using the optimized Cone + RBF surface model.

The method improves the 3D distances, resulting in a more accurate generative model of the image formation process. Analysis of distortions shows a linear dependence on inverse depth, suggesting potential for simpler camera models.

Most distortion estimation methods directly model pixel displacements on the image plane, defining a single, fixed distortion map for a given camera. In contrast, our model estimates the distortion map by explicitly modeling the refractive material using a raycasting algorithm. This approach provides a unified and consistent generative model for directional distortions. The physical model introduces a depth-dependent component in the distortion map, requiring the distance of a 3D point to determine the image distortion.

To compute the image distortion vector, we start from a distorted pixel on the image. Using the distorted pixel $\boldsymbol{p}_d$, the raycasting algorithm computes the 3D coordinate $\boldsymbol{x}_t$ of the object point at a given distance. The undistorted pixel coordinates $\boldsymbol{p}_u$ are then computed using the pinhole camera model. The distortion vector $\Delta\boldsymbol{p}$ is given by the difference between the distorted and undistorted coordinates.

The distortion vector field for the real dataset is visualized, showing the dependence of distortions on pixel depth. The middle pixels exhibit little distortion due to near-orthogonal rays to the refractive surface, while lateral pixels show larger distortions due to significant refractions. The linear dependence of distortion on inverse depth is confirmed, indicating that closer object points cause larger distortions.

4.5 Conclusions

In this Chapter, we presented a camera model that accounts for the uneven surface of a transparent object using an RBF network. By explicitly modeling light refractions at the surfaces of the object and employing Snell's law, we developed a comprehensive model that considers both the global shape and local properties of the object. The model parameters were estimated using calibration images with checkerboard patterns.

Our experiments demonstrated that the proposed model significantly reduces the error of back-projected pixels. The model effectively mitigates the bias introduced by the horizontal curvature of the cone-shaped object, resulting in errors that resemble an isotropic Gaussian distribution. The linear dependence of distortions on inverse depth offers opportunities for developing simpler camera models that can be more easily integrated into computer vision systems.

A limitation of our method is the fixed base shape, which in this study was a cone. Although this shape closely approximated the actual object used in the experiments, parameterizing the base shape could extend the model's applicability to other geometries. However, increasing the number of free parameters also introduces further modeling difficulties, making optimization more challenging.

In summary, this work provides a detailed exploration of explicit modeling of refractive surfaces, contributing to the field of camera calibration for systems operating behind transparent materials. The approach has the potential to improve the accuracy of vision systems in automotive and other applications where refractive distortions are prevalent.

Chapter 5

Ellipsoid Object Model

This Chapter presents our publication titled *An Ellipsoid Object Model of the Refraction Surface* (58).

Geometric distortions can arise when the camera is placed behind a protective cover, such as a windshield of a car. These distortions are influenced by global properties of the object (e.g., position relative to the camera, curvature of the surface, and thickness of the material) as well as irregularities of the surface, resulting in local distortions. In Ch. 4, we modeled these irregularities using an RBF-network while assuming that the global properties of the refractive object are known. This assumption limits the applicability of the camera models.

In this work, we address the global properties of the refractive object using a similar methodology as in Ch. 4. We model the surface of the refractive media as an *ellipsoid*, which can approximate a variety of objects in the camera's view. The model is designed to be composable with the RBF-network model of the local surface.

The contributions of this Chapter are:

- A *global model* based on an *ellipsoid shape* is proposed and the raycasting algorithm is defined.
- The chapter addresses the issue of arising *symmetries* in the distortion estimation process, where a regularization term is included to guide the minimization.
- The method is *evaluated* on a synthetic dataset.

5.1 Ellipsoid Model

We model the refractive object as the space between two ellipsoids sharing the same center position and orientation. The inner ellipsoid has semi-axes a, b, c, and the outer ellipsoid's semi-axes are defined by adding a small thickness t to each semi-axis of the inner ellipsoid. The parameters of the ellipsoid object model are the center position, orientation, semi-axes, and thickness.

To simplify further computation, the ellipsoid is viewed as an affine image of a unit sphere centered at the origin. The transformation involves a 3×3 matrix and a translation vector. Using this affine mapping, necessary operations, including intersection with a ray and surface normal evaluations, can be reduced to operations on the unit sphere.

The goal of the camera model is to associate pixels with light rays from the outside world. In a distortion-free setup, the light ray from the camera center goes through the image pixel as per the pinhole camera model. In our model, the direction of the original light ray changes when it enters or leaves the refractive object, computed using Snell's law of refraction. This change is a function of the incident ray, the surface normal at the intersection point, and the relative refractive index of the materials. The raycasting process is fully differentiable, allowing gradient-based optimization of the ellipsoid model parameters. The method is implemented in PyTorch to utilize automatic differentiation for optimization.

5.2 Symmetries of the Ellipsoid Object Model

The ellipsoid model overparameterizes the image distortions, resulting in symmetries in the physical model. The observed distortions are invariant to certain transformations of the object. Consequently, the full parameter set cannot be recovered without prior knowledge about the object. While distortion estimation may be sufficient in some cases, reconstructing an approximate 3D model of the object can be desirable. Identifying and handling these symmetries using regularization techniques is crucial.

An intuitive example of these symmetries is a 2D case where the refractive object is a thick circle. We consider two variables: the relative distance of the circle center to the camera center and the radius of the circle. This analysis also applies to the center position of the ellipsoid and the length of the semi-axes in 3D. By comparing the distortion error relative to a reference setup for different parameters, we observe that low distortion errors can be achieved by adjusting both parameters appropriately.

5.3 Optimization of the Model Parameters

We use a standard static camera calibration setup to estimate the model parameters. A planar checkerboard pattern serves as the target object, with known dimensions of the squares. Using model inversion based on images of the checkerboard patterns, the parameters of the ellipsoid model can be recovered through gradient-based minimization.

The loss function comprises a reconstruction and a regularization term. The reconstruction error is the mean squared error between the estimated and ground truth corner coordinates of the checkerboard. The regularization term includes prior knowledge as a constraint on the distance between the camera center and the point where the principal axis intersects the inner surface of the ellipsoid. The total loss function is minimized using the L-BFGS optimization method, chosen for its efficiency and compatibility with automatic differentiation.

5.4 Conclusions

This Chapter addressed the scenario where a camera is placed behind a transparent object with an unknown global shape. We modeled the object as an ellipsoid and used machine learning techniques to estimate the model parameters. The model is compatible with the RBF surface model from Chapter 4, allowing simultaneous modeling of both global and local properties of the refractive surface. We analyzed potential failure cases, proposed a regularization to address them, and tested the method on a synthetic dataset. The method achieved a close approximation of the object's surface in the camera's view.

Chapter 6

Single View Distortion Estimation

This Chapter presents the publication *Single View Distortion Correction using Semantic Guidance* (48).

Previous Chapters discussed an *initial (offline) calibration* method addressing distortions caused by transparent objects like car windshields. The camera model was split into local distortions and a global shape optimization, as detailed in Chapters 4 and 5 respectively. In automotive settings, cameras need to maintain functionality over time despite vibrations and temperature changes, necessitating an online calibration component based on *self-calibration*, which functions without specific calibration targets.

We propose a self-calibration method using *deep learning* with a distortion model based on *thin plate splines* (TPS). The neural network predicts the parameters of the distortion model, including control points for local components and polynomial coefficients for global components. Experiments demonstrate the model's capability to estimate complex distortions, making it suitable for practical applications in autonomous driving systems and other domains where camera systems are exposed to varying conditions over long periods.

The contributions presented in this Chapter are:

- The Chapter presents *scalable deep learning approach* that can correct distortions. While the deep learning methods presented in the literature usually predict only a small number of parameters for a global camera model, the proposed distortion model is applicable to arbitrarily complex distortions (including local ones).

- Similar to other methods in the literature, *two datasets* comprising of real-world (KITTI odometry (23)) and synthesized (Carla (21)) images and corresponding semantic segmentation are constructed, on which parametric distortions sampled from a distribution derived from real-world measurements in the presence of different windshields are applied.
- Networks are trained in an end-to-end manner without using hard to obtain ground truth distortions as supervision, and instead recent advancements in differentiable image sampling are leveraged to formulate an *image reconstruction loss*.
- Results show, that *auxiliary tasks* (semantic segmentation and optical flow) improve the quality of the predictions.

6.1 Windshield Distorted Dataset

To validate our model's ability to undistort images, we constructed two datasets, *Distorted Carla (DC)* and *Distorted KITTI (DK)*, following established methodologies. These datasets were designed to test the model's performance in both synthetic and real-world scenarios, providing a comprehensive evaluation of its capabilities.

We use a proprietary dataset from 240 car windshields, where images were captured with and without the windshield and pixel-wise distortion is measured. We fit a high-order polynomial function to these measurements, and new distortions were generated by sampling and perturbing polynomial coefficients. This method maintained realistic variability across images, ensuring that the synthetic distortions closely resembled those found in real-world conditions.

Distorted Carla (DC) comprises 10,000 synthetic images and semantic labels, generated using the Carla simulator (21). The images, captured at 5 frames per second in a preset environment, were split into 8000 training and 2000 validation samples. The dataset included RGB images and semantic segmentation maps, which were used to provide additional context for the distortion correction model. The Carla simulator's flexibility allowed us to create a diverse set of conditions, including different times of day, weather conditions, and dynamic elements like vehicles and pedestrians.

For real-world data, we used the KITTI odometry dataset (23). This dataset includes sequences captured from a moving vehicle in various urban environments. We downscaled the images and applied synthetic distortions, creating the *Distorted KITTI (DK)* dataset with 10684 training and 4539 validation images.

Optical flow was incorporated as an auxiliary task, using image triplets as input to a flow estimation network. By including optical flow, we provided the model with temporal information, which is particularly useful for understanding dynamic scenes and improving the accuracy of distortion correction.

6.2 Distortion Model

Bookstein (7) showed that a pair of *thin plate splines* (TPS) could model 2D deformations. We modeled geometric distortions using TPS pairs, which are particularly effective for representing smooth and continuous deformations. This choice of model allows us to handle both global distortions that affect the entire image and local distortions that are confined to specific regions.

The transformed coordinates $\boldsymbol{f}_{tps}(\boldsymbol{G}_i)$ at image coordinate $\boldsymbol{G}_i = [x_i, y_i]^\top$ assuming n control points are defined as:

$$\boldsymbol{f}_{tps}(\boldsymbol{G}_i) = \boldsymbol{A} \begin{bmatrix} \boldsymbol{G}_i \\ 1 \end{bmatrix} + \sum_{k=1}^{n} \phi(\|\boldsymbol{p}'_k - \boldsymbol{G}_i\|) \cdot \boldsymbol{w}_k, \quad \text{where } \phi(r) = r^2 \log r \tag{6.1}$$

We used 16 control points, evenly distributed on a 4 × 4 grid. The affine transformation $\boldsymbol{A}$ modeled global distortions, while the radial basis kernel $\phi(r)$ and warping coefficient matrix $\boldsymbol{W}$ captured local deformations. This combination of global and local components allows our model to handle a wide range of distortion types, from simple linear transformations to complex nonlinear warping.

The TPS model's flexibility makes it ideal for applications where the distortions are not uniform across the image. For example, in automotive settings, the distortions caused by a windshield can vary significantly depending on its shape and position relative to the camera. By using TPS, we can accurately model these variations and correct them in a seamless manner.

6.3 Proposed Architecture

Our end-to-end architecture inputs a single distorted image $\boldsymbol{I}$ and outputs the undistorted image $\boldsymbol{I}'$ and optionally its semantic labels. It follows an encoder-decoder structure with auxiliary tasks that provide additional context for the distortion correction process.

A ResNet-18 (33) pretrained on ImageNet (65) served as the core network. This network extracts low-level features from the input image, which are then used by the decoder to estimate and correct distortions. When using optical flow, the network processed concatenated image triplets. This modification allowed the core network to handle multiple frames simultaneously, providing a richer set of features for distortion correction.

The semantic segmentation network upsampled feature maps and concatenated them with the input image for distortion prediction. By including semantic segmentation, the model could leverage high-level information about the scene, such as the locations of objects and boundaries. This information helps the model make more informed decisions about how to correct distortions, particularly in areas with complex structures.

The distortion correction network followed the Spatial Transformer Network (39) architecture. It localized control points, calculated the sampling grid, and sampled the distorted image to create the corrected image. The Spatial Transformer Network's differentiable nature allows the entire process to be trained end-to-end, ensuring that all components work together seamlessly.

Incorporating auxiliary tasks like semantic segmentation and optical flow not only improves the model's performance but also provides additional outputs that can be useful in other applications. For instance, the semantic segmentation maps can be used for object detection and scene understanding, while optical flow can provide information about motion and dynamics in the scene.

6.4 Model Training

The model parameters were initialized using "He" uniform initialization (32). We used various loss functions, including an image reconstruction loss based on MS-SSIM (81) and a grid loss formulated as the mean squared error between the estimated and ground truth sampling grids. These loss functions ensure that the model learns to produce accurate undistorted images and align the sampling grid closely with the ground truth.

The final loss function was a weighted sum of image reconstruction, grid, and semantic segmentation losses, optimized using Adam (43) with a batch size of 8 and specific learning rates for different network components. This combination of loss functions balances the need for accurate image reconstruction with the requirement to align the sampling grid and produce correct semantic labels.

During training, we experimented with different settings to find the optimal configuration. We found that using both reconstruction and grid losses provided the best results, as each loss function complements the other. The reconstruction loss ensures that the overall image quality is preserved, while the grid loss focuses on aligning the distortions precisely.

The inclusion of semantic segmentation and optical flow as auxiliary tasks further improved the model's performance. These tasks provide additional information that helps the model better understand the scene, leading to more accurate distortion correction. By training the model end-to-end, we ensured that all components worked together seamlessly, resulting in a robust and reliable distortion correction system.

6.5 Training Results

We trained the networks on the Distorted Carla Train set and tested them on both the Distorted Carla Test and Distorted KITTI Test sets. Fine-tuning on the Distorted KITTI Train set further improved results, demonstrating the model's ability to adapt to different datasets and environments.

Quantitative evaluation used the residual distortion norm, a metric that measures the average distance between the distorted and undistorted grid points. Our method significantly reduced distortion on both datasets, with the best performance achieved using a combination of reconstruction and grid losses with optical flow as an auxiliary task.

Without fine-tuning, the network trained on Distorted Carla transferred well to Distorted KITTI, except when optical flow was used as an auxiliary task. Fine-tuning improved results in all configurations, indicating that the model can adapt to different types of data and environments with additional training.

Using auxiliary tasks improved performance, with optical flow yielding the best results. This improvement is likely due to the additional temporal information provided by optical flow, which helps the model understand motion and dynamics in the scene. Semantic segmentation also improved performance, providing high-level contextual information that helps the model make more informed decisions about how to correct distortions.

Both reconstruction and grid losses were effective, indicating that ground truth distortions were not necessary for training. This finding is significant because it means that the model can be trained on a wide range of datasets, even those without ground truth distortions. By using pairs of distorted and undistorted images, the model can learn to correct distortions effectively, making it versatile and widely applicable.

6.6 Conclusions

This work presents a deep learning method for correcting complex distortions, useful for self-calibration without specific calibration targets. Our distortion model, based on thin plate splines, handles both global and local distortions, making it suitable for a wide range of applications, including autonomous driving and other domains where cameras are used in dynamic environments.

We generated two datasets using real-world distortion distributions. Our neural network effectively reduced residual distortions, with auxiliary tasks enhancing performance. The experiments demonstrated that it is possible to train the network without access to ground truth distortions, allowing extension to datasets with parallel recordings of distorted and undistorted images. This capability makes our method highly versatile and applicable to a wide range of scenarios where traditional calibration methods may fall short.

Overall, our approach represents a significant advancement in the field of camera calibration and distortion correction. By leveraging deep learning and auxiliary tasks, we have developed a robust and scalable solution that can handle complex distortions in real-world settings. Future work could explore additional auxiliary tasks and further refine the model to improve performance and extend its applicability to new domains and challenges.

Chapter 7

Conclusions

When cameras are mounted behind transparent objects, due to light refraction, images will be deformed, distorted. This issue affects advanced driver assistance systems, where a camera used for sensing the world around the car is often mounted behind the windshield. Geometric computer vision algorithms require a precise camera model, which is able to map 3D world coordinates to 2D pixels. In the presence of a windshield, this camera model also has to take into the account the distortions. These are usually large and highly nonlinear, having both global and local components. In our work we studied the problem of camera calibration in the presence of transparent, refractive surfaces.

We proposed an initial calibration method, where we construct a precise model of distortions caused by transparent objects in the optical path (Ch. 4 and Ch. 5). We chose a physics based approach: instead of abstracting away the components of the camera system, we explicitly model them, including the refractive surfaces. We trace the path of individual light rays from the center of the camera to rays in the 3D world, taking into account changes in direction at the boundaries of transparent materials. We base this decision on the fact, that in our use-cases we have information about the components, which we can incorporate into our physics based models.

First, we modeled the uneven surface of a transparent object with a global shape. This can cover use-cases, where physical elements such as windshields can be manufactured only up to certain tolerances, and can have irregularities. We modeled the surface using radial basis functions, which we can use to estimate the distortions locally. The model parameters – the weights (amplitudes) – of the radial basis functions were estimated using optimization techniques, based on images of checkerboard calibration targets. Our model significantly improved the errors for the checkerboard corners, both on synthetic and real datasets. We also provided an analysis of the observed distortions, where we identified that these have a direct linear relationship with the inverse depth of the pixels.

In this method we formulated the model by defining the back-projection function. As this function involves the tracing of the rays through multiple surfaces, it has two main limitations. Due to the complexity of the model, inverting it to formulate the forward projection can only be done using iterative methods. Also, the high complexity can be a drawback for embedded use-cases, where the camera model needs to be evaluated frequently, and the large number of computations can increase runtime. Future work should focus on finding solution to this problem, by searching for simpler models, which can still capture the main properties of these distortions, but at the same time can be integrated into real systems. The analysis of the distortions can provide a good starting point in this direction.

After proposing a model for the local uneven surface of the transparent object, we turned our attention towards the global shape for use-cases, where this is unknown. We proposed to model the global shape as an ellipsoid, which is general enough to approximate the shape of different objects in the region of interest of the camera. This model is directly compatible with the surface model from our previous work. We found out, that minimizing this problem is difficult, because it is underconstrained: different parameters (global shapes) can result in very similar image distortions. A regularization term was added, which was able to guide the minimization to the right direction.

The main limitations in our proposed method are related to our testing methodology, as this was done only on synthetic data. Extending this to real data should be the main focus of future work. After proving it on real data, the two proposed frameworks – an ellipsoid global model with a radial basis function based local model – could be an interesting future research problem.

Finally, in Ch. 6 we proposed a solution for self-calibration based on deep learning. We used a dataset of real windshield distortion measurements, and constructed a synthetic and a real world dataset by sampling distortions around the measured ones. Our convolutional neural network based architecture can predict distortions based on single images, or based on a sequence of 3 images without constraints on the environment of the car. The architecture also includes semantic segmentation or optical flow as auxiliary tasks, which we show that can significantly improve our results. We use once again a distortion model, which includes both a global and a local component: we use thin plate splines with an additional affine component to model image deformations. Two loss functions were tested, one based on ground truth distortions, and a second one based on image reconstruction. The best results were achieved using the combination of the two, but training only based on image reconstruction provided competitive results.

The main limitation of this method is our dataset: the distortions although are based on real windshield distortion measurements, were synthetically applied to the images. The viability of only using image reconstruction loss opens up however possibility of future work. Instead of using known distortions, a parallel recording setup could be set up, with one windshield distorted and one undistorted camera. The reconstruction based loss function can be extended, where assuming a correct distortion estimate, we synthesize the undistorted image based on the distorted one. Proving this method without synthetically generated data could open up the possibility of real applications.

References

[1] S. Agarwal, K. Mierle, and Others. Ceres solver. `http://ceres-solver.org`. Accessed: 2024-06-10.

[2] A. Agrawal, S. Ramalingam, Y. Taguchi, and V. Chari. A theory of multi-layer flat refractive geometry. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3346–3353. IEEE, 2012. 9

[3] K. Anjyo, J. P. Lewis, and F. Pighin. Scattered data interpolation for computer graphics. In *ACM SIGGRAPH 2014 Courses*, pages 1–69. Association for Computing Machinery, 2014. 10

[4] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[5] J. Beck and C. Stiller. Generalized b-spline camera model. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 2137–2142. IEEE, 2018. 9

[6] C. Bishop. *Pattern Recognition and Machine Learning.* Springer-Verlag, New York, NY, USA, 2006. 15

[7] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on pattern analysis and machine intelligence*, 11(6):567–585, 1989. 25

[8] C. Bräuer-Burchardt and K. Voss. Automatic lens distortion calibration using single views. In *Mustererkennung 2000*, pages 187–194. Springer Berlin Heidelberg, 2000.

[9] C. Brauer-Burchardt and K. Voss. A new algorithm to correct fish-eye-and strong wide-angle-lens-distortion from single images. In *Proceedings 2001 International Conference on Image Processing*, volume 1, pages 225–228. IEEE, 2001.

[10] D. C. Brown. Decentering distortion of lenses. *Photogrammetric Engineering and Remote Sensing*, 32(3):444–462, 1966.

[11] M. D. Buhmann. Radial basis functions. *Acta numerica*, 9:1–38, 2000. 10

[12] M. Cassidy, J. Mélou, Y. Quéau, F. Lauze, and J.-D. Durou. Refractive multi-view stereo. In *2020 International Conference on 3D Vision (3DV)*, pages 384–393. IEEE, 2020.

[13] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 12

[14] T. Choi, S. Yoon, J. Kim, and S. Sull. Noniterative generalized camera model for near-central camera system. *Sensors*, 23(11):5294, 2023. 9

[15] D. Claus and A. W. Fitzgibbon. A rational function lens distortion model for general cameras. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 213–219. IEEE, 2005. 8

[16] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223. IEEE, 2016.

[17] S. Derrien and K. Konolige. Approximating a single viewpoint in panoramic imaging devices. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings*, volume 4, pages 3931–3938. IEEE, 2000.

[18] F. Devernay and O. Faugeras. Straight lines have to be straight. *Machine vision and applications*, 13(1):14–24, 2001. 8

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021*. OpenReview.net, 2021.

[20] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In

Proceedings of the IEEE International Conference on Computer Vision, pages 2758–2766. IEEE, 2015. 12

[21] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16. PMLR, 2017. 24

[22] A. W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–125–I–132. IEEE, 2001. 8

[23] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012. 24

[24] D. B. Gennery. Generalized camera calibration including fish-eye lenses. *International Journal of Computer Vision*, 68:239–266, 2006. 8

[25] C. Godard, O. Mac Aodha, and G. J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 270–279. IEEE, 2017.

[26] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3828–3838. IEEE, 2019.

[27] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning.* MIT Press, 2016. `http://www.deeplearningbook.org`. 6

[28] M. D. Grossberg and S. K. Nayar. A general imaging model and a method for finding its parameters. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 108–115. IEEE, 2001. 9

[29] M. D. Grossberg and S. K. Nayar. The raxel imaging model and ray-based calibration. *International Journal of Computer Vision*, 61(2):119–137, 2005.

[30] A. F. Habib, M. Morgan, and Y.-R. Lee. Bundle adjustment with self–calibration using straight lines. *The Photogrammetric Record*, 17(100):635–650, 2002.

[31] T. Hanning. *High precision camera calibration.* Springer, 2011. 5

[32] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034. IEEE, 2015. 26

[33] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. IEEE, 2016. 11, 25

[34] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 630–645. Springer, 2016. 11

[35] J. Heikkila. Geometric camera calibration using circular control points. *IEEE Transactions on pattern analysis and machine intelligence*, 22(10):1066–1077, 2000.

[36] T. J. Herbert. Calibration of fisheye lenses by inversion of area projections. *Applied optics*, 25(12):1875–1876, 1986.

[37] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470, 2017. 12

[38] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32Nd International Conference on Machine Learning*, volume 37 of *ICML'15*, pages 448–456. JMLR.org, 2015.

[39] M. Jaderberg, K. Simonyan, A. Zisserman, and k. kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, volume 28, pages 2017–2025. Curran Associates, Inc., 2015. 12, 26

[40] J. Kannala and S. S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE transactions on pattern analysis and machine intelligence*, 28(8):1335–1340, 2006. 8

[41] J. Kannala, S. S. Brandt, and J. Heikkilä. Self-calibration of central cameras from point correspondences by minimizing angular error. In A. Ranchordas, H. J. Araújo, J. M. Pereira,

and J. Braz, editors, *Computer Vision and Computer Graphics. Theory and Applications*, pages 109–122. Springer Berlin Heidelberg, 2009.

[42] J. Kim, C. Kim, S. Yoon, T. Choi, and S. Sull. Rbf-based camera model based on a ray constraint to compensate for refraction error. *Sensors*, 23(20):8430, 2023. 9

[43] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 11, 26

[44] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[45] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[46] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[47] M. Lopez, R. Mari, P. Gargallo, Y. Kuang, J. Gonzalez-Jimenez, and G. Haro. Deep single image camera calibration with radial distortion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11817–11825. IEEE, 2019.

[48] S.-B. Lőrincz, S. Pável, and L. Csató. Single view distortion correction using semantic guidance. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6. IEEE, July 2019. 6, 23

[49] L. Ma, Y. Chen, and K. L. Moore. Rational radial distortion models of camera lenses with analytical solution for distortion correction. *International Journal of Information Acquisition*, 1(02):135–147, 2004.

[50] H. Martins, J. R. Birk, and R. B. Kelley. Camera models based on data from two calibration planes. *Computer Graphics and Image Processing*, 17(2):173–180, 1981. 8

[51] C. Mei and P. Rives. Single view point omnidirectional camera calibration from planar grids. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 3945–3950. IEEE, 2007.

[52] B. Micusik and T. Pajdla. Estimation of omnidirectional camera model from epipolar geometry. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I. IEEE, 2003.

[53] P. Miraldo and H. Araujo. Calibration of smooth camera models. *IEEE transactions on pattern analysis and machine intelligence*, 35(9):2091–2103, 2012.

[54] S. Morinaka, F. Sakaue, J. Sato, K. Ishimaru, and N. Kawasaki. 3d reconstruction under light ray distortion from parametric focal cameras. *Pattern Recognition Letters*, 124:91–99, 2019.

[55] A. Odena, V. Dumoulin, and C. Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016.

[56] E. Olson. AprilTag: A robust and flexible visual fiducial system. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3400–3407. IEEE, May 2011.

[57] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8024–8035, 2019.

[58] S. Pável. An ellipsoid object model of the refraction surface. In *Proceedings of the 11th International Conference on Applied Informatics (ICAI)*, volume 2650, pages 272–279. CEUR Workshop Proceedings, January 2020. 6, 20

[59] S. Pável, C. Sándor, and L. Csató. Distortion estimation through explicit modeling of the refractive surface. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Image Processing*, pages 17–28. Springer, September 2019. 6, 14

[60] L. Qin, Y. Hu, Y. Wei, Y. Zhou, and H. Wang. Approach for camera self-calibration based on five straight lines. In *2008 4th International Conference on Wireless Communications, Networking and Mobile Computing*, pages 1–4. IEEE, 2008.

[61] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár. Designing network design spaces. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10428–10436. IEEE, 2020.

[62] J. Rong, S. Huang, Z. Shang, and X. Ying. Radial lens distortion correction using convolutional neural networks trained with synthesized images. In S.-H. Lai, V. Lepetit, K. Nishino, and Y. Sato, editors, *Computer Vision – ACCV 2016*, pages 35–49, Cham, 2017. Springer International Publishing.

[63] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 12

[64] D. E. Rumelhart and J. L. McClelland. *Learning Internal Representations by Error Propagation*, pages 318–362. MIT Press, 1987.

[65] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 25

[66] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520. IEEE, 2018.

[67] M. Schönbein, T. Strauß, and A. Geiger. Calibrating and centering quasi-central catadioptric cameras. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4443–4450. IEEE, 2014.

[68] B. Shi, X. Wang, P. Lyu, C. Yao, and X. Bai. Robust scene text recognition with automatic rectification. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4168–4176. IEEE Computer Society, 2016.

[69] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *The 3rd International Conference on Learning Representations*, pages 1–15, 2015.

[70] P. Sturm and S. Ramalingam. A generic concept for camera calibration. In T. Pajdla and J. Matas, editors, *Computer Vision - ECCV 2004*, pages 1–13. Springer, Springer Berlin Heidelberg, 2004.

[71] P. Sturm, S. Ramalingam, J.-P. Tardif, S. Gasparini, J. Barreto, et al. Camera models and fundamental concepts used in geometric computer vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(1–2):1–183, 2011. 5

[72] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9. IEEE, 2015.

[73] R. Szeliski. *Computer vision: algorithms and applications.* Springer Science & Business Media, 2010. 5

[74] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019*, pages 6105–6114. PMLR, 2019.

[75] T. Thormählen and H. Broszio. Automatic line-based estimation of radial lens distortion. *Integrated Computer-Aided Engineering*, 12(2):177–190, 2005.

[76] J. Tischendorf, C. Trautwein, T. Aach, D. Truhn, T. Stehle, et al. Camera calibration for fish-eye lenses in endoscopywith an application to 3d reconstruction. In *2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1176–1179. IEEE, 2007.

[77] R. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, 1987.

[78] Y.-H. Tsai, X. Shen, Z. Lin, K. Sunkavalli, X. Lu, and M.-H. Yang. Deep image harmonization. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2799–2807, 2017.

[79] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[80] F. Verbiest, M. Proesmans, and L. Van Gool. Modeling the effects of windshield refraction for camera calibration. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 397–412. Springer, 2020. 9

[81] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. IEEE, 2003. 12, 26

[82] S. Workman, C. Greenwell, M. Zhai, R. Baltenberger, and N. Jacobs. Deepfocal: A method for direct focal length estimation. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 1369–1373. IEEE, 2015. 12

[83] Y. Wu and K. He. Group normalization. In *Computer Vision – ECCV 2018*, pages 3–19. Springer International Publishing, 2018.

[84] Y. Xiong and K. Turkowski. Creating image-based vr using a self-calibrating fisheye lens. In *Proceedings of IEEE computer society conference on computer vision and pattern recognition*, pages 237–243. IEEE, 1997.

[85] X. Yin, X. Wang, J. Yu, M. Zhang, P. Fua, and D. Tao. Fisheyerecnet: A multi-context collaborative deep network for fisheye image rectification. In *Computer Vision – ECCV 2018*, pages 475–490. Springer International Publishing, 2018. 12

[86] S. Yoon, T. Choi, and S. Sull. Depth estimation from stereo cameras through a curved transparent medium. *Pattern Recognition Letters*, 129:101–107, 2020. 9

[87] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola. *Dive into Deep Learning.* Cambridge University Press, 2023. `https://D2L.ai`. 6

[88] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22:1330–1334, 2000. 8, 17

[89] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127:302–321, 2019.

[90] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1851–1858. IEEE, 2017. 12