Distinguishing Refracted Features using Light Field Cameras with Application to Structure from Motion

Dorian Tsai[1], Donald G. Dansereau[2], Thierry Peynot[1] and Peter Corke[1]

***Abstract*—To be effective, robots will need to reliably operate in scenes with refractive objects in a variety of applications; however, refractive objects can cause many robotic vision algorithms, such as structure from motion, to become unreliable or even fail. We propose a novel method to distinguish between refracted and Lambertian image features using a light field camera. Where previous refracted feature detection methods are limited to light field cameras with large baselines relative to the refractive object, our method achieves comparable performance, and we extend these capabilities to light field cameras with much smaller baselines than previously considered, where we achieve up to 50% higher refracted feature detection rates. Specifically, we propose to use textural cross-correlation to characterise apparent feature motion in a single light field, and compare this motion to its Lambertian equivalent based on 4D light field geometry. For structure from motion, we demonstrate that rejecting refracted features using our distinguisher yields lower reprojection error, lower failure rates, and more accurate pose estimates when the robot is approaching refractive objects. Our method is a critical step towards allowing robots to operate in the presence of refractive objects.**

***Index Terms*—Computer Vision for Automation; Visual-Based Navigation; Computational Imaging; Light Fields**

I. Introduction

Robots for the real world will inevitably interact with refractive objects. Robots must contend with wine glasses and clear water bottles in domestic applications [1]; glass and clear plastic packaging for quality assessment and packing in manufacturing [2]; as well as water and ice for outdoor operations [3]. Automating these applications typically requires either object structure and/or robot motion to automate. Structure from motion (SfM) is a technique to recover both scene structure and camera pose from 2D images, and is widely applicable to many systems in computer and robotic vision [4], [5]. Many of these systems assume the scene is Lambertian, in that a 3D point's appearance in an image does not change significantly with viewpoint. However, non-Lambertian effects, including specular reflections, occlusions, and refraction, violate this assumption. They pose a major problem for modern robotic vision systems because their appearance depends on the camera's viewing pose and the visual texture of the object's background.

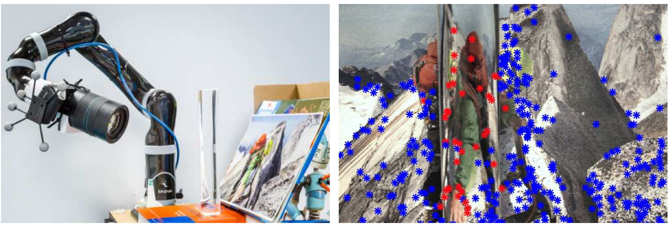


Fig. 1. (Left) A light field camera mounted on a robot arm was used to distinguish refractive objects in a scene in SfM experiments. (Right) SIFT features that were distinguished as Lambertian (blue) and refracted (red), revealing the presence of the refractive cylinder in the middle of the scene.

Image features are distinct points of interest in the scene that can be repeatedly and reliably identified from different viewpoints, and have been used in SfM, but also many other robotic vision algorithms, such as object recognition, image segmentation, visual servoing, visual odometry, and simultaneous localization and mapping (SLAM). In SfM, features are often used for image registration, and serve as a basis for the entire SfM pipeline. When reconstructing a scene containing a refractive object, such as Fig. 1, image features visible through the object appear to move differently from the rest of the scene. They can cause inconsistencies, errors, and even failures for modern robotic vision systems.

Light field (LF) cameras offer a potential solution to the problem of refractive objects. LF cameras simultaneously capture multiple images of the same scene from different viewpoints in a regular and dense sampling. The LF could allow robots to more reliably and efficiently capture the behaviour of refractive objects in a single shot by exploiting the known geometry of the multiple views. We take 2D image features from the central view of the LF, and determine which of these have been distorted in the 4D LF, which we refer to as refracted features (RFs). We use this as a method of distinguishing good features for SfM.

Our main contributions are the following.

- We extend previous work to develop a light field feature distinguisher for refractive objects. In particular, we detect the differences between the apparent motion of non-Lambertian and Lambertian features in the 4D light field to distinguish refractive objects more reliably than previous work.
- We propose a novel approach to describe the apparent motion of a feature observed within the 4D light field based on textural cross-correlation.

Manuscript received: Sep, 10, 2018; Accepted Nov, 14, 2018.

This paper was recommended for publication by Editor Cesar Cadena, upon evaluation of the Associate Editor and Reviewers' comments.

This research was partly supported by the Australian Research Council (ARC) Centre of Excellence for Robotic Vision (CE140100016).

[1]D. Tsai, T. Peynot and P. Corke are with the Australian Centre for Robotic Vision, Queensland University of Technology (QUT), Brisbane, Australia {dy.tsai, t.peynot, peter.corke}@qut.edu.au

[2]D. Dansereau is with the Australian Centre for Field Robotics, University of Sydney d.dansereau@acfr.usyd.edu.au

Digital Object Identifier (DOI): see top of this page.

- We extend RF distinguishing capabilities to lenslet-based LF cameras that are limited to much smaller baselines by considering non-uniform, non-Lambertian apparent motion in the light field. All light fields captured for these experiments are available at *https://tinyurl.com/LFRefractive*.
- We show that by distinguishing and rejecting refracted features with our method, SfM performs better in scenes that include refractive objects.

The main limitation of our method is that it requires background visual texture to be distorted by the refractive object. Our method's effectiveness depends on the extent to which the appearance of the object is warped in the light field. This depends on the scene geometry and refractive indexes of the object involved.

Next we describe the related work, provide background on LF geometry, and explain our method for distinguishing RFs. We show experimental results for detection with different LF cameras, and validation in the context of monocular SfM. Finally, we conclude the paper and explore future work.

II. RELATED WORK

A variety of strategies for detecting and reconstructing refractive objects using vision have been investigated [2]. However, many of these methods require known light sources with bulky configurations that are impractical for mobile robot applications. Multiple monocular images have been used to recover refractive object shape and pose [6]; however, image features were manually tagged throughout camera motion, emphasizing the difficulty of automatically identifying and tracking RFs due to the severe magnification of the background and image distortion from the object.

LFs have been used to obtain better depth maps for Lambertian and occluded scenes [7]; however, the performance of these algorithms suffers for refractive objects. Wanner et al. recently considered planar refractive surfaces and reconstructed different depth layers that accounted for both refraction through a thin sheet of glass, and the reflection caused by its glossy surface [8]. However, this work was limited to thin planar surfaces and single reflections. Which depth layer was Lambertian, reflective or refractive was not distinguished, and refractive objects that caused significant distortion were not handled. Although our work does not determine the dense structure of the refractive object, our approach can distinguish features from objects that significantly distort the LF.

For refractive object recognition, Maeno et al. proposed a light field distortion feature (LFD), which models an object's refraction pattern as image distortion based on differences in the corresponding image points between the multiple views of the LF, captured by a large-baseline (relative to the refractive object) LF camera array [9]. However, the authors observed significantly poor recognition performance due to specular reflections, as well as changes in camera pose.

Xu et al. used the LFD as a basis for refractive object image segmentation [10]. Corresponding image features from all views in the LF were fitted to the single normal of a 4D hyperplane using singular value decomposition (SVD). The smallest singular value was taken as a measure of error to the hyperplane of best fit, for which a threshold was applied to distinguish RFs. However, we will show that a 3D point cannot be described by a single hyperplane in 4D. Instead, it manifests as a plane in 4D that has two orthogonal normal vectors. Our approach builds on Xu's method and solves for both normals to find the plane of best fit in 4D; thus allowing us to distinguish more types of refractive objects with a higher rate of detection.

Furthermore, a key difficulty in feature-based approaches in the LF is obtaining the corresponding feature locations between multiple views. Both Maeno and Xu used optical flow between two views for correspondence, which does not exploit the unique geometry of the LF. We propose a novel textural cross-correlation method to associate features in the LF by describing their apparent motion in the LF, which we refer to as feature curves. This method directly exploits LF geometry and provides insight on the 4D nature of features in the LF.

Our interest in LF cameras stems from robot applications that often have mass, power and size constraints. Thus, we are interested in employing compact lenslet-based LF cameras to deal with refractive objects. However, most previous works have utilized gantries [8], or large camera arrays [9], [10]; their results do not transfer to LF cameras with much smaller baselines, where distortion is less apparent, as we show later. We then demonstrate the performance of our method over two different LF camera architectures with dramatically different baselines. Ours is the first method, to our knowledge, capable of identifying RFs using lenslet-based LF cameras.

For LF cameras, LF-specific features have been investigated. SIFT features augmented with "slope", a LF-based property related to depth, were proposed by the authors for visual servoing using a LF camera [11]; however, transparent objects were not considered. Recent work by Teixeira et al. projects SIFT features found in all views into their corresponding epipolar plane images (EPIs), and identifies reliable Lambertian features as features that are repeatedly grouped in their respective EPIs [12]. However, their approach did not consider any non-linear feature behaviour, while our method aims to detect these non-Lambertian features, and is focused on characterising them. In this paper, we detect unique keypoints that reject distorted content and work well for SfM. This could be useful for many feature-based algorithms, including recognition, segmentation, visual servoing, simultaneous localization and mapping, visual odometry, and SfM.

We are interested in exploring the impact of our RF distinguisher in a SfM framework. While there has been significant development in SfM in recent year for conventional monocular and stereo cameras [5], Johannsen et al. were the first to consider LFs in the SfM framework [13]. Although our work does not yet explore LF-based SfM, we investigate SfM's performance with respect to RFs, which has not yet been fully explored. We show that rejecting RFs reduces reprojection error and failure rate near refractive objects, improving camera pose estimates.

III. Light Field Background

We parameterize the LF using the relative two-plane parameterization (2PP) [14]. A ray with coordinates $\phi = [s, t, u, v]^T$, where ^{T} represents the vector transpose, is described by two points of intersection with two parallel reference planes; an s, t plane conventionally closest to the camera, and a u, v plane conventionally closer to the scene, separated by arbitrary distance D.

For a Lambertian point in space $\boldsymbol{P} = [P_x, P_y, P_z]^T \in \mathbb{R}^3$, the rays follow a linear relationship [3]

$$\begin{bmatrix} u \\ v \end{bmatrix} = \left(\tfrac{D}{P_z}\right) \begin{bmatrix} P_x - s \\ P_y - t \end{bmatrix}, \tag{1}$$

where each of these equations describes a hyperplane in 4D. A hyperplane is defined as a vector subspace that has 1 dimension less than the space it is contained within [15]. Thus a hyperplane in 4D is a 3D manifold, and can be described by a single equation

$$n_1 s + n_2 t + n_3 u + n_4 v + n_5 = 0, \tag{2}$$

where $\boldsymbol{n} = [n_1, n_2, n_3, n_4]^T$ is the normal of the hyperplane. Similarly, a plane is defined as a 2D manifold; it can be described by two linearly independent vectors. Therefore, a plane in 4D can be defined by the intersection of two hyperplanes and (1) can be re-written in the form,

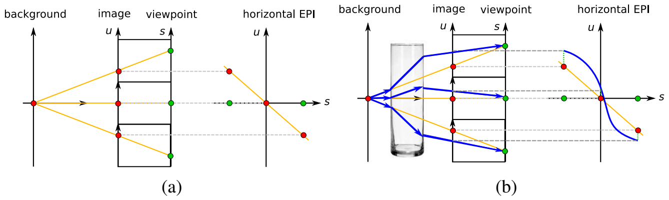
$$\underbrace{\begin{bmatrix} \frac{D}{P_z} & 0 & 1 & 0 \\ 0 & \frac{D}{P_z} & 0 & 1 \end{bmatrix}}_{\boldsymbol{m}} \begin{bmatrix} s \\ t \\ u \\ v \end{bmatrix} = \begin{bmatrix} \frac{DP_x}{P_z} \\ \frac{DP_y}{P_z} \end{bmatrix}, \tag{3}$$

where $\boldsymbol{m}$ contains the two orthogonal normals to the plane. Therefore, a Lambertian point in 3D manifests as a plane in 4D, which is characterized by two linearly-independent normal vectors that each define a hyperplane in 4D. In the literature, this relationship is sometimes referred to as the point-plane correspondence [3].

Light field slope w relates the rate of change of image plane coordinates, with respect to viewpoint position, for all rays emanating from a point in the scene. In the literature, slope is sometimes referred to as "orientation" [8], and other works compute slope as an angle [16]. The slope comes directly from (1) as $w = -D/P_z$, and is clearly related to depth.

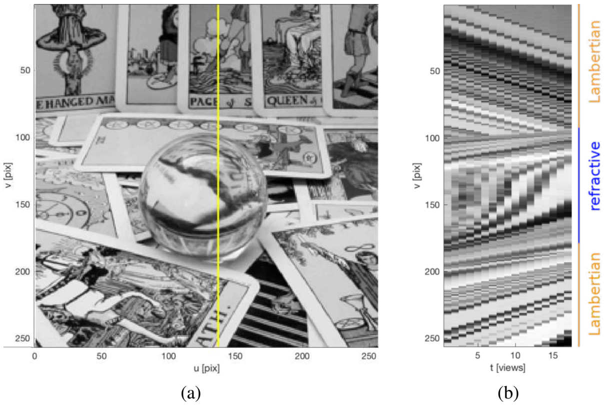
IV. Distinguishing Refractive Features

EPIs graphically illustrate the apparent motion of a feature across multiple views [17]. If the entire light field L is given as $L(s, t, u, v)$, EPIs represent a 2D slice of the 4D LF. A horizontal EPI is given as $L(s, t^*, u, v^*)$, and a vertical EPI is denoted as $L(s^*, t, u^*, v)$, where * indicates a variable is fixed while others may vary. The central view of the LF is $L(s_0, t_0, u, v)$, and is equivalent to what a monocular camera would provide from the same pose. As shown in Fig. 2, features from a Lambertian scene point are linearly distributed with respect to viewpoint, unlike features from highly-distorting refractive objects. We compare this difference in apparent motion between Lambertian and non-Lambertian features to distinguish RFs.



(a) (b)

Fig. 2. (a) Projection of the linear behaviour of a Lambertian feature, and (b) the non-linear behaviour of a refracted feature with respect to linear motion along the viewpoints of a light field.



(a) (b)

Fig. 3. (a) In the crystal ball LF [18], a vertical EPI (b) is sampled from a column of pixels (yellow), where nonlinear RF motion caused by the crystal ball are apparent in the middle (blue). Straight lines correspond to Lambertian features (orange).

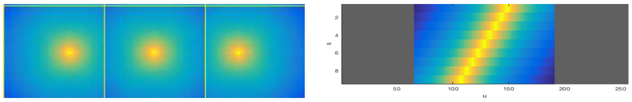
Fig. 3 shows the central view and an example EPI of a crystal ball LF (large baseline) from the New Stanford Light Field Archive, captured by a camera array [18]. The physical size of cameras often necessitates larger baselines for LF capture. A Lambertian point forms a straight line in the EPI, shown in Fig. 3b. The relation between slope and depth is also apparent in this EPI.

RFs appear as nonlinear curves in the EPI, as seen in Fig. 3b. RF detection in the LF simplifies to finding features that violate (1) via identifying nonlinear feature curves in the EPIs and/or inconsistent slopes between two independent EPI lines, such as the vertical and horizontal EPIs. We note that occlusions and specular reflections also violate (1). Occlusions appear as straight lines, but have intersections in the EPI. Edges of the refractive objects, and objects with low distortion also appear Lambertian. Specular reflections appear as a superposition of lines in the EPI. We will address these issues in future work. In this paper, we discuss how we extract these 4D feature curves and how we identify RFs.

A. Extracting Feature Curves

For a given feature from the central view at coordinates (u_0, v_0), we must determine the feature correspondences (u', v') from the other views, which is equivalent to finding the feature's apparent motion in the LF. In this paper, we start by detecting SIFT features [19] in the central view, although the proposed method is agnostic to feature type.

Next, we select a template surrounding the feature which is k-times the feature's scale. We determined $k = 5$ to yield the most consistent results. 2D Gaussian-weighted normalized



(a) (b)

Fig. 4. (a) The cross-correlation response for corresponding views for a typical scene. (b) The resultant correlation EPI, created by stacking the cross-correlation response from adjacent views. The ridge (yellow) along this correlation EPI corresponds to the desired feature curve.

cross-correlation (WNCC) is used across views to yield correlation images, such as Fig. 4a. To reduce computation, we only apply WNCC along the central row and column of LF views.

For Lambertian features, we plot the feature's correlation response with respect to the views to yield a correlation EPI. Illustrated in Fig. 4b, the ridge of the correlation EPI corresponds to the feature curve from original EPI.

For RFs, we hypothesize that the distortion of the feature's appearance between views won't be too strong as to make the correlation response unusable. Thus, the correlation response will be sufficiently strong that the ridge of the correlation EPI will still correspond to the desired feature curve. This textural cross-correlation method allows us to focus on the image structure, as opposed to the image intensities. Our method can be applied to any LF camera, and directly exploits the geometry of the LF.

B. Fitting 4D Planarity to Feature Curves

Similar to [10], we consider the ray passing through the central view $\phi(0, 0, u_0, v_0)$. The corresponding feature coordinates in other views are $\phi'(s, t, u', v')$. The LFD is then defined as the set of relative differences between ϕ and ϕ' as in [9]:

$$LFD(u, v) = \{(s, t, \Delta u, \Delta v) | (s, t) \neq (0, 0)\}, \tag{4}$$

where $\Delta u = u' - u_0$, and $\Delta v = v' - v_0$ are feature disparities.

These disparities are linear with respect to linear camera translation. The disparities from RFs deviate from this linear relation. Fitting them to (1) yields the plane of best fit in 4D, and the error of this fit provides a measure of whether or not our feature is Lambertian.

This plane in 4D can be estimated from the feature correspondences given by the feature curves $f_h(s, t^*, \Delta u, v^* - v_0)$, and $f_v(s^*, t, u^* - u_0, \Delta v)$ that we extract from the horizontal and vertical EPIs, respectively.

As discussed in Section III, our plane in 4D has two orthogonal normals, $\boldsymbol{n}_h$ and $\boldsymbol{n}_v$. The 4D plane containing ϕ can be given as

$$\underbrace{\begin{bmatrix} n_{h,1} & n_{h,2} & n_{h,3} & n_{h,4} \\ n_{v,1} & n_{v,2} & n_{v,3} & n_{v,4} \end{bmatrix}}_{\begin{bmatrix} \boldsymbol{n_h} & \boldsymbol{n_v} \end{bmatrix}^T} \begin{bmatrix} s \\ t \\ \Delta u \\ \Delta v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{5}$$

Note that the constants on the right-hand side of (3) cancel out because we consider the differences relative to u_0 and v_0. The positions for s, t can be obtained by calibration [20], although the nonlinear behaviour still holds when working with uncalibrated units of "views".

We can estimate $\boldsymbol{n}_h$ and $\boldsymbol{n}_v$ by fitting the N points from f_h and M points from f_v into the system,

$$\underbrace{\begin{bmatrix} (s, & t^*, & \Delta u, & v^* - v_0)_1 \\ \vdots & \vdots & \vdots & \vdots \\ (s, & t^*, & \Delta u, & v^* - v_0)_N \\ (s^*, & t, & u^* - u_0, & \Delta v)_1 \\ \vdots & \vdots & \vdots & \vdots \\ (s^*, & t, & u^* - u_0, & \Delta v)_M \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix}}_{\boldsymbol{n}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{6}$$

We then use SVD on $\boldsymbol{A}$ to compute the singular vectors, and corresponding singular values. The 2 smallest singular values, λ_1 and λ_2, correspond to 2 normals $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ that best satisfy (6) in the least-squares sense. The magnitude of the singular values provides a measure of error of the planar fit. Smaller errors imply stronger linearity, while larger errors imply that the feature deviates from the 4D plane.

The norm of λ_1 and λ_2 may be taken as a single measure of planarity; however, doing so masks the case where a refractive object has unequal errors between the two EPIs, such as a 1D refractive object (glass cylinder) that causes severe distortion along one direction, but relatively little along the other. Therefore, we reject those features that have large errors in either horizontal or vertical hyperplanes. This planar consistency, along with the slope consistency discussed in the following section, make the proposed method more sensitive to distorted texture than prior work that considered only the smallest singular value, which we refer to as hyperplanar consistency [10].

C. Measuring Slope Consistency

Slope consistency is a measure of how different the slopes are between the two hyperplanes for a given feature. As seen in (1), these slopes must be equal for Lambertian points. We can compute the slopes for each hyperplane given their normals. For the horizontal hyperplane, we solve for in-plane vector $\boldsymbol{q} = [q_s, q_u]^T$, by taking the inner product of the two vectors $\boldsymbol{n_h}$ and $\boldsymbol{n_v}$ from (5) in

$$\begin{bmatrix} n_{h,1} & n_{h,3} \\ n_{v,1} & n_{v,3} \end{bmatrix} \begin{bmatrix} q_s \\ q_u \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{7}$$

where $\boldsymbol{q}$ is constrained to the s, u plane, because we choose the first and third elements of $\boldsymbol{n}_h$ and $\boldsymbol{n}_v$. This system is solved using SVD, and the minimum singular vector yields $\boldsymbol{q}$. The slope for the horizontal hyperplane, w_{su} is then $w_{su} = q_s/q_u$. The slope for the vertical hyperplane w_{tv} is similarly computed from the second and fourth elements of $\boldsymbol{n}_h$ and $\boldsymbol{n}_v$. Slope consistency c is calculated as the square of differences between slopes. Thus features with large planar errors and inconsistent slopes are identified as belonging to a highly-distorting refractive object. Three thresholds for planar consistency and slope consistency are used to determine if a feature has been distorted (though we refer to it as a refracted feature).

Note that our method is not limited to detecting distortion aligned with the horizontal and vertical axes of the LF. We can further check for λ_1, λ_2 and c along other axes by rotating the LF's s, t, u, v frame and repeating the check.

V. Experimental Results

We present our experimental set-up and show how our methods extend from LF camera arrays to lenslet-based LF cameras. Finally, we use our method to reject RFs for monocular SfM in the presence of refractive objects, and demonstrate improved reconstruction and pose estimates.

A. Experimental Set-up

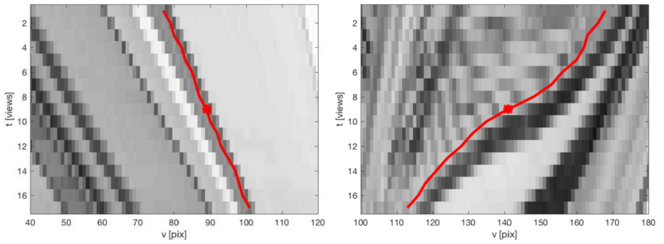
For LFs captured by a camera array, we used the Stanford New Light Field Database [18]. We focused on two LFs that captured the same scene of a crystal ball surrounded by textured tarot cards. The first was captured with a large baseline (16.1 mm/view over 275 mm), while the second was captured with a smaller baseline (3.7 mm/view over 64 mm). This allowed us to compare the effect of LF camera baseline for RFs.

Smaller baselines were considered using a lenslet-based LF camera. Also known as a plenoptic camera, these LF cameras are of interest in robotics due to their simultaneous view capture, and typically lower size and mass, compared to LF camera arrays and gantries. In this section, the Lytro Illum was used to capture LFs with 15×15 views, each 433×625 pixels. Dansereau's Light Field Toolbox was used to decode the LFs from raw LF imagery to the 2PP, converting the Illum to an equivalent camera array with a baseline of 1.1 mm/view over 16.6 mm [20]. To compensate for the extreme lens distortion of the Illum, we removed the outer views, reducing our LF to 13×13 views. The LF camera was fixed at 100 mm zoom.

It is important to remember that our results depend on a number of factors. The geometry and refractive index of a transparent object affects its appearance. Higher curvature and thickness yield more warping. Second, viewing distance, and background distance to the object directly affect how much distortion can be observed. Similarly, a larger camera baseline captures more distortion. When possible, these factors were held constant throughout different experiments.

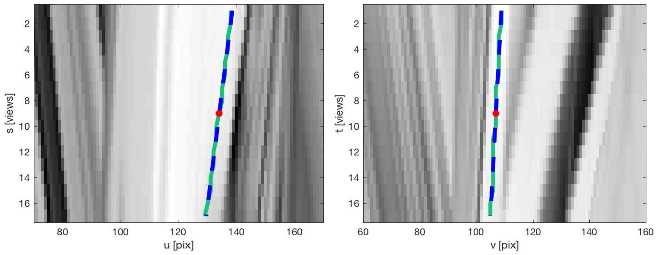
B. RF Detection with Different LF Cameras

For the large baseline crystal ball LF captured by the camera array, Lambertian features were captured by our textural cross-correlation approach as straight lines, while RFs were captured as nonlinear curves, as shown in Fig. 5. We observed that while the RF's correlation response often had a much weaker response compared to the Lambertian case, local maxima were observed near the feature's corresponding location in the central view. Thus, taking the local maxima of the correlation EPI yielded the desired feature curves. Our textural cross-correlation method enables us to extract RF curves without focusing on image intensities.



(a) (b)

Fig. 5. Sample feature curves extracted from the large baseline LF's correlation EPI. (a) A straight Lambertian feature curve (red) in the EPI. (b) A RF curve exhibiting nonlinear behaviour in the EPI.



(a) (b)

Fig. 6. Sample (a) horizontal and (b) vertical EPIs from the crystal ball LF with small baseline. From the feature's (u, v) location in the central view (red), extracted feature curves (green) match the plane of best fit (dashed blue). RFs appear almost linear, and are thus much more difficult to detect.

In contrast, Fig. 6 shows the horizontal and vertical EPIs for a RF taken from the small baseline crystal ball LF. The feature curves appear straight, despite being distorted by the crystal ball. However, we observed that the slopes were inconsistent, which could still be used to distinguish RFs.

To distinguish RFs, thresholds for planarity and slope consistency were selected by exhaustive search over a set of training LFs, while evaluated on a different set of LFs, with the exception of the crystal ball LFs where only 1 was available for each baseline. For comparison to state of the art, parameter search was performed for both Xu's method and our method independently, to allow for the best performance of each method.

The ground truth RFs were identified via hand-drawn masks in the central view. It was assumed that all features visible and passing through the refractive object were distorted. Detecting a RF was considered positive, while returning a Lambertian feature was negative. Thus a true positive (TP) is a correctly detected RF, while a true negative (TN) is a correctly detected Lambertian feature. A false positive (FP) is an incorrectly detected RF. A false negative (FN) is an incorrectly detected Lambertian feature.

Table I shows the detection results, and Fig. 7 shows sample views of refracted features (red) and Lambertian features (blue). For the camera array, only 1 LF was available [18] for each baseline b. For the lenslet-based camera, 10 LFs from a variety of different backgrounds were used for each object type. Our method had up to a 50% higher TP rate (TPR), up to a 58% lower FN rate (FNR), and similar FP rates (FPR) and TN rates (TNR) compared to Xu's method for the camera array, which we attributed to more accurately fitting the plane in 4D, as opposed to a single hyperplane. For the lenslet-based camera, we attributed our 10 times increase in TPR and 3.8

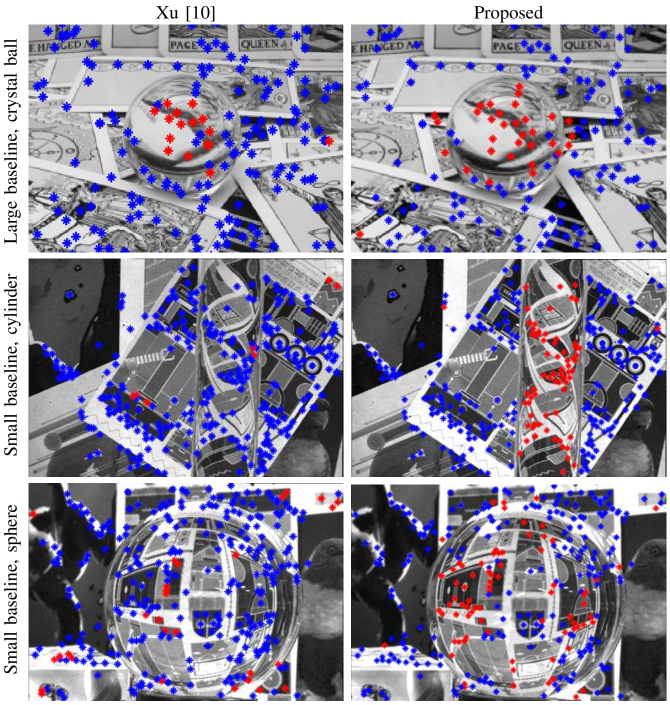


Fig. 7. Comparison of Xu's method (left), and our method (right), detecting Lambertian (blue), and refracted (red) SIFT features. The top row shows the crystal ball captured with a large baseline LF (cropped) [18]. Both methods detect RFs; however, our method outperforms Xu's. In the second and third rows, a cylinder and sphere captured with a small-baseline lenslet-based LF camera. Our method successfully detects more RFs with fewer false positives and negatives.

TABLE I
COMPARISON OF OUR METHOD AND STATE-OF-THE-ART USING A LF CAMERA ARRAY AND LENSLET-BASED CAMERA FOR DETECTING RFS

		Xu's				**Proposed**			
	b [mm]	**TPR**	**TNR**	**FPR**	**FNR**	**TPR**	**TNR**	**FPR**	**FNR**
array		**crystal ball**							
	275	0.58	**0.97**	**0.02**	0.41	**0.66**	0.95	0.05	**0.34**
	68	0.42	0.91	0.08	0.89	**0.63**	**0.94**	**0.05**	**0.37**
lenslet		**sphere**							
	1.1	0.43	0.36	0.64	0.58	**0.48**	**0.95**	**0.04**	**0.52**
		cylinder							
	1.1	0.08	0.80	0.20	0.92	**0.82**	**0.81**	**0.13**	**0.24**

times decrease in FNR for the cylinder case to accounting for slope consistency, which Xu did not address.

The FPs included some occlusions, which appeared nonlinear in the EPI [21], but were not yet distinguished in our implementation. However, this may still be beneficial as occlusions are non-Lambertian, and thus undesirable for most algorithms. Sampling from all the views in the LF would likely improve the results for both methods, as more data would improve the planar fit.

With the lenslet-based LF camera, we investigated two different types of refractive objects: a glass sphere and an acrylic cylinder, shown in the bottom two rows of Fig. 7. The sphere exhibited significant distortion along both the horizontal and vertical viewing axes, while the cylinder only exhibited significant distortion perpendicular to its longitudinal axis. As shown in Table I, Xu's method was unable to detect the refractive cylinder (TPR at 0.08), while our method succeeded with 10 times higher TPR. Nonlinear feature curves were not apparent from the small baseline of the lenslet-based camera, but slope consistency proved to be a very strong indicator of distortion.

Our method detected RFs of the refractive sphere with an 11% increase in TPR, a significant 164% increase in TNR, and 93% decrease in FPR. We attribute this success to accounting for slope consistency. Features that were located close to the edge of the sphere appeared more linear, and thus were not always detected. Other FPs were due to specular reflections that appeared like well-behaved Lambertian points. Finally, there were some FNs near the middle of the sphere, where there is identical apparent motion in the horizontal and vertical hyperplanes.

C. Rejecting RFs for Structure from Motion

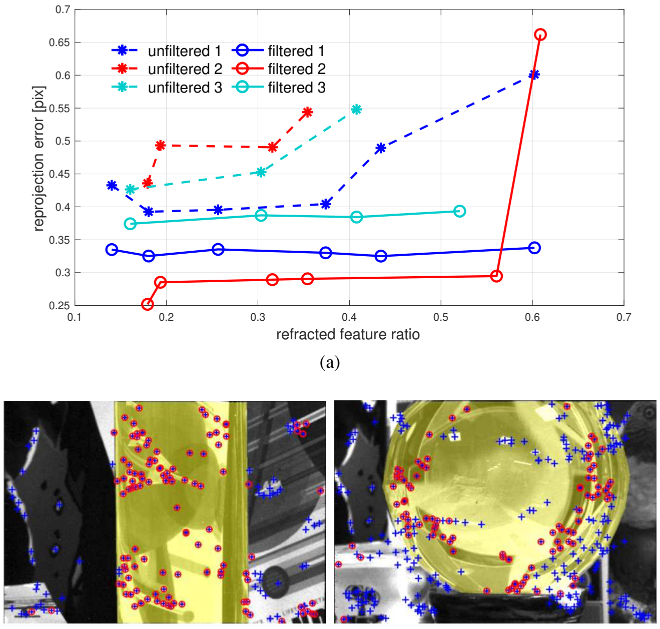
We examine the impact of rejecting RFs in a SfM pipeline. We captured 10 sequences of LFs that gradually approached a refractive object using the same lenslet-based LF camera. We used Colmap, a publicly-available SfM implementation [22]. Incremental monocular SfM using the central view of the LF was performed on the sequences of images. Each successive image had an increasing number of RFs, making it increasingly difficult for SfM to converge. If SfM converged, a sparse reconstruction was produced, and the estimated poses were further analysed. The scene is shown in Fig. 1 with a textured, slanted background plane behind a refractive cylinder.

For each LF, SIFT features in the central view were detected, creating an unfiltered set of features, some of which were refracted. Our distinguisher was then used to remove RFs, creating a filtered set of features. Both sets were imported separately into the SfM pipeline, which included its own outlier rejection and bundle adjustment. This produced respective unfiltered and filtered SfM results for comparison.

We note that outlier rejection schemes, such as RANSAC, are often used to reject inconsistent features, including RFs. While RANSAC successfully rejected many RFs, we observed more than 53% of inlier features used for reconstruction were actually RFs in some unfiltered cases. This suggested that in the presence of refractive objects, RANSAC is insufficient on its own for robust and accurate structure and motion estimation.

We measured the ratio of RFs $r = i_r / i_t$, where i_r is the number of RFs in the image, and i_t is the total number of features detected in the image. We considered the reprojection error as it varied with r. Shown in Fig. 8a, the error for the unfiltered case was consistently higher (up to 42.4% higher for $r < 0.6$ in the red case). Additionally, the unfiltered case often failed to converge, while the filtered case was successful, suggesting better convergence. Sample scenes that caused the unfiltered SfM to fail are shown in Fig. 8b and 8c. These scenes could not be used for SfM without our method to retain consistent features for reconstruction.

For the monocular SfM, scale was obtained by solving the absolute orientation problem using Horn's method between the estimated pose $\boldsymbol{p_s}$ and ground truth pose $\boldsymbol{p_g}$, and only



(a)

(b)

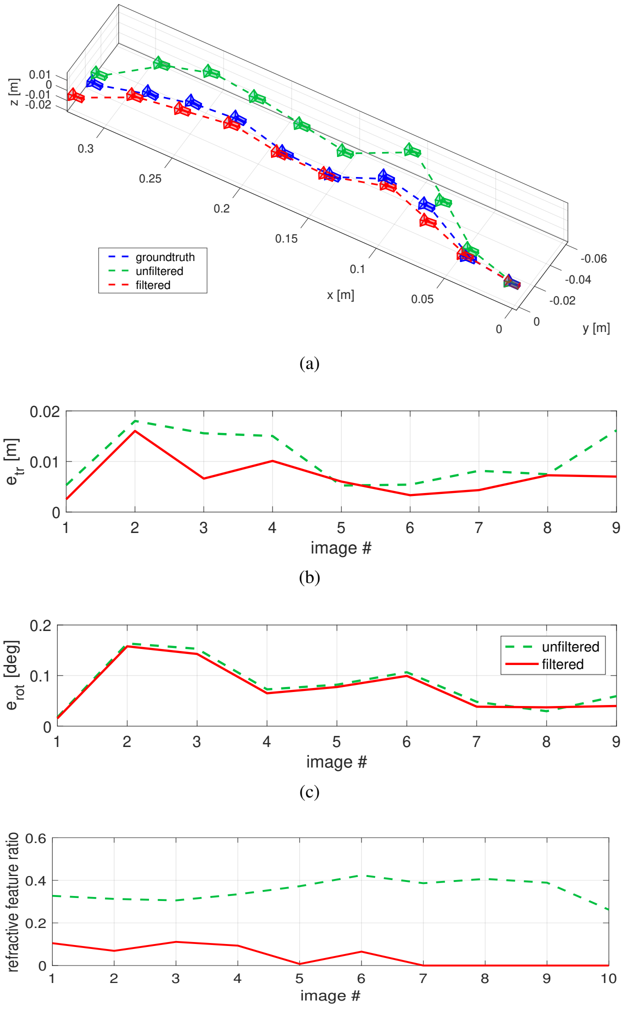
(c)

Fig. 8. Rejecting RFs with our method yielded lower reprojection errors and better convergence for the same image sequences. (a) SfM reprojection error vs RF ratio for the unfiltered case containing all the features, including RFs (dashed), and filtered case excluding RFs (solid). The spike in error at 0.6 r for filtered sequence 2 was due to insufficient inlier matches for SfM to provide reliable results. (b) and (c) show example images for the refractive cylinder and sphere (yellow), respectively, where SfM could not converge without filtering RFs using our method. Detected features are shown in blue crosses, with features identified as refracted shown in red circles.

using the scale. An OptiTrack system was used for ground truth camera pose. Fig. 9a shows example pose trajectories reconstructed by SfM for a filtered and unfiltered LF sequence with the ground truth. The filtered trajectory had a more accurate absolute pose over the entire sequence of images. Fig. 9b and 9c show the relative instantaneous pose error e_i, computed as $e_i = |(\boldsymbol{p_{s,i}} - \boldsymbol{p_{s,i-1}}) - (\boldsymbol{p_{g,i}} - \boldsymbol{p_{g,i-1}})|^2$ for image i, in translation and rotation. Although e_{rot} was similar $< 0.02^\circ$, e_{tr} had larger errors up to 10 mm more than the filtered case. This suggested that filtering for RFs yielded more accurate pose estimates from SfM.

In Table II, we show filtering RFs leads to an average of 4.28 mm lower e_{tr}, and 0.48° lower e_{rot} relative instantaneous pose errors over 5 LF sequences with different objects, poses and backgrounds, except for Seq. 6, where the number of inlier feature matches for SfM dropped below 50. The number of LFs in each sequence varied, because the unfiltered case could not converge with more images at the end of the sequence (that had higher r). Seq. 7 and 8 shows where only our filtered case converged, so that SfM produced a trajectory for analysis. Thus, filtering RFs using our method yielded more consistent (non-refractive) features that improved the accuracy of the SfM pose estimates, and made it more robust in the presence of refractive objects.

For the cases where SfM converged in the presence of refractive objects, we created a sparse reconstruction of the scene of Fig. 1, which was primarily the Lambertian background plane, since we attempted to remove points distorted by the cylinder. Sample reconstructions for both the unfiltered and filtered cases are shown in Fig 10. Both point clouds were centered about the origin and rotated into a common frame. For visualization, an overlay of the scene geometry best fit to the background plane is provided. The unfiltered case had



(a)

(b)

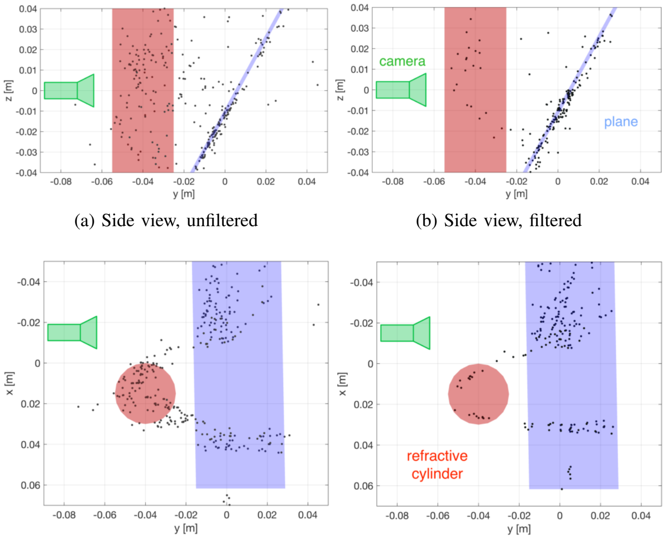
(c)

(d)

Fig. 9. For cases where SfM converged, filtering out the RFs yielded more accurate pose estimates. (a) Sample pose trajectory with the filtered (red) closer to ground truth (blue), compared to the unfiltered case (green). Relative instantaneous pose error for translation (a) and rotation (b) are shown over a sample LF sequence, where the filtered case was consistently lower than the unfiltered case. (c) With our method, the refractive feature ratio for the filtered case was lower than the unfiltered case.

TABLE II
COMPARISON OF MEAN RELATIVE INSTANTANEOUS POSE ERROR FOR UNFILTERED AND FILTERED SFM-RECONSTRUCTED TRAJECTORIES

		Unfiltered			**Filtered**		
Seq.	#LFs	e_{tr} **[mm]**	e_{rot} **[°]**	#inliers	e_{tr}	e_{rot}	#inliers
1	10	18.86	5.72	160	**8.09**	**4.52**	127
2	10	10.45	4.66	285	**7.10**	**4.29**	140
3	10	10.17	4.52	281	**6.94**	**4.09**	186
4	9	11.13	4.70	296	**7.50**	**4.37**	224
5	8	6.07	4.47	201	**5.66**	**4.39**	196
6	10	**6.52**	**0.74**	207	15.21	1.58	50
7	10	N/A	N/A	N/A	**8.51**	**4.02**	155
8	10	N/A	N/A	N/A	**6.95**	**4.16**	230



(a) Side view, unfiltered

(b) Side view, filtered

(c) Top view, unfiltered

(d) Top view, filtered

Fig. 10. For the scene shown in Fig. 1a, (a,c) the unfiltered case resulted in a sparse reconstruction where many points were generated between the refractive cylinder (red) and the background plane (blue). In contrast, (b,d) the filtered case resulted in a reconstruction with fewer such points, and the resulting camera pose estimates were more accurate. The cylinder and plane are shown to help with visualization only. The camera (green) represents the general viewpoint of the scene, not the actual position of the camera.

to be re-scaled according to the scene geometry (as opposed to via the poses done previously) for comparison. Scaling via scene geometry resulted in severely worse pose trajectories for the unfiltered case, although the same observations were made: with our method, there were fewer points placed within the empty space between the refracted object and the plane. This is an important difference since the absence of information is treated very differently from incorrect information in robotics. For example, estimated refracted points might incorrectly fill an occupancy map, preventing a robot from grasping refractive objects.

VI. Conclusions

In this paper, we proposed a method to distinguish refracted features based on a planar fit in 4D and slope consistency. To achieve this, we introduced a novel textural cross-correlation technique to extract feature curves from the 4D LF. Our approach demonstrated higher detection and lower failure rates than previous work for LF camera arrays, and extended the detection capability to lenslet-based LF cameras. For these cameras, slope consistency proved to be a much stronger indicator of distortion than planar consistency. This is appealing for mobile robot applications, such as domestic robots that are limited in size and mass, but will have to navigate and eventually interact with refractive objects. Future work will relate feature slopes to surface curvature to aid grasping.

It is important to note that while we have developed a set of criteria for refracted features in the LF, these criteria are not necessarily limited to refracted features. Depending on the surface, specular reflections may appear as non-linear. Such features are typically undesirable, and so we retain features that are strongly Lambertian, and thus good candidates for matching, which ultimately leads to more robust robot performance in the presence of refractive objects.

In our experiments, we have shown that our method can exclude refracted features in a scene containing spherical and cylindrical objects; however, it is likely that not all planar objects, such as windows, would be detected by our method. Some types of glass with a consistent refractive index may not be detected by our method because they do not significantly distort the LF by design. However, features that pass through curved surfaces or inconsistent refractive indexes, such as those commonly seen through privacy glass and stained glass windows, should be detected based on the nonlinearities created by the distortions of the object.

Finally, in this paper, we explored the effect of removing the refractive content from the scene. We demonstrated that rejecting refracted features for monocular SfM yields lower reprojection errors and more accurate pose estimates. In future work, we plan to exploit the refractive content for robot motion and refractive shape recovery.

References

[1] C. C. Kemp, A. Edsinger, and E. Torres-Jara, "Challenges for robot manipulation in human environments," *IEEE Robotics Automation Magazine*, vol. 14, no. 1, 2007.

[2] I. Ihrke, K. Kutulakos, H. Lensch, M. Magnor, and W. Heidrich, "Transparent and specular object reconstruction," *Computer Graphics Forum*, vol. 29, no. 8, 2010.

[3] D. G. Dansereau, "Plenoptic signal processing for robust vision in field robotics," Ph.D. dissertation, University of Sydney, 2014.

[4] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, 2003.

[5] Y. Wei, L. Kang, B. Yang, and L. Wu, "Applications of structure from motion: a survey," *J. of Zhejiang University*, vol. 14, no. 7, 2013.

[6] M. Ben-Ezra and S. K. Nayar, "What does motion reveal about transparency," in *ICCV*, 2003.

[7] O. Johannsen *et al.*, "A taxonomy and evaluation of dense light field depth estimation algorithms," in *CVPR Workshop*, 2017.

[8] S. Wanner and B. Golduecke, "Reconstructing reflective and transparent surfaces from epipolar plane images," *German Conf. on Pattern Recognition*, vol. 8142, 2013.

[9] K. Maeno, H. Nagahara, A. Shimada, and R. Taniguchi, "Light field distortion feature for transparent object recognition," in *CVPR*, 2013.

[10] Y. Xu, H. Nagahara, A. Shimada, and R. Taniguchi, "Transcut: Transparent object segmentation from a light-field image," *ICCV*, 2015.

[11] D. Tsai, D. Dansereau, T. Peynot, and P. Corke, "Image-based visual servoing with light field cameras," *RA-L*, vol. 2, no. 2, January 2017.

[12] J. A. Teixeira, C. Brites, F. Pereira, and J. Ascenso, "Epipolar based light field key-location detector," in *Multimedia Signal Processing*, 2017.

[13] O. Johannsen, A. Sulc, and B. Goldluecke, "On linear structure from motion for light field cameras," in *ICCV*, 2015.

[14] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH*. ACM, 1996.

[15] E. W. Weisstein, "Hyperplane," 2017. [Online]. Available: http://mathworld.wolfram.com/Hyperplane.html

[16] I. Tosic and K. Berkner, "3D keypoint detection by light field scale-depth space analysis," in *ICIP*, 2014.

[17] R. Bolles, H. Baker, and D. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *IJCV*, vol. 1, no. 7, 1987.

[18] M. Levoy, "The (new) stanford light field archive," 2008. [Online]. Available: http://lightfield.stanford.edu/lfs.html

[19] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, 2004.

[20] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Decoding, calibration and rectification for lenselet-based plenoptic cameras," in *CVPR*, 2013.

[21] S. Wanner and B. Goldeluecke, "Variational light field analysis for disparity estimation and super-resolution," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, 2014.

[22] J. Schoenberger and J.-M. Frahm, "Structure-from-motion revisited," *CVPR*, 2016.