Unsupervised Learning of Depth Estimation and Visual Odometry for Sparse Light Field Cameras

S. Tejaswi Digumarti* †,§, Joseph Daniel* †, Ahalya Ravendran†,§, Ryan Griffiths†,§, Donald G. Dansereau†,§

***Abstract*— While an exciting diversity of new imaging devices is emerging that could dramatically improve robotic perception, the challenges of calibrating and interpreting these cameras have limited their uptake in the robotics community. In this work we generalise techniques from unsupervised learning to allow a robot to autonomously interpret new kinds of cameras. We consider emerging sparse light field (LF) cameras, which capture a subset of the 4D LF function describing the set of light rays passing through a plane. We introduce a generalised encoding of sparse LFs that allows unsupervised learning of odometry and depth. We demonstrate the proposed approach outperforming monocular, stereo and conventional techniques for dealing with 4D imagery, yielding more accurate odometry and depth maps and delivering these with metric scale. We anticipate our technique to generalise to a broad class of LF and sparse LF cameras, and to enable unsupervised recalibration for coping with shifts in camera behaviour over the lifetime of a robot. This work represents a first step toward streamlining the integration of new kinds of imaging devices in robotics applications.**

I. INTRODUCTION

Integrating new imaging devices into robotics applications is a skilled and challenging task. While an exciting variety of new imaging capabilities is emerging, dealing with calibration, compensating for non-idealities, and interpreting new forms of visual information have historically been time-consuming. This has limited the uptake of new visual sensors in robotics.

Emerging imaging technologies could allow robots to see better in a range of scenarios. Recent capabilities include imaging around corners, directly observing light propagation, adaptive and long-range depth sensing, and imaging through occluders like rain, snow, and fog using light field (LF) cameras [1]–[5]. Before these technologies can be used in robotics we must find ways of dealing with their unique characteristics.

In this work, we take a step towards the automated interpretation of new cameras by adapting unsupervised learning techniques to deal with sparse LF cameras. LF cameras in general have been shown to offer improved performance in low light and underwater, and by simplifying conventionally complex tasks like visual odometry and change detection [4]–[7]. A full LF camera captures a regular grid of views, yielding a 4D image that encodes the behaviour of light in terms of both ray position and direction. A sparse LF camera like the one shown in Fig. 1 captures a subset of these views. This has been shown to offer many of the same advantages as LFs [8], [9] while using a fraction of the imaging bandwidth. It does however require more sophisticated algorithms for carrying out tasks like visual odometry.

Fig. 1. We propose an unsupervised approach to interpret new imaging devices like the EPIModule from EPIImaging LLC shown here. We propose a novel encoding scheme that benefits from the view diversity of these devices while allowing a broad family of cameras to be used without manual intervention. To demonstrate the technique we learn visual odometry and depth estimation, delivering metric results with greater accuracy and detail than prior approaches.

To make sense of these cameras, we leverage recent work in unsupervised learning that shows how using prediction as a training signal one can learn useful tasks without need for costly labelled data [10]–[12]. We show how to generalise this idea to estimate odometry and depth from sparse LF cameras, as a first step toward automating the interpretation of general imaging devices for robotics applications.

Our key contributions are:

- We generalize unsupervised odometry and depth estimation to operate on sparse 4D LFs;
- We introduce an encoding scheme for sparse LFs appropriate to odometry and shape estimation, show it outperforming naïve LF stacking and focal stack approaches,

* Authors contributed equally to the work presented.

†Tejaswi Digumarti, Joseph Daniel, Ahalya Ravendran, Ryan Griffiths and Donald Dansereau are with the School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, 2006 NSW, Australia. joe.daniel@outlook.com.au , arav3215@uni.sydney.edu.au

§Tejaswi Digumarti, Ahalya Ravendran, Ryan Griffiths and Donald Dansereau are with the Sydney Institute for Robotics and Intelligent Systems, 2006 NSW, Australia. tejaswi.digumarti, donald.dansereau@sydney.edu.au

and evaluate using full vs. partial LF reconstruction as a training signal; and
- We demonstrate the proposed methods outperforming monocular and stereo approaches, yielding more accurate trajectories and depth maps, with known scale.

To validate our method we mounted an EPIModule from EPIImaging, LLC on a UR5e robotic arm, as shown in Fig. 1. We collected 46 trajectories in a variety of indoor scenes, yielding 8298 LFs, each with 17 views, and all with accurately known poses as enabled by the robotic arm. We are releasing all data and code along with the paper[1].

To evaluate our method we compare against monocular and stereo approaches, and more conventional LF-based stacking and focal stacking methods. We also compare two prediction modes, one dealing only with the central LF view, and the other reconstructing the entire LF, drawing on a prior estimate of the inter-camera spacing and camera geometry. We show the proposed methods outperform monocular, stereo and naïve LF-based approaches in terms of visual odometry accuracy, depth estimation and qualitative 3D level of detail. We also report training and inference times, showing that the feasibility of our method for practical robotic applications.

Our approach captures both the geometric and textural information present in sparse LFs, and we expect it to work well for other types of cameras with regular overlapping views. This includes regularly spaced 1D and 2D camera arrays, sparse cameras like the EPIModule, and lenslet-based plenoptic cameras like the Lytro and Raytrix devices. A robot equipped with this capability could swap cameras in and out, requiring only an unsupervised training period to adapt to new imaging hardware. We anticipate this to be of interest in evaluating new devices and sensor placements in practical applications.

Limitations: Although our method is unsupervised and does not require calibration, metric pose and depth require an estimate of the camera layout, including the distances between camera lenses. The module employed in this work conducts onboard distortion correction, but we anticipate the method would work without this feature. Importantly, the learning process can be employed over the lifespan of a robot, allowing it to automatically adapt to shifts in extrinsic or intrinsic camera properties associated with temperature fluctuations, vibrations and fatigue.

II. RELATED WORK

LF cameras encode light in a 4D structure that captures light's behaviour in terms of both ray position and direction [13]. These conventionally parameterise light rays in terms of their points of intersection with two reference planes: an s,t plane, close to the camera array, captures ray position. A second u,v plane placed at an arbitrary distance D and parallel to the first captures ray direction. The combination s,t,u,v uniquely identifies a pixel measured by an LF camera, and a ray in world space.

[1]https://roboticimaging.org/Projects/LearnLFOdo/

LFs have been shown to improve imaging performance in challenging conditions [5], [14] and to simplify a range of tasks, offering effective and sometimes closed-form solutions to both depth estimation and visual odometry [6], [15], [16].

Sparse LFs capture much of the same information [8], [9], but interpreting their imagery is less obvious, and different configurations offer different tradeoffs in robustness and the algorithmic complexity required for interpretation. We take the sparse LF camera as representative of a class of cameras with overlapping and redundant views. As a step towards demonstrating general autonomous interpretation of newly developed imaging devices, we show how unsupervised learning can be adapted to deal with this class of cameras. We anticipate that the proposed method applies to linear camera arrays, combinations of linear arrays as in the EPIModule shown in Fig. 1, and full LFs as captured by arrays and lenslet-based cameras.

The use of unsupervised learning has recently emerged as a means of accomplishing complex tasks by cleverly combining prior knowledge of the problem and use of prediction as a feedback mechanism. This has been used to predict camera motion and depth from both monocular and stereo cameras [10]–[12], [17]–[20]. These works typically separate the problem into two parts: pose estimation and depth estimation, each handled by a separate network. In the case of stereo cameras some prior knowledge of the camera setup, generally the inter-aperture spacing, is used to obtain metric results [11], [12]. We draw inspiration from this work and extend it to handle new kinds of cameras.

While prior work has established how to handle monocular and stereo inputs, it is less obvious how one should operate on 4D LFs, let alone sparse versions of the same. Getting this kind of data into a 2D convolutional neural network (CNN) is not obvious. Previous work has sliced the 4D LF and concatenated the resulting 2D slices into stacks [21], superimposed multiple shifted variations of these 2D slices into focal stacks [22] and used a combination of the two [23]. More sophisticated approaches have sliced in different pairs of dimensions, or interleaved slicing strategies [24].

In this work we adopt previous work applying machine learning to LFs, and extend them by slicing and concatenating in multiple dimensions, offering the network a mixture of forms of information. We slice in the textural (u,v) dimensions, capturing scene appearance, as well as *epipolar* dimensions (s,u and t,v), capturing scene geometry. We further build on this by applying a layer of convolutional features to the stacked epipolar slices, similar to the approach used for depth estimation by Shin et al. [25]. This offers the ability to extract salient geometric features prior to estimating depth and pose. It also has the added advantage of placing the two forms of information, textural and epipolar, in a similar space, facilitating learning. Note that our approach allows the use of conventional 2D CNNs, and while some work has generalised to using 3D or even 4D convolutions [26], this can be more computationally expensive and loses the ability to exploit existing 2D architectures.

To build networks that estimate depth and pose, we leverage prior work that applied a hand-crafted but differentiable warping function to estimate a future image from a prior image, depth map, and relative pose [10]. As in previous approaches employing stereo imagery [12], we consider all input images in the warping process, yielding an estimated LF with the same dimensions as the input LFs. This requires a generalisation of the 2D warping function to operate on LFs, which we present here along with a comparison to the more naive single-view approach.

III. METHODS

A. Dataset

LF images were collected using an EPIModule from EPIImaging, LLC, mounted on a robotic arm, while executing 46 trajectories. Ground truth poses were recorded for evaluation. The EPIModule captures images from 17 sub-apertures arranged in a plus sign pattern, as shown in Fig. 1. The captured images were rectified using off-the-shelf rectification enabled within the module and downsampled to a size of 256×192 pixels. A central crop of 224×160 pixels was then taken from all the images. The dataset is split into 37 trajectories for training, 6 trajectories for validation and 3 trajectories for testing. The test split also contains objects not present during training and validation.

B. Network Architecture

In keeping with the aim of extending existing unsupervised learning approaches for depth and pose estimation from monocular and stereo images to LFs, we draw inspiration from the network architecture presented in [10]. The three main components of the network are the following.

- A single view depth estimation network that predicts per pixel depth for an input image. We use the encoder-decoder architecture of 'DispNet' [27], with skip connections, as the depth estimation network.
- A multi-view pose estimation network (similar to [10], [19], [28]) that takes as input images from two nearby viewpoints and estimates the relative pose of one viewpoint with respect to the other.
- A differentiable warp module that couples the depth and pose estimation networks by minimizing a loss based on view synthesis. This loss is the photometric error between a target image and a reference image warped to the viewpoint of the target, using the estimated depth and pose.

Generalizing the architecture to sparse LFs poses the question of how best 4 dimensional data can be arranged as 2 dimensional slices, so that it forms an informative input to convolutional neural networks that predict depth and pose. Prior work [24] approached this challenge with the assumption that a complete grid of 2D images is available. However, the specific configuration of the apertures in the imaging module used in this work (see Fig. 1), prohibits doing so without introducing significant redundant data. Therefore, we address this challenge by proposing a novel *encoding scheme* that captures both geometric and textural information from the LF. The complete network architecture with the encoders is shown in Fig. 2.

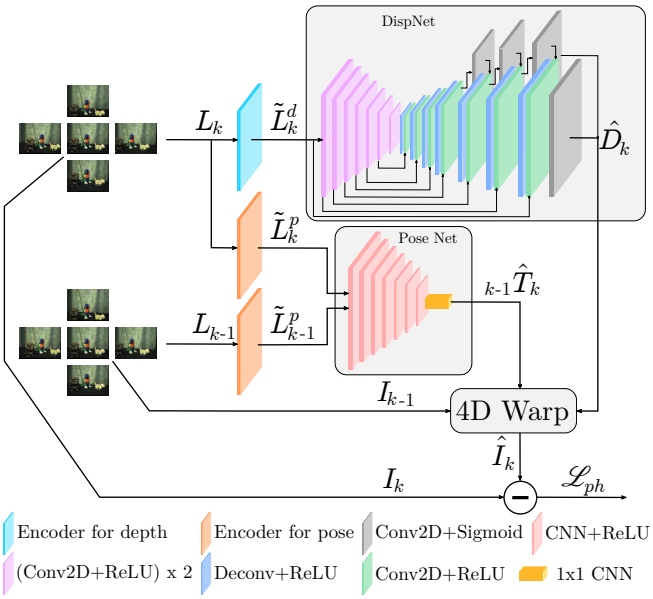


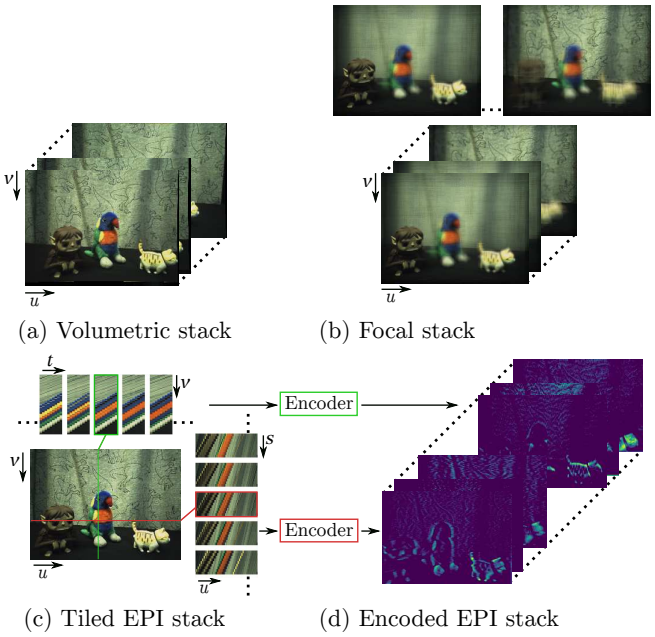
Fig. 2. Proposed architecture: Encoders convert each sparse 4D input LF L into a form $\tilde{L}^d$ and $\tilde{L}^p$ ingestible by 2D CNNs. From the encoded $\tilde{L}^d_k$, the depth network estimates per-pixel depth D_k, and from $\tilde{L}^p$ the pose network estimates the pose ${}_{k-1}\hat{T}_k$ of the camera from time $k-1$ to time k. The two networks drive a differentiable warp that predicts an LF $\hat{I}_k$ by warping I_{k-1} to time k. The photometric loss $\mathscr{L}_{ph}$ between the true and estimated LFs drives training of the networks and encoders. We evaluate different encoding schemes, and use of 2D vs. 4D warping and photometric loss.

C. Sparse LF Image Encodings

In order to motivate the choice of the proposed encoding scheme, we first present two approaches of stacking LFs as 2D slices in the textural dimension followed by the proposed approach where slicing is performed in the epipolar dimensions. The three encoding schemes are illustrated in Fig. 3.

1) Volumetric Stack: A volumetric stack is obtained by stacking images of the sparse LF along the (u,v) direction, i.e. the colour channel dimension. This was proposed and evaluated in [24], for material classification. With this encoding, we expect a CNN to learn features related to parallax, occlusion and depth. For a camera array with N sub-apertures and an image size of $(H \times W)$ pixels, the volumetric stack has a size of $(N \times H \times W)$ pixels.

2) Focal Stack: Superimposing images from each sub-aperture and averaging the intensity at each pixel results in an image where some regions exhibit interference and are 'out-of-focus', while other regions are 'in-focus' and remain crisp. Shifting pixels of different sub-apertures by varying amounts prior to superposition, results in images that are in focus at different distances from the camera. Stacking superimposed images, with different planes of focus, along the colour channel dimension constitutes the focal stack. As the focal stack encodes depth in the form of interference at each region, we expect a CNN to use this information to estimate depth and pose [22]. However, the trade-off is that aliasing artefacts in the focal stack may affect training. For



(a) Volumetric stack (b) Focal stack

(c) Tiled EPI stack (d) Encoded EPI stack

Fig. 3. Encodings of a sparse LF. (a) Volumetric stacking of images along the colour dimension, (b) Superimposed images in-focus at varying distances from the camera (notice the change in focus from the object in the foreground to the curtain in the background) and the corresponding focal stack, (c) the horizontal and vertical tiling of EPIs and (d) the *proposed* encoding scheme where the horizontal and vertical tiled EPIs pass through a single CNN layer encoder. The resultant feature maps are stacked to form the proposed encoded EPI stack.

N planes of focus and an image size of $(H \times W)$ pixels, the focal stack has a size of $(N \times H \times W)$ pixels.

3) Tiled Epipolar Plane Image Stack: An EPI [29] is a slice of the LF in the s,u or t,v direction and encodes depth and occlusion information in the slope of the sheared lines in the image. In order to utilize this information, we propose tiling these images, vertically for the s,u slices and horizontally for the t,v slices, as shown in Fig. 3c. For a camera array with N sub-apertures, such a tiling results in images that are *tall* ($(N \cdot W \times H)$ pixels) and *wide* ($(W \times N \cdot H)$ pixels) respectively.

Passing these tiled images as input to the depth and pose estimation networks is not trivial. Instead of modifying the base architecture to accept these images as inputs, we propose the use of an additional convolutional layer on the tiled EPIs, that downsamples them to the shape expected by the networks. Both the wide and tall tiled EPIs are convolved with a kernel of size $(N \times N)$, but with a horizontal stride of N for the wide tiled EPI and a vertical stride of N for the tall tiled EPI. This is followed by a Rectified Linear Unit (ReLU) activation layer. The resulting encoded EPIs are stacked along the colour channel to form the *Encoded EPI stack*, as shown in Fig. 3d.

We input the encoded EPI directly to the depth estimation network. However, for the pose estimation network, we additionally concatenate images from the volumetric stack. We hypothesize that by stacking slices in the epipolar dimension with slices in the textural dimension, the pose estimation network can leverage both structural and semantic features from the different input spaces during the learning process. On the other hand, to improve the ability of the depth network to generalize to unseen objects and not rely on global features in the image, such as the background and the table, we omit the slices in the textural dimension in its input.

D. Loss Formulation

Given a pair of successive viewpoints, with indices $(k-1,k)$, from which LFs (L_{k-1},L_k) are captured, we compute the corresponding encoded LFs $\left(\tilde{L}_k^d,\tilde{L}_{k-1}^p,\tilde{L}_k^p\right)$ as described earlier, where the superscripts d and p indicate the specific encoding for the depth and pose estimation networks respectively. With $\tilde{L}_k^d$ as input, the depth estimation network outputs the pixel-wise depth map[2], $\hat{D}_{c,k}$. We interpret this as the depth map of the central sub-aperture, c, of the imaging module corresponding to the LF at viewpoint k. Next, given $\tilde{L}_{k-1}^p$ and $\tilde{L}_k^p$ as inputs the pose estimation network outputs the relative pose ${}_{c,k-1}\hat{T}_{c,k}$, which we interpret as the relative pose of the central sub-aperture of the imaging module at viewpoint k with respect to the same sub-aperture at viewpoint $k-1$.

Photometric consistency loss can then be computed as

$$\mathscr{L}_{ph} = \frac{1}{n}\sum_{i}^{n} |I_{c,k}(i) - \hat{I}_{c,k}(i)|, \tag{1}$$

where i is the index over the pixel coordinates, n is the number of pixels in the image, $\hat{I}_{c,k}$ is the image $I_{c,k-1}$ of the central sub-aperture at viewpoint $k-1$ warped to the viewpoint k.

The warped image, $\hat{I}_{c,k}$, is computed by sampling the image $I_{c,k-1}$ with the projected homogeneous pixel coordinates $\mathbf{p}_{c,k-1}$ using differentiable bilinear sampling [30].

The projected homogenous pixel coordinates $\mathbf{p}_{c,k-1}$ are computed from the homogeneous pixel coordinates $\mathbf{p}_{c,k}$ as

$$\mathbf{p}_{c,k-1} \sim K \; {}_{c,k-1}\hat{T}_{c,k} \; \hat{D}_{c,k} \; K^{-1} \; \mathbf{p}_{c,k}, \tag{2}$$

where K is the matrix of intrinsic parameters of the central sub-aperture. This is consistent with the approach presented in [10].

Additionally, we also employ the multi-scale smoothness loss [10], [19], [20], [31], to overcome the issue of poor training in low texture regions. Through empirical evaluation, we also found that replacing the smoothness loss with a loss based on total-variation error [32] after a few iterations of training helped reduce noisy estimates of depth while preserving edges, especially in low-texture regions. The total loss was thus a weighted sum of the individual loss terms.

E. Single-warp versus Multi-warp Reconstruction

We refer to the pipeline described thus far as the *Single-warp* reconstruction pipeline. We highlight that the single-warp pipeline requires only the knowledge of the intrinsic

[2]The network actually estimates inverse depth. For improving readability we omit this technicality.

camera parameters of the sub-apertures and does not rely on the arrangement of the individual sub-apertures within the imaging module. However, this pipeline suffers from the issue of scale ambiguity.

We address this issue by taking into account the additional information available when imaging a scene using a camera array. Instead of using the photometric warp to reconstruct the image of a single sub-aperture, the pipeline is modified to reconstruct the LF. Unlike the previous case, the depth estimation network outputs the depth of M sub-apertures instead of a single sub-aperture, while the pose estimation network remains unaltered. The photometric loss is now computed between the corresponding M sub-apertures across the viewpoints, and the total loss is the mean of the individual losses.

Therefore equations (1) and (2) can be modified as

$$\mathscr{L}_p = \frac{1}{M \cdot n} \sum_{j \in M} \sum_{i}^{n} |I_{j,k}(i) - \hat{I}_{j,k}(i)|, \tag{3}$$

and

$$p_{j,k-1} \sim K \; {}_cT_j^{-1} \; {}_{c,k-1}\hat{T}_{c,k} \; {}_cT_j \; \hat{D}_{j,k} \; K^{-1} \; p_{j,k}, \forall j \in M \tag{4}$$

When j represents the index of a non-central sub-aperture then ${}_cT_j$ is the pose of a non-central sub-aperture relative to the central sub-aperture. In this work, we assume that this transformation is known, up to a scale factor, and is constant for all the sub-apertures. However, as stated earlier, this assumption may be relaxed and in turn be predicted using using a photometric consistency constraint similar to (1) imposed between sub-apertures of the same viewpoint.

One can see that this formulation of the photometric error in (3) penalizes an incorrect estimate of scale because an error in the depth estimate results in a large photometric error for the other sub-apertures. We call the reconstruction pipeline with this modification the *Multi-warp* reconstruction pipeline.

IV. RESULTS

A. Implementation Details

We evaluate three different encodings of the sparse LF as described in Sec. III-C. We consider two variants of the focal stack, one with 5 planes of focus (coarser spacing between the planes) and one with 9 planes of focus (finer spacing between the planes). We refer to the two configurations as focalstack-5 and focalstack-9 respectively. Furthermore, we compare the performance of all the encoding schemes and reconstruction pipelines against the monocular depth and pose estimation approach from [10], and a stereo based approach from [12] utilizing only photometric and smoothness losses.

For the proposed encoding scheme images from central sub-aperture and its 4 closest sub-apertures constitute the volumetric (u,v) stack that is concatenated with the EPI feature maps. These sub-apertures are also used for computing the photometric error in the multi-warp reconstruction pipeline.

All the networks were trained for 100 epochs, with weights initialized from a Xavier uniform distribution. The Adam [33] optimizer was used during training with a momentum of 0.9 and β of 0.999. We weigh down the smoothness and the total variation loss by a factor of 0.3.

B. Depth Estimation

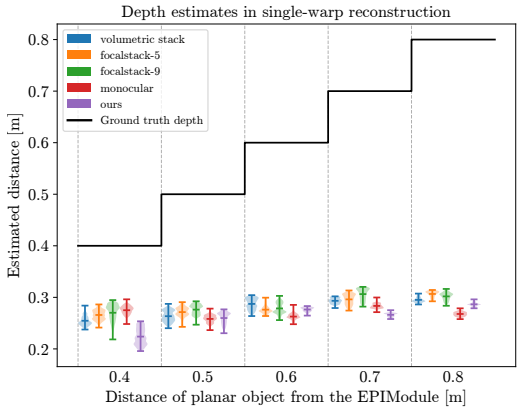


Fig. 4. Estimates of depth in the single-warp pipeline for a planar object placed at multiple distances from the imaging module. The single-warp pipeline suffers from scale ambiguity and as a result the estimates are far from the ground truth depth (black).

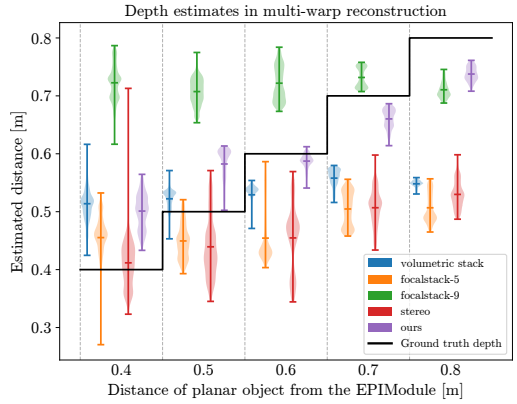


Fig. 5. Estimates of depth in the multi-warp pipeline for a planar object placed at multiple distances from the imaging module. The network trained with the proposed encoding scheme (purple) is able to estimate depths that are reasonably close to the ground truth depth (black), while with the other encoding schemes the network struggles to learn accurate scale. During training most objects were placed 0.4-0.7m away from the imaging module, and scenes at 0.8m yield reasonable results despite being outside this range.

1) Quantitative Evaluation: To evaluate the accuracy of the depth estimates, we placed a planar object, with a random texture on it, at multiple known distances away from the EPIModule, fronto-parallel to the imaging module. Depths estimated for each of the encodings was compared against the ground-truth depth. These are presented in Tab. I and illustrated in Fig. 4 and Fig. 5. Due to the aforementioned ambiguity of scale in the monocular and single-warp reconstruction pipelines, the estimated depth is far from the actual value (see Fig. 4), despite some networks being able to estimate qualitatively good shape (see Fig. 6). On the other hand while the multi-warp pipeline improves overall scale estimate, only our method shows a trend that follows the

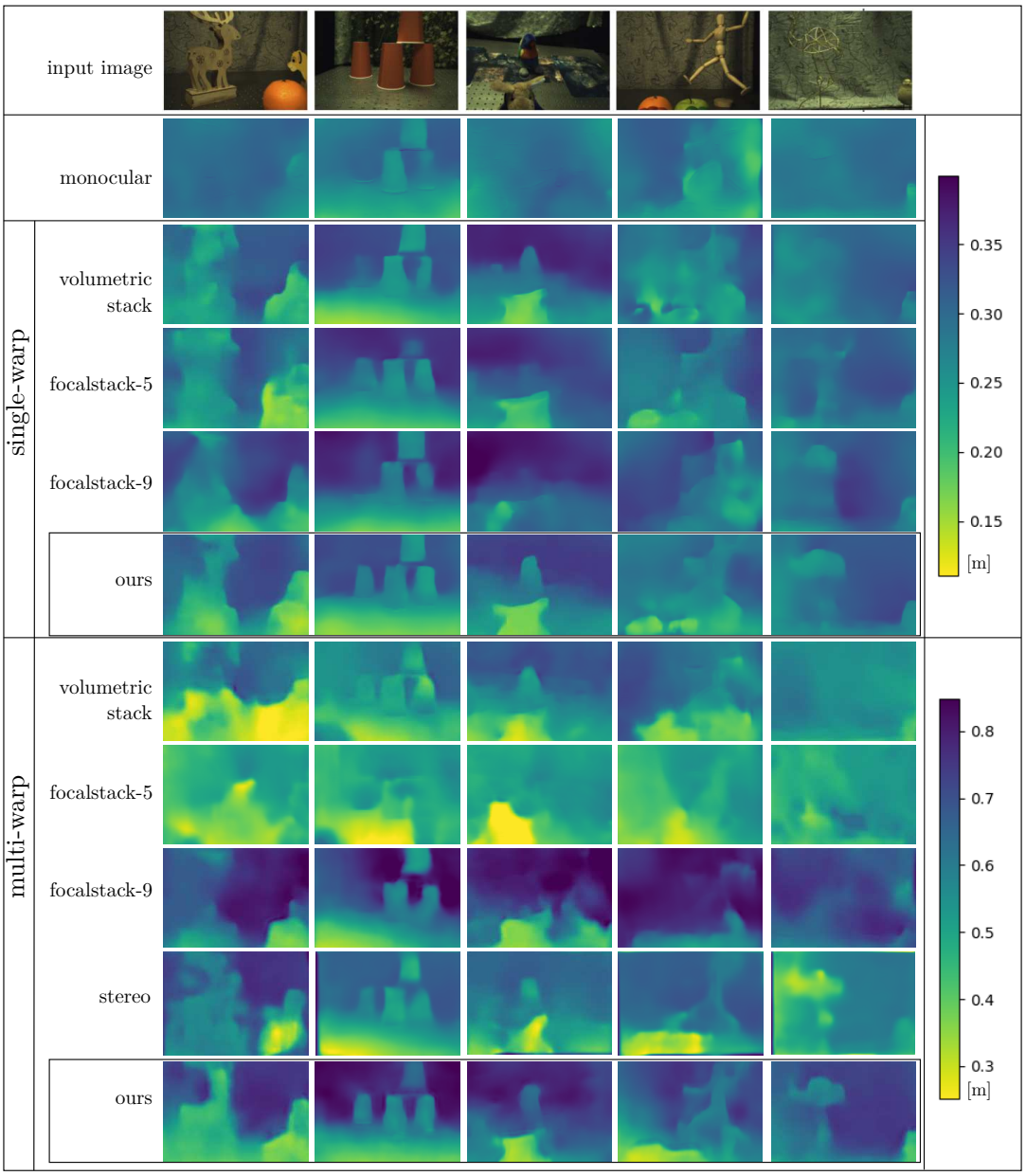


Fig. 6. Depth estimates for a few representative images of the dataset for the different input encodings: The network trained with the proposed Encoded Tiled EPI stack outperforms the networks with other encoding schemes and monocular and stereo depth estimates resulting in better shape estimation and 3D level of detail. It is able to generalize well to previously unseen objects (column 4) and distinguish thin structures (columns 4 and 5), where the other encodings struggle. Note that the colour scales are different between the single-warp and multi-warp pipelines and were chosen for clear visualization.

true depth values, albeit with some error at close and far distances from the module. Note that during training most scene content was 0.4-0.7 m from the imaging module, but the proposed encoding generalizes somewhat beyond this range as seen in Fig. 5.

2) Qualitative Evaluation: Depth estimates for representative images of the test dataset are shown in Fig. 6. For both the warp pipelines, the network trained with the proposed encoding outperforms the other encoding schemes, with better shape estimation and greater level of detail in the depth estimates. Our encoding also enables the network to estimate the overall shape of challenging and thin structures (last two columns in Fig. 6) and at the same time generalize well to previously unseen objects (column 4), while the other methods, except stereo, fail to even detect the object. This clearly shows the advantage of incorporating both textural and geometric information in the encoding.

C. Pose Estimation Results

We evaluate pose estimation by computing the frame-to-frame Relative Pose Error (RPE) [34] between the estimated relative pose of successive camera frames and the ground truth relative pose of the same frames. The evaluation was performed on the test split, the results of which are summarized in Tab. II. Our encoding scheme outperforms all the

TABLE I

QUANTITATIVE EVALUATION OF DEPTH ESTIMATES: THE NETWORK WITH THE PROPOSED ENCODING SCHEME ESTIMATES VALUES CLOSE TO GROUND TRUTH IN THE MULTI-WARP RECONSTRUCTION PIPELINE

Method		Distance of planar object from the EPIModule [m]										Overall RMSE [m]
		0.4		0.5		0.6		0.7		0.8		
		Mean	Std.dev	Mean	Std.dev	Mean	Std.dev	Mean	Std.dev	Mean	Std.dev	
Monocular		0.275	0.008	0.258	0.007	0.263	0.006	0.283	0.004	0.268	0.004	0.359
Single	volumetric stack	0.255	0.009	0.263	0.011	0.287	0.011	0.293	0.005	0.295	0.005	0.345
	focalstack-5	0.266	0.010	0.271	0.010	0.276	0.007	0.296	0.010	0.307	0.006	0.341
	focalstack-9	0.270	0.016	0.276	0.009	0.279	0.011	0.306	0.011	0.302	0.007	0.339
	ours	0.224	0.016	0.260	0.011	0.276	0.004	0.266	0.003	0.286	0.004	0.359
Multi	volumetric stack	0.514	0.031	**0.522**	0.018	0.529	0.019	0.558	0.016	0.548	0.006	0.143
	focalstack-5	0.455	0.040	0.449	0.030	0.454	0.044	0.505	0.029	0.507	0.025	0.174
	focalstack-9	0.723	0.031	0.707	0.024	0.722	0.030	**0.732**	0.017	0.711	0.014	0.185
	stereo	**0.412**	0.046	0.439	0.051	0.455	0.057	0.507	0.032	0.530	0.023	0.164
	ours	0.501	0.030	0.582	0.027	**0.587**	0.012	0.660	0.017	**0.738**	0.012	**0.067**

TABLE II

FRAME-TO-FRAME RELATIVE POSE ERROR: THE NETWORK WITH OUR PROPOSED ENCODING SCHEME OUTPERFORMS OTHER ENCODING SCHEMES, AND MONOCULAR POSE ESTIMATES IN TERMS OF BOTH TRANSLATION AND ROTATION ERROR

Method		Relative Pose Error in Translation [m]			Relative Pose Error in Rotation [deg]		
		Mean	Std. dev.	RMSE	Mean	Std. dev.	RMSE
Monocular		0.029	0.016	0.033	1.522	0.969	1.808
Single	volumetric stack	0.028	0.015	0.032	1.453	0.880	1.703
	focalstack-5	0.030	0.015	0.033	1.439	0.883	1.693
	focalstack-9	0.030	0.016	0.034	1.452	0.912	1.716
	ours	**0.026**	0.016	**0.031**	**1.308**	0.885	**1.583**
Multi	volumetric stack	0.024	0.016	**0.029**	1.366	0.802	1.585
	focalstack-5	0.031	0.016	0.035	1.457	0.779	1.653
	focalstack-9	0.035	0.017	0.039	1.585	0.868	1.807
	stereo	0.028	0.015	0.032	2.413	1.655	2.938
	ours	**0.023**	0.017	**0.029**	**1.282**	1.311	**1.534**

TABLE III

TRAINING AND VALIDATION TIMES

Method		Training time [hr]	Validation time [ms] (mean over 300 inferences)	
			Data Loading	Inference
Single	monocular	17.6	1.9	7.9
	volumetric stack	17.6	1.9	39.6
	focalstack-5	25.0	1.9	51.9
	focalstack-9		1.9	61.3
	ours	17.9	10.8	48.1
Multi	volumetric stack	25.4	9.1	47.1
	focalstack-5	30.4	9.1	59.2
	focalstack-9		9.0	69.0
	stereo	3.5	3.8	12.1
	ours	29.5	18.0	55.8

other approaches in terms of both translation and rotation error. This is true for both of the warp pipelines. We further see that due to a better estimate of scale, multi-warp exhibits better performance than single-warp, as was expected.

D. Training and Validation times

We report training and validation times in Tab. III. Our method does not require prohibitively large amounts of training time in comparison to the monocular approach. Networks were trained on NVIDIA RTX 2080 and NVIDIA V100 GPUS, except for the stereo approach. This was trained on an NVIDIA RTX 3060 GPU and hence exhibits significantly faster training times. We also anticipate a speed up by four times through code optimization. Inference times were measured on an NVIDIA RTX 2080 GPU. Inference takes longer time for our method due to time spent on encoding the LF. However, despite our approach using 16 times more data than the monocular approach, the inference times are within practical limits for deployment on a robotic platform.

V. CONCLUSIONS

We have presented an approach for adapting existing techniques developed for traditional cameras to novel imaging devices. We show that by incorporating ideas from plenoptic

imaging and unsupervised learning one can successfully estimate depth and odometry from sparse LF cameras which outperforms the state-of-the-art monocular and stereo reconstruction pipelines.

We anticipate follow-on work in generalising to irregularly sampled LFs, as well as to other modalities like event-based cameras, and multi-modal sensing incorporating inertial measurements. As the approach effectively allows ongoing lifelong calibration, adapting to how cameras change over time, the presented work can be extended to allow an autonomous car to adapt to optical shifts due to thermal warping, vibration, or fatigue, or for an underwater robot to adapt to changes in index of refraction or housing deformation associated with temperature, salinity, or pressure shifts.

ACKNOWLEDGMENT

We would like to thank EPIImaging LLC and the University of Sydney Aerospace, Mechanical and Mechatronic Engineering FabLab for their support. We acknowledge the University of Sydney HPC service for providing HPC resources that have contributed to the results reported within this paper.

REFERENCES

[1] M. O'Toole, D. B. Lindell, and G. Wetzstein, "Confocal non-line-of-sight imaging based on the light-cone transform," *Nature*, vol. 555, no. 7696, p. 338, 2018.

[2] D. B. Lindell, M. O'Toole, and G. Wetzstein, "Towards transient imaging at interactive rates with single-photon detectors," in *International Conference on Computational Photography (ICCP)*. IEEE, 2018, pp. 1–8.

[3] J. R. Bartels, J. Wang, W. Whittaker, and S. G. Narasimhan, "Agile depth sensing using triangulation light curtains," in *International Conference on Computer Vision (ICCV)*, 2019, pp. 7900–7908.

[4] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Linear volumetric focus for light field cameras," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 2, p. 15, Feb. 2015.

[5] A. Bajpayee, A. H. Techet, and H. Singh, "Real-time light field processing for autonomous robotics," in *Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4218–4225.

[6] F. Dong, S.-H. Ieng, X. Savatier, R. Etienne-Cummings, and R. Benosman, "Plenoptic cameras in real-time robotics," *Interational Journal for Robotics Research (IJRR)*, vol. 32, no. 2, pp. 206–217, 2013.

[7] D. G. Dansereau, S. B. Williams, and P. I. Corke, "Simple change detection from mobile light field cameras," *Computer Vision and Image Understanding (CVIU)*, vol. 145C, pp. 160–171, 2016.

[8] L. Shi, H. Hassanieh, A. Davis, D. Katabi, and F. Durand, "Light field reconstruction using sparsity in the continuous fourier domain," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 1, p. 12, 2014.

[9] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 193, 2016.

[10] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[11] T. Dharmasiri, A. Spek, and T. Drummond, "ENG: End-to-end neural geometry for robust depth and pose estimation using CNNs," in *Asian Conference on Computer Vision (ACCV)*. Springer, 2018, pp. 625–642.

[12] H. Zhan, R. Garg, C. Saroj Weerasekera, K. Li, H. Agarwal, and I. Reid, "Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction," in *Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 340–349.

[13] M. Levoy and P. Hanrahan, "Light field rendering," in *SIGGRAPH*. ACM, 1996, pp. 31–42.

[14] D. Dansereau, O. Pizarro, and S. Williams, "Linear volumetric focus for light field cameras," *ACM Transactions on Graphics*, vol. 34, pp. 1–20, 03 2015.

[15] T. Leistner, H. Schilling, R. Mackowiak, S. Gumhold, and C. Rother, "Learning to think outside the box: Wide-baseline light field depth estimation with epi-shift," *International Conference on 3D Vision (3DV)*, Sep 2019. [Online]. Available: http://dx.doi.org/10.1109/3DV.2019.00036

[16] E. H. Adelson and J. Y. A. Wang, "Single lens stereo with a plenoptic camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 14, no. 2, pp. 99–106, 1992.

[17] H. Zhan, C. S. Weerasekera, J.-W. Bian, and I. Reid, "Visual odometry revisited: What should be learnt?" in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4203–4210.

[18] V. Anisimovskiy, A. Shcherbinin, S. Turko, and I. Kurilin, "Unsupervised monocular depth estimation CNN robust to training data diversity," in *Canadian Conference on Artificial Intelligence*. Springer, 2020, pp. 36–48.

[19] R. Garg, V. K. Bg, G. Carneiro, and I. Reid, "Unsupervised CNN for single view depth estimation: Geometry to the rescue," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 740–756.

[20] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 270–279.

[21] S. Heber and T. Pock, "Convolutional networks for shape from light field," in *Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3746–3754.

[22] J. Peng, Z. Xiong, D. Liu, and X. Chen, "Unsupervised depth estimation from light field using a convolutional neural network," in *International Conference on 3D Vision (3DV)*. IEEE, 2018, pp. 295–303.

[23] W. Zhou, E. Zhou, G. Liu, L. Lin, and A. Lumsdaine, "Unsupervised monocular depth estimation from light field image," *Transactions on Image Processing*, vol. 29, pp. 1606–1617, 2019.

[24] T.-C. Wang, J.-Y. Zhu, E. Hiroaki, M. Chandraker, A. A. Efros, and R. Ramamoorthi, "A 4D light-field dataset and CNN architectures for material recognition," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 121–138.

[25] C. Shin, H.-G. Jeon, Y. Yoon, I. So Kweon, and S. Joo Kim, "Epinet: A fully-convolutional neural network using epipolar geometry for depth from light field images," in *Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4748–4757.

[26] Á. Faluvégi, Q. Bolseé, S. Nedevschi, V. Dădârlat, and A. Munteanu, "A 3D convolutional neural network for light field depth estimation," in *International Conference on 3D Immersion (IC3D)*, 2019, pp. 1–5.

[27] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, "A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation," in *Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4040–4048.

[28] J. Bian, Z. Li, N. Wang, H. Zhan, C. Shen, M.-M. Cheng, and I. Reid, "Unsupervised scale-consistent depth and ego-motion learning from monocular video," in *Advances in neural information processing systems*, 2019, pp. 35–45.

[29] R. Bolles, H. Baker, and D. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *Intl. Journal of Computer Vision (IJCV)*, vol. 1, no. 1, pp. 7–55, 1987.

[30] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, "Spatial transformer networks," in *Advances in neural information processing systems*, 2015, pp. 2017–2025.

[31] S. Vijayanarasimhan, S. Ricco, C. Schmid, R. Sukthankar, and K. Fragkiadaki, "SfM-Net: Learning of structure and motion from video," *arXiv preprint arXiv:1704.07804*, 2017.

[32] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Computing Research Repository (CoRR)*, vol. abs/1412.6980, 2015.

[34] M. Grupp, "evo: Python package for the evaluation of odometry and SLAM." https://github.com/MichaelGrupp/evo, 2017.