Single Image Camera Calibration with Lenticular Arrays for Augmented Reality

Ian Schillebeeckx
Washington University in St. Louis
1 Brookings Dr, St. Louis, MO 63130
ischillebeeckx@wustl.edu

Robert Pless
Washington University in St. Louis
1 Brookings Dr, St. Louis, MO 63130
pless@wustl.edu

Abstract

We consider the problem of camera pose estimation for a scenario where the camera may have continuous and unknown changes in its focal length. Understanding frame by frame changes in camera focal length is vital to accurately estimating camera pose and vital to accurately rendering virtual objects in a scene with the correct perspective. However, most approaches to camera calibration require geometric constraints from many frames or the observation of a 3D calibration object — both of which may not be feasible in augmented reality settings. This paper introduces a calibration object based on a flat lenticular array that creates a color coded light-field whose observed color changes depending on the angle from which it is viewed. We derive an approach to estimate the focal length of the camera and the relative pose of an object from a single image. We characterize the performance of camera calibration across various focal lengths and camera models, and we demonstrate the advantages of the focal length estimation in rendering a virtual object in a video with constant zooming.

1. Introduction

Camera calibration is often the first step for Computer Vision and Augmented Reality (AR) applications because it defines how the scene is projected onto an image by the camera. Calibration characterizes the photometric and geometric properties of the camera, that define, respectively, how the pixels of the camera report color and intensity of the scene, and where scene elements appear on the image.

Typically, camera calibration techniques are based on images of a recognizable object at different views. Constraints on the camera geometry are created by finding 3D points on the object and corresponding them to 2D image points. When it is necessary to calibrate the intrinsic camera parameters such as the camera focal length, it is often necessary to user either multiple images of a planar object, or an object with substantial 3D extent. This is because the image of a single flat object may look similar in an image

Figure 1. A calibration object made from 3 lenticular arrays. Each lenticular array has an observed color that changes depending on its viewing angle. (Left) When viewed from reasonably far away, the arrays have relatively consistent colors because they are being viewed from approximately the same angle. (Right) A wide angle view from much closer has significant color variation because the direction from the camera to different parts of the object varies substantially. This paper uses this color variation to derive strong geometric constraints for simultaneous, single-image pose estimation and camera calibration.

taken from a wide angle camera near the object or a more zoomed in camera far away.

This creates a challenge for augmented reality applications that are required to work with dynamic scenes and changing camera parameters such as a movie shot that requires camera zooming. This paper offers a new approach to geometric calibration of a camera that requires a single image of a calibration object that may lie within a larger scene. The calibration object that we propose is shown in Figure 1.

This calibration object is based on several lenticular arrays. A lenticular array is a sheet of plastic which is comprised of many tiny parallel cylindrical lenses. Figure 2 de-
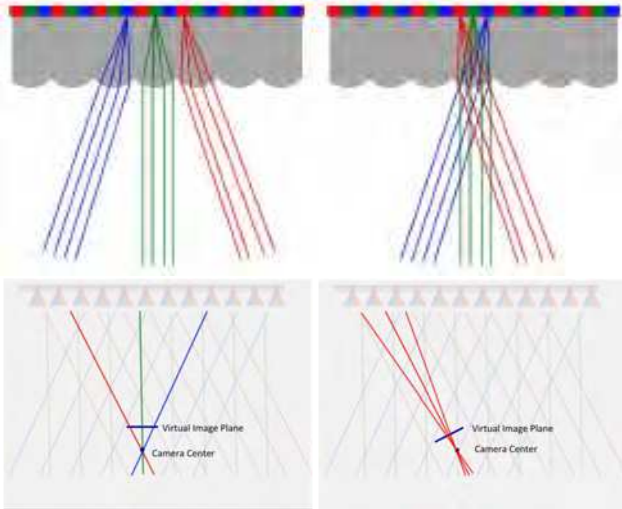
Figure 2. (Top Left) The side view of a lenticular array shows how parallel light rays are focused onto the back focal plane. (Top Right) Interleaving different hues behind each lens makes a lenticular pattern whose apparent hue depends on the viewing angle. At the scale of the whole lenticular array, this means that the perspective of a camera with a wider field of view will see many different colors (Bottom Left), while a camera with a narrow field of view will only see a few hues (Bottom Right).

picts how these lenses focus parallel rays of light onto an interleaved pattern on the back of the lenticular array. As a result, for different viewpoints, the lenticular array has different appearances. In the case of children's toys, different frames of an animation are interleaved, and thus create the appearance of an animation as the lenticular array is rotated [14]. Some modern TVs use lenticular arrays to show different images to different viewer directions in order to create a 3D effect without the need for extra equipment like special glasses [12, 8].

We construct a lenticular pattern inspired by [17], based on a pattern of interleaved colors. This creates an apparent hue dependent on the relative incident angle of light rays viewing the lenticular array. For a perspective camera viewing a planar surface, pixels may have differing viewing angle and therefore will measure a different hue. Therefore as seen in the bottom row of Figure 2, a camera with a wide field of view would see many different hues, while a camera with a narrow field of view would see fewer hues. This fundamental relationship between a color-coded lenticular array and a camera provides a novel geometric constraint to calibrate a camera.

The contributions of this paper are:

- an extension of the constraints derived in [17] to allow lenticular patterns to constrain the camera focal length,
- an approach to correct manufacturing problems of alignment and stretching that make relationship be-

tween color and angle vary across a lenticular array, and

- experimentation with a physical instantiation of a prototype calibration object, showing calibration accuracy in different settings and a complete end-to-end augmented reality demonstration with a variable zoom video.

2. Related Work

Geometric camera calibration is a well studied problem, and geometric constraints relating correspondences, camera motion and camera calibration are well understood [7, 19, 18], including the development of popular toolboxes [24, 2]. Perhaps the most common approach to camera calibration is Zhang's method based on taking many pictures of a grid of known size [25]. The method simultaneously solves for the pose of the grid in each frame and the camera parameters that are consistent across many frames.

Calibrating a camera with a single image is possible by imaging calibration objects with known 3D geometry, for example with 2 orthogonal planes [18]. Other methods use known shapes such as 1D lines [13] and 2 co-planar circles [4] or images of man-made objects such as buildings in a city and exploit vanishing lines [6, 1, 5] or low-rank textures [26] as a source of calibration constraints. With many images from a variety of viewpoints, more standard structure from motion approaches can be used, and these have been specialized for the case of unknown focal lengths in various ways, including understanding the minimal constraint sets [3], extensions to uses features such as line segments and right angles to provide stronger geometric constraints [10], and a linearization of the EPnP problem [11] to speed up the estimation of pose for the uncalibrated camera case [15].

In Augmented Reality (AR), one seeks to render digital content on top of a video feed of a scene to digitally enhance the physical world. In order to properly project digital elements into the video feed, the relative pose of the digital element and the camera must be known, and the camera calibration must be known.

Calibration methods which require multiple images of an object in different poses or calibration objects with substantial variation in all 3 dimensions may not be appropriate for all applications. To our knowledge, the only papers to study this problem in the context of an augmented reality application are the recent work of Taketomi, et al. [20, 21]. They use a KLT tracker to find many points in the scene and use a Structure from Motion formulation to solve for 3D scene points, and the camera position, rotation and focal length in every frame, regularizing all camera parameters to vary smoothly throughout the video. Then, a fiducial marker pattern is located in each frame, and an object is rendered in a position defined by that fiducial marker, with a perspective

that depends on the estimated camera focal length.

In contrast to these existing approaches, our approaches calibrates every image independently using a flat calibration object. This object is based on a lenticular pattern. Lenticular arrays and their 2D counter-part, microlens arrays, give geometric constraints on the incident angle of light rays viewing the arrays. Previous research has used color coded microlens arrays as "light field probes" for Schlieren photography [22], imaged reflections of these arrays to support the reconstruction of surface geometry of transparent objects [23], and for the reconstruction of the refractive index of gases [9]. Large lenticular arrays have been used to estimate object rotation [17] with correspondences, while very small lenticular arrays were used as fiducial markers to estimate the pose of an object [16].

Our work addresses the problem of joint intrinsic and extrinsic camera calibration needed in AR applications where cameras may change their zoom to keep track of an object. Such an approach is necessary to integrate AR with new commercial systems such as Soloshot (http://soloshot.com) that sell Pan-Tilt-Zoom cameras that automatically track a radio tag, but which do not have encoders on their zoom lenses that tag imagery with the zoom level.

The next section introduces our calibration object that is suitable for such applications, We then derive the geometric constraints this object offers and evaluate the performance both quantitatively and qualitatively.

3. AR Calibration Object

Our calibration object is inspired by the lenticular array used to estimate object rotation [17]; we adopt their notation in our discussion here. Three lenticular arrays are mounted perpendicular to each other on a plane, where the 2 flanking arrays have the same orientation, but orthogonal to the middle array. These arrays are orthogonal so that any rotation of the calibration object creates a change; when the object is oriented as shown on the left of Figure 1, rotation around the horizontal axis causes the two edge arrays to change color, while rotating around the vertical axis causes the central part to change color. Small black strips are added to make it easier to distinguish the 3 arrays when they are oriented so that their colors are similar.

Calibrating a Color Coded Lenticular Array   The relationship of the apparent viewing angle to the observed color depends on the relative angle of the lenticular array, in particular the rotation around the axis of the lenticular lenses. The relationship between this rotation and observed color is captured in the Hue Response Function (HRF), which is a 1-to-1 relationship for incident angles of up to ≈ 40 degrees (after which the colors repeat).
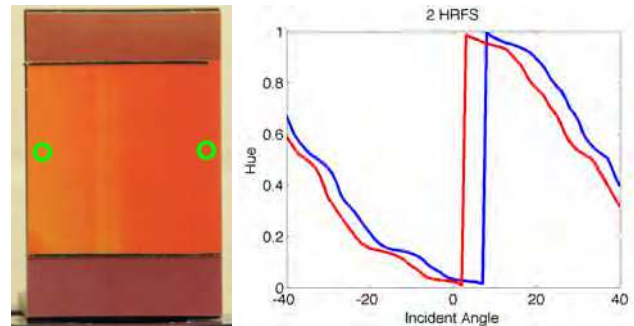

Figure 3. (Left) an image of the calibration object taken with a very long lens, observing all parts of the calibration array with a nearly orthographic imaging geometry. The observed color differences indicate the the hue for a given viewing direction is not consistent. (Right) The observed hue measured at the two yellow circles as the calibration object is rotated, showing a consistent bias.

For a lenticular array created as described in [17], we found that the hue response function varies across the array. We demonstrate this in Figure 3, which shows a picture of the calibration object taken from far away with a very long focal length length lens, giving a field of view in this picture of 1 degree. Therefore the color difference observed at the two highlighted circles is substantial. When measuring this color difference as the calibration array rotates, we see a consistent shift. We believe this is due to the challenges of manufacturing and printing the color coded pattern that sits behind each lenticular lens. For this lenticular array that has 2 lenticular lenses per millimeter, if the backplane is stretched 0.1mm extra over the course of this array, this would cause the observed color shift.

To address this challenge, while still employing standard manufacturing processes for lenticular arrays, we explicitly calibrate the HRF, the function that relates hue to orientation, at regular intervals in the local reference frame of the arrays. We use the corners of the rectangular lenticular arrays as anchor points, and for each image compute a homography mapping the observed lenticular patterns to a canonical coordinate system. The process is illustrated in Figure 4. This calibration object is placed on a controlled rotation mount and rotated through 1 degree increments. For each calibration point, we record the angle at which that calibration point is viewed (which may vary across the calibration grid because of perspective effects), and the measured hue for that angle. The result of this is a curve like those shown in Figure 3 for each of the calibration points. This process is repeated separately for the center and each of the two side lenticular arrays that make up the calibration object.

The next section derives how images of this calibration object can exploit the measured colors for additional geometric cues to the pose of the calibration object and the focal length of the camera. When converting the observa-
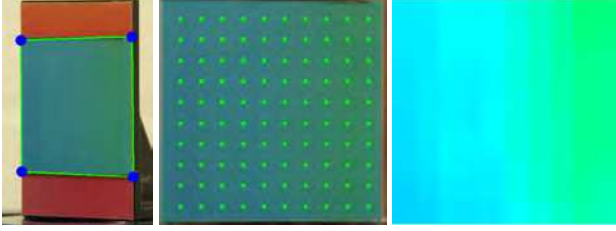
Figure 4. To take hue measurements for calibration and estimation, we employ the following strategy: With the original image (left), we identify anchor points, shown as blue points, at the corners of the lenticular array. These points are used to learn a homography (center) where we then take local measurements evenly across the array, shown as green crosses. (Right) a simplified image of the hue recorded at each calibration point.

tions of the object into geometric constraints, we again find the corners of the array to compute the homography, and sample the colors at these same grid locations.

4. Calibration Constraints

In this section we derive the calibration constraints that arise from observing the lenticular calibration object. We follow the presentation of [17] which solved for rotation given known calibration. They derive the geometric constraint that says that a ray observing a particular color must lie in a plane, and they represent that plane by it's surface normal n_{hue}. The overall constraints for the imaging geometry start with a pixel $\vec{p}$ that observes the lenticular array. In the coordinate system of the camera, a pixel $\vec{p}$ captures light traveling along a ray r that depends on the calibration matrix K as:

$$\vec{r} = K^{-1}p \qquad (1)$$

In the coordinate system of the lenticular pattern, this ray has direction $RK^{-1}p$, and it must satisfy the constraint $(RK^{-1}p) \cdot \vec{n}_{hue} = 0$.

$$
\begin{aligned}
\text{This can be written as} \quad & (RK^{-1}p)^\top \vec{n}_{hue} = 0, \\
\text{which is equivalent to:} \quad & p^\top K^{-1^\top} R^\top \vec{n}_{hue} = 0. \\
\text{Collecting terms, we write:} \quad & p^\top (K^{-1^\top} R^\top) \vec{n}_{hue} = 0, \\
\text{and re-write as:} \quad & p^\top (RK^{-1})^\top \vec{n}_{hue} = 0.
\end{aligned}
$$

Given a pixel location p and a $\vec{n}_{hue}$, this linear constraint on K and R. Previous work [17] uses this constraint and a nonlinear optimization to solve for R given a known K matrix. In this paper we use a similar optimization to get an initial estimate for R and K by parameterizing K by it's focal length f:

$$K(f) = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (2)$$

In $K(f)$, pixels are assumed to be square and x_0 and y_0 are assumed to be the center of the image.

Algorithm: For each frame, our algorithm follows these steps to get an initial estimate of the calibration object pose and camera focal length:

1. Find the four corners of the lenticular calibration object.

2. Solve for the homography to map image coordinates onto object coordinates.

3. Measure the hue at each grid point on the homography, and use these hue measurements to solve for an initial estimate of the rotation and the focal length.

4. Given that rotation and focal length, use the lenticular marker based constraints introduced in [16] to get an estimate of the object translation.

The initial estimate is refined by minimizing the following cost function:

$$\operatorname*{argmin}_{\rho,T,f} \sum_i \left(h(R(\rho),T,f,i) - hue_i\right)^2 + \lambda \|g(R(\rho),T,f) - p_i\|_2^2 \qquad (3)$$

where the first term penalizes the difference between hue_i which is the measured hue at grid-point i (of all lenticular arrays), and $h(R(\rho),T,f,i)$, the hue predicted for grid point i when it is projected onto the image based on camera intrinsic and extrinsic parameters R, T, f, using the HRF function calibrated for grid-point i. Here, R is parameterized via rodrigues parameters ρ. The second term measures the spatial reprojection error between the location p_i and the predicted location for that grid point $g(R,T,F,i)$ based on R, T and f. A relative weighting function λ was found empirically to balance hue and position error which are measured in very different coordinate systems. In all experiments we show, λ was set to 1/4000.

5. Experimental Results

In this section we quantify the ability of our algorithm and this calibration object to estimate the object pose and camera focal length. We also explore the sensitivity of parts of the algorithm to various intermediate processing steps.

The first stage of the algorithm is tracking the corners of the calibration object. This is a vital step in most AR pose estimation algorithms, but it has additional importance in our algorithm because we are modeling the fact that the HRF that maps color to angle may vary across the calibration object. Thus, in Section 5.1 we evaluate the sensitivity

Distribution of Lookup errors per HRF

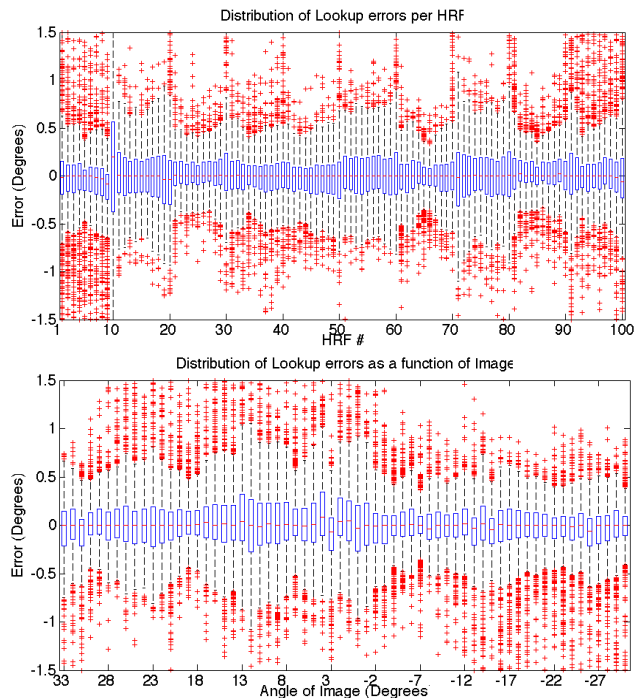**Distribution of Lookup errors as a function of Image**

Figure 5. To create location specific HRF functions that map measured hues to angular constraints, we sample hues of a lenticular array at a local grid of points. The location of these points depends on localizing the anchor points at the corners of the lenticular array. We show the small prediction errors for 8 px permutations of these anchor points per HRF (top) and per image (bottom).

of the algorithm to errors in corner tracking. Second, Section 5.2 explores the accuracy of our approach for estimating the focal length of the camera and rotation and translation of the calibration object. We characterize the error using our physical prototype with different cameras and with the calibration object oriented in different directions. We conclude with Section 5.3 with results showing an object added to a video taken with varying focal length, and compare the realism of the added AR object when there isn't the ability to dynamically estimate the focal length.

5.1. Sensitivity to point tracking

Because the lenticular array may not have the same mapping from angle to color everywhere, we need to know where on the calibration pattern we are measuring the color in order to look up the correct location-specific HRF. Therefore, this approach may be especially sensitive to estimating the position of the corners of the calibration object. We evaluate this by rotating the lenticular array around the vertical axis in one degree increments from −35 to 35 degrees. For each image, we follow these steps:

1. determine 4 anchor points of the lenticular array,

2. project the lenticular array into the local reference frame via a homography

3. sample the hue from the grid-points of the local reference frame image.

For each grid point we compute the angle at which the point was viewed to the angle predicted by the measured hue. To estimate the effect of noise in estimating the lenticular array corners, we perturb the anchor points by 8 pixels in random directions 20 times per image and assess the difference in angle predicted by the HRFs. We show the scale of one such perturbation in the supplementary material (Figure S1).

Figure 5 shows results. The top shows a box and whisker plot showing the distributions of errors in estimating the angle for each of the 100 grid points where the HRF was calculated. The box in each columns shows the 25th and 75th percentiles of the distribution. This experiment shows that modeling the HRF at each location of the lenticular array leads to nearly all angular measurements being within 0.5 degrees of the true incident angle.

We also evaluate if the errors in estimating angle from hue depend on the angle at which the calibration object is observed. Figure 5 computes the distribution of errors across the entire array for each image angle. Again the error is consistently small, even though these statistics are computed using anchor points that are substantially perturbed.

We believe that an error of $0.25°$ is near the limit of a simple geometric constraint based on hue measured at one pixel. The lenticular array shows colors across the hue spectrum over a range of about $40°$, so $0.25°$ is less than 1% of the range of angles that are viewed. Reliably measuring the hue of pixels in 8-bit RGB images to better than 1% precision is also challenging. In Section B of the supplementary material, we explore the color precision of an 8-bit RGB camera and subsequently the angular precision when viewing a lenticular array.

5.2. Pose and Focal Length Estimation

In a laboratory setting, we assess the performance of rotation, translation, and focal length estimation across different viewpoints. On a motorized stage we rotate the calibration object in increments of 5 degrees from −25 to 25 degrees around the vertical axis and take images at each increment. We calibrate the ground truth camera focal length with the MATLAB 2014a implementation of Zhang's method [25].

Figure 6 shows the rotation estimation performance per image in the left column as well as in summary in the right column. We show rotation error for each local axis as the angular difference of our estimate to the true rotation. The estimates from our initialization algorithm is shown in the top row and show errors at the scale of a few degrees. The
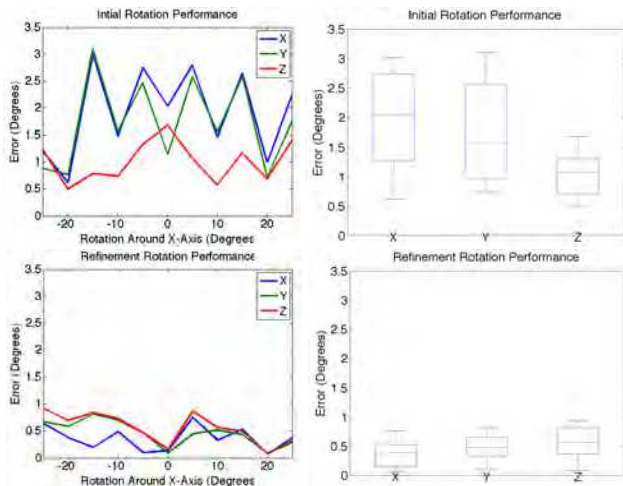
Figure 6. We compare the rotation estimations our method gets initially and after refinement. In the left column, we see the rotation error per local axes for each image as the calibration object is rotated, while in the right column we see the summary statistics. Although we start with good rotation estimates, the refinement process still gives improvement.
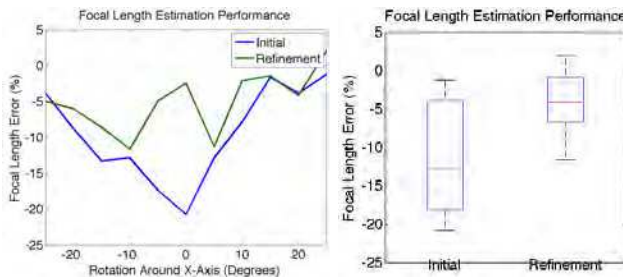

Figure 7. We report the focal length estimations for different orientations of the calibration object per image (on the left) and in summary (on the right). The initial estimations (top) start with considerable error in focal length estimation. After reprojection refinement, however, the results are improved significantly achieving a median of less than 5% error.

bottom of this plot shows results after minimizing the reprojection error as defined in Equation 3, when we get rotation estimates with a median error of 1 degree.

Figure 7 quantifies error in the focal length estimation, and Figure 8 quantifies error in the translation estimation. Both the initialization and refinement results shown strong correlations between the focal length error and the translation error. The refinement step reduces the error of both to a median error of about 4%. The correlation in error between the focal length and the translation arises from the ambiguity that an object can appear bigger either by moving closer to the camera or by the camera changing its focal length. In the AR demo shown later, we see that a 4% error does not appear to lead to a perceptually noticeable error in rendering
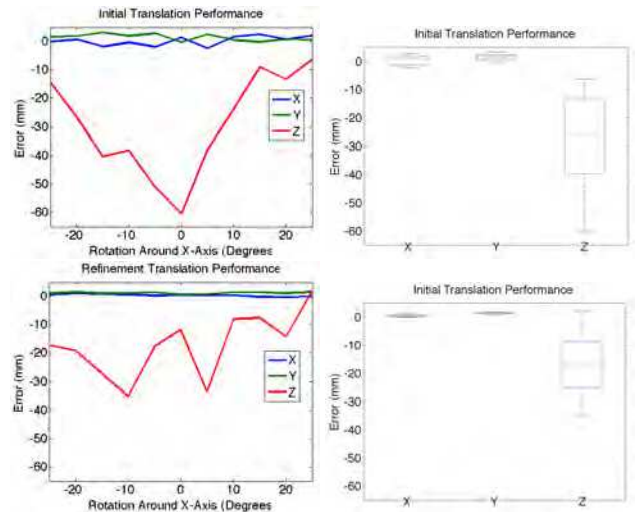

Figure 8. We present the per image (left column) and summary performance statistics (right column) for initial translation estimation (top) and refined translation estimation (bottom). In these plots, we show the distance error for each axis. The overwhelming majority of error is in the Z-axis, which is the depth of the camera. The z-axis translation errors reflect errors in estimating focal length.

the correct perspective of the object.

Figure 9 shows quantitative results for focal length estimation from single images of the calibration object taken at different orientations and different focal lengths. For each image, we show the results that visualize rotation by rendering the local coordinate system on top of the original image. The image title shows the ground truth focal length, our estimated focal length, and the percent error. We include images from cell phone cameras, as well as a DSLR camera. The first two images are from an iPhone 5 and a Galaxy S6 with focal lengths of 5 and 5.8 mm. The images following those are from a Nikon D90 at focal lengths of 18, 49, 90, 115, and 185 mm.

Focal length estimates are relatively accurate for shorter focal lengths. Very long focal lengths correspond to imaging geometries with a smaller field of view. For small fields of view, small errors in estimating angular constraints may lead to larger errors in estimating focal length. To ground this, we show the impact of mis-estimating the field of view by $0.25°$ degrees on the estimate of the focal length.

5.3. Augmented Reality Application

In a desktop scene, we record video of the calibration object while moving the camera in a freehand trajectory. When the camera is moved farther away from the scene and the calibration object, we digitally zoom to keep the calibration object as large as possible in the image. For each frame we estimate the camera focal length, rotation, and transla-
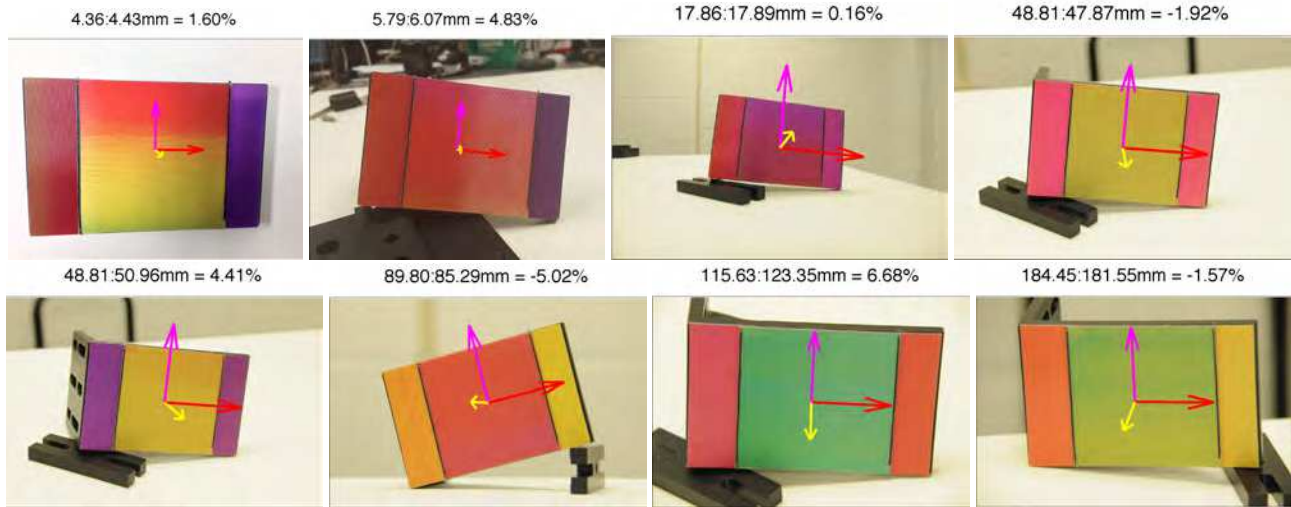
Figure 9. Eight examples of single frame estimation of the focal length and object rotation estimates. The first two images are taken from an iPhone and Galaxy S6, while the remaining images are taken by a Nikon DSLR camera at different zoom settings.
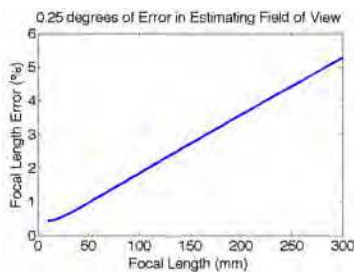

Figure 10. The focal length error that arises from mis-estimating the field of view by $0.25°$ changes as the field of view gets smaller (and, correspondingly, the focal length gets longer).
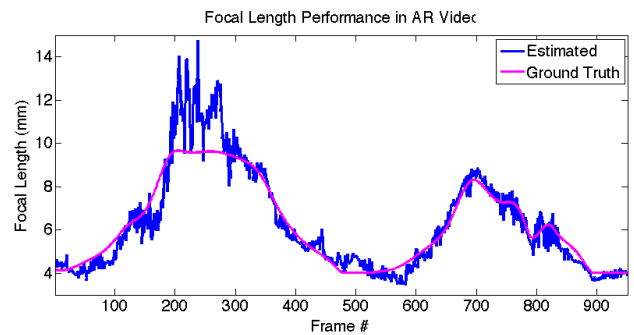

Figure 11. We show the focal length estimates used to render a box into a video in our AR demonstration. We show how our estimates follow the focal length changes from zooming.

tion using the calibration object as detailed in Section 4. In Figure 11, we compare our estimated focal length with the ground truth focal length (which we know because this is a digital zoom) per frame. We can see that the focal length estimations follow the zooming trajectory well. We emphasize that our algorithm does not have access to this digital zoom information.

As a comparison, we consider an AR algorithm that doesn't have access to the digital zoom and does not have the ability to estimate it from image data. When such an algorithm uses a pre-calibrated focal length which becomes wrong in part of the video sequence, virtual objects are rendered with incorrect perspective. Figure 12 shows 3 frames from the video in each column. We render a virtual wire-frame box to highlight perspective effects. The top row shows the original images, the center row shows the box rendering given the estimates made with a dynamic focal length, and the bottom row shows the box rendering given the estimates made with a static focal length. The digital

box has a base the size of the calibration object and is 45mm deep.

Our scene contains graph paper that is aligned to show a cartesian coordinate to help the viewer assess perspective effects. The wire-frame box should appear aligned just short (10mm or 2 boxes) of the end of the paper grid. In comparing our method of estimating a dynamic focal length against estimating a static focal length, we see that the rendered box looks unnaturally too large and with too much perspective in the case of a static focal length. This holds true in general for all frames, and we include the entire video in the supplementary materials.

In the supplementary material, we show an additional AR video. In this video, we render a 3D model into a video with free-hand camera zoom.
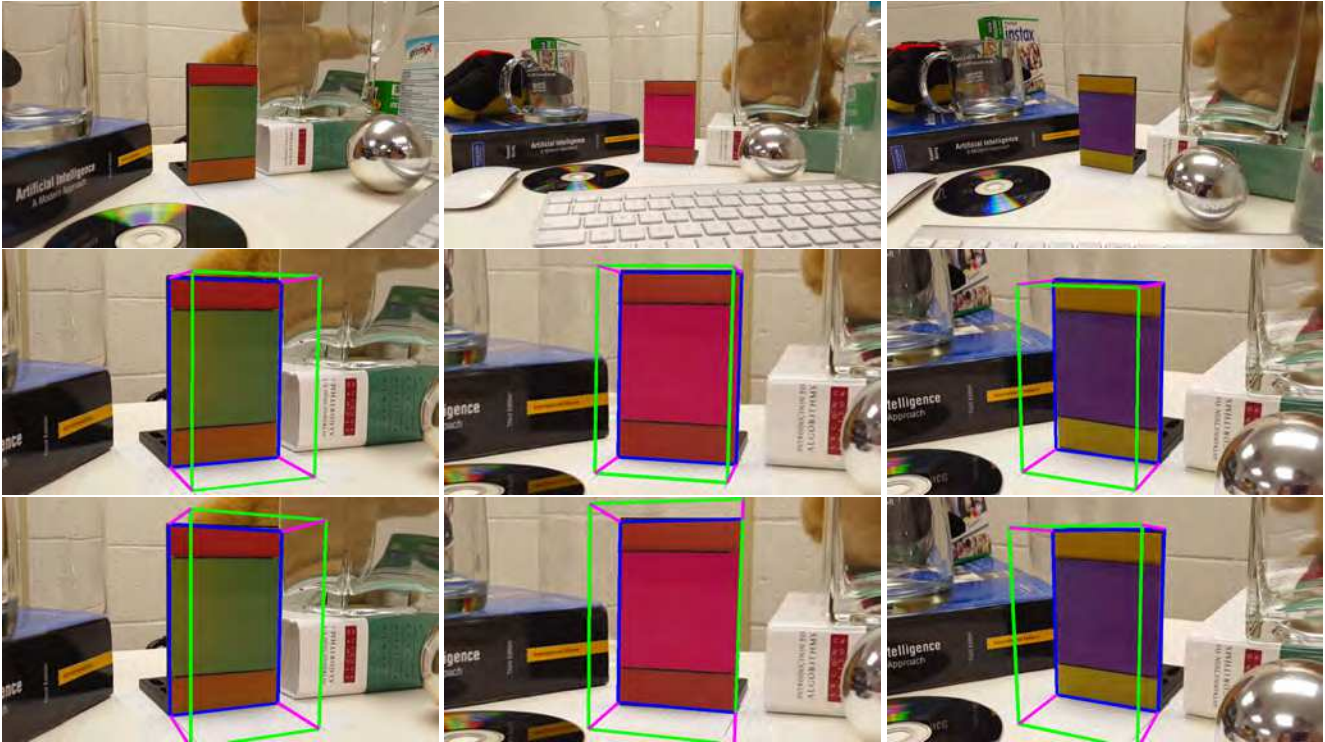
Figure 12. We show focal length estimation results in 3 frames of a video where we render a box over the calibration object. The original image (top row) is digitally zoomed to maximize the size of the calibration object in the image. By estimating the focal length dynamically in each image (middle row) versus estimating a single static focal length (bottom row), we achieve a much more natural rendering that is the correct relative size and has the right amount of perspective.

6. Discussion

We present an end to end system and physical calibration object for simultaneous camera focal length estimation and pose-estimation. This calibration object uses lenticular arrays that offer geometric constraints based on the observed color, allowing for the estimation of camera focal length from a single image. In contrast to [3] which suffers from formal ambiguity and thus high error for fronto-parallel views when using coplanar points [18], our calibration object based on lenticular arrays is robust for all viewing angles. The ability to estimate focal length from a single image is vital for augmented reality applications from cameras that are zooming as they record a scene.

Our geometric constraints are based on measuring image colors. Color is notoriously difficult to measure accurately. Apparent colors may change based on lighting or camera white balance settings. In this paper we did not address this issue, but recent work suggests that these effects can be mitigated by explicitly including a color correction term in the geometric optimization [16].

References

[1] Using vanishing points for camera calibration. *International Journal of Computer Vision*, 4(2):127–139, March 1990. 2

[2] J.-Y. Bouget. Camera calibration toolbox for matlab. http://vision.caltech.edu/bouguetj/calib_doc/index.html. Accessed: 2013-10-09. 2

[3] M. Bujnak, Z. Kukelova, and T. Pajdla. A general solution to the p4p problem for camera with unknown focal length. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 2, 8

[4] Y. Chen, H. Ip, Z. Huang, and G. Wang. Full camera calibration from a single view of planar scene. *Proceedings of the International Symposium on Advances in Visual Computing*, pages 815–824, 2008. 2

[5] J. Deutscher, M. Isard, and J. MacCormick. Automatic camera calibration from a single manhattan image. In *Proc. European Conference on Computer Vision*, pages 175–205, 2002. 2

[6] G. Gracie. Analytical photogrammetry applied to a single terrestrial photograph mensuration. 1968. 2

[7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003. 2

[8] M. Hirsch, G. Wetzstein, and R. Raskar. A compressive light field projection system. *ACM Transactions on Graphics (TOG)*, 33(4):58, 2014. 2

[9] Y. Ji, K. Ye, and J. Yu. Reconstructing gas flows using light-path approximation. In *International Conference on Computational Photography (ICCP)*, 2013. 3

[10] Y. Kuang and K. Astrom. Pose estimation with unknown focal length using points, directions and lines. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 529–536. IEEE, 2013. 2

[11] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2):155–166, 2009. 2

[12] W. Matusik and H. Pfister. 3d tv: a scalable system for real-time acquisition, transmission, and autostereoscopic display of dynamic scenes. In *ACM Transactions on Graphics (TOG)*, volume 23 (3), pages 814–824. ACM, 2004. 2

[13] I. Miyagawa, H. Arai, and H. Koike. Simple camera claibration from a single image using five points on two orthogonal 1-d objects. *IEEE Transactions on Image Processing*, 19(6), 2010. 2

[14] T. Okoshi. *Three-dimensional imaging techniques*. Elsevier, 1976. 2

[15] A. Penate-Sanchez, J. Andrade-Cetto, and F. Moreno-Noguer. Exhaustive linearization for robust camera pose and focal length estimation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(10):2387–2400, 2013. 2

[16] I. Schillebeeckx, J. Little, B. Kelly, and R. Pless. The geometry of colorful, lenticular fiducial markers. In *International Conference on 3D Vision*, 2015. 3, 4, 8

[17] I. Schillebeeckx and R. Pless. Structured light field design for correspondence free rotation estimation. In *International Conference on Computational Photography (ICCP)*, 2015. 2, 3, 4

[18] P. Sturm and S. Maybank. On plane-based camera calibration: A general algorithm, singularities, applications. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 1999. 2, 8

[19] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer, 2010. 2

[20] T. taketomi, K. Okada, G. Yamamoto, J. Miyazaki, and H. Kato. geometric registration for zoomable camera using epipolar constraint and pre-calibrated intrinsic camera parameter change. In *mixed And Augmented Reality (ismar), 2013 Ieee International Symposium On*, pages 295–296. ieee, 2013. 2

[21] T. Taketomi, K. Okada, G. Yamamoto, J. Miyazaki, and H. Kato. Camera poes estimation under dynamic intrinsic parameter change for augmented reality. *Computers and Graphics*, 2014. 2

[22] G. Wetzstein, R. Raskar, and W. Heidrich. Hand-Held Schlieren Photography with Light Field Probes. In *International Conference on Computational Photography (ICCP)*, 2011. 3

[23] G. Wetzstein, D. Roodnick, W. Heidrich, and R. Ramesh. Refractive shape from light field distortion. In *Proc. IEEE International Conference on Computer Vision*, 2011. 3

[24] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Proc. IEEE International Conference on Computer Vision*, pages 666–673, 1999. 2

[25] Z. Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEE Transactions on*, 22(11):1330–1334, 2000. 2, 5

[26] Z. Zhang, Y. Matsushita, and Y. Ma. Camera calibration with lens distortion from low-rank textures. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2321–2328. IEEE, 2011. 2