Learning Depth and Visual Odometry From Light Fields
AMME4111 Thesis A Progress Report

Joseph Daniel

November 2019

Contents

Chapter 1

Introduction

Humans have a remarkable ability to perceive visual stimuli emanating from the world around them. Not only do we identify different objects, scene depth, movement, and colour with ease, but we often draw meaning and even enjoyment from the combination of light rays bouncing around space and arriving at our eyes. We are able to automatically compensate for differences in lighting and viewing angle, and we can extract complex information such as emotion from what we process visually. To word it concisely, the human visual perception system is one of the most versatile and robust image processing machines that we know of. Most man-made machines on the other hand are much more easily decomposed into a set of deterministic modules that require explicit, well defined instruction sets. Often, when given an image, there is no straightforward function that directly maps from pixel intensities to the semantic meaning of that image. As researchers tackling problems in computer vision, we are usually interested in developing the algorithms that can help machines not only sense the world, but perceive it in a way that is useful. The objective of this thesis project is to develop a pipeline that simultaneously learns to perform two tasks. The first is visual odometry - the act of estimating camera motion from a sequence of images. The growth of autonomous robotics has meant that there is an ever-growing need for accurate navigation and localisation when satellite positioning is unavailable or wheel encoder odometry is unreliable. An autonomous unmanned aerial vehicle navigating an unknown indoor environment is one example where visual odometry is a valuable feedback signal that can be used to precisely control the UAV's position. The second is depth estimation, that is, discerning the distance of objects in a scene from the imaging device. The ability to perform depth estimation from visual cues is desirable in applications such as autonomous vehicles where navigating new environments whilst avoiding collisions is a basic requirement. Being able to draw depth information from a camera sensor will help to enable these capabilities, especially in applications where methods such as Lidar or RGBD sensors are prohibitive - small quad-copters and internal medical imaging devices are examples where this might be the case.

Humans are exceptionally well adapted to these tasks - our two eyes allow us to process the 3D geometry of the scene, while our learned experiences are often able to fill in the gaps where geometric information is insufficient or unavailable. Unfortunately, image processing algorithms are not equipped with this same kind of human intuition, and so the deceptively complex task of estimating the 3D structure of a scene from a sequence of images continues to attract attention from the computer vision community.

The price of camera components is decreasing while image quality continues to improve, not only making cameras an attractive perception module for autonomous robotics, but also spurring the popularity of multiple-view imaging. Embracing this idea, this thesis project seeks to develop a pipeline that performs these tasks by taking advantage of the rich geometric information exposed when using multiple views. Thus, this thesis project adopts the vocabulary and the concepts from the area of light field imaging. The light field describes the intensities and directions of each light ray in space, providing a convenient way of thinking about multi-view geometry which is embraced throughout this thesis.

Another core objective of this thesis is to develop a pipeline that relies principally on unsupervised machine learning models to perform the two tasks side-by-side. This approach is chosen with the intention that a data-driven model can build a robustness to scenarios that are difficult to design for using hand crafted solutions. Furthermore, this pipeline will benefit from being self-supervised - unlike data-driven models that require direct supervision from labelled data to improve their performance (classifying images of cats and dogs is a common example), this pipeline has no intermediary requirements between the data collection and training routines. Not only does this mean that the expensive process of hand-labelling data is unnecessary for this pipeline to function, but the model will be able to continue learning, benefiting from new data collected in situ. An adaptive perception module that learns to operate effectively in new environments is a desirable capability in robotics, especially when considering that effects such as thermal expansion, vibration and shocks can frequently render our equipment calibrations meaningless in the field.

In this progress report we will first establish the conceptual groundwork that this thesis builds upon with a discussion surrounding the seminal texts that have shaped the state-of-the-art literature in visual odometry and depth estimation. In understanding those texts, we demonstrate that existing methods have yielded excellent results, whilst identifying that there is a substantial potential for improvement on state-of-the-art results using multi-view imaging devices.

Chapter 2

Background

2.1 Geometry in Computer Vision

The pixel is the building block of digital imagery. Thanks to our ability to fabricate advanced circuits on the scale of nanometers, digital camera sensors have become ubiquitous - making their way into all manner of consumer devices. Because the photodiodes in camera sensors are typically arranged on a 2D plane, most of our operations on digital images manipulate the 2D structures present in that pixel data. The question of how a 3 dimensional world comes to form an image on that series of photodiodes is one best explored from the perspective of projective geometry and linear algebra.

2.1.1 The Pinhole Camera

The most basic kind of camera, the pinhole camera projects a 3D world point (X, Y, Z) to a homogenous coordinate on a 2D plane (u, v, w). This relationship is captured in the 3×3 camera intrinsics matrix, and is typically written as

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{2.1}$$

The intrinsics matrix can alternatively be thought of as mapping ray directions to inhomogenous pixel coordinates:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} tan^{-1}(\theta) \\ tan^{-1}(\phi) \\ 1 \end{bmatrix} \tag{2.2}$$
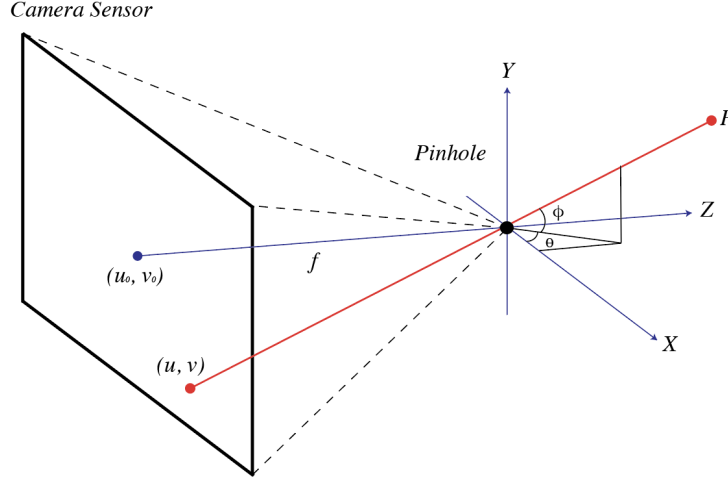
Figure 2.1: The pinhole model describes how a light ray originating from world coordinate P is received at pixel coordinate (u, v). Equivalently, the pinhole model can be thought of as associating the light ray of elevation ϕ and azimuth θ with the specific pixel coordinate (u, v). Since the relationship is a one-to-one mapping between ray directions and pixel coordinates, the inverse of the intrinsics matrix K^{-1} associates each pixel with a corresponding ray direction.

2.1.2   Epipolar Geometry and the Fundamental Matrix

Epipolar geometry describes the relationship between two cameras, and imposes a set of constraints which makes it possible for us to draw meaningful geometric information when we have two views of the same scene. This relationship is encapsulated in a 3×3 matrix called the Fundamental matrix F. The culmination of the epipolar constraint is that any point in 3 dimensional space that appears at pixel $(u_1, v_1, 1)$ in one view and $(u_2, v_2, 1)$ in the second view must satisfy the relationship:

$$\left[u_1, v_1, 1 \right]^T F \left[u_2, v_2, 1 \right] = 0 \tag{2.3}$$

The Fundamental matrix thus describes the rotation and translation between two cameras up to scale. The constraint is easy to visualise, as shown in figure 2.1.2. Any point projected from pixel coordinate $(u_1, v_1, 1)$ must lie on a line that forms by intersecting the epipolar plane with the imaging plane.

2.2   The 4D Light Field

Light field imaging has emerged as a powerful tool in computer vision for robotics, offering a rich higher-dimensional representation than what can be captured by conventional optics. The under-

lying principle used to describe the light field is the plenoptic function, a 7-dimensional mapping that assigns radiance values to the light rays at every position in space, in every orientation, at all wavelengths, throughout all of time [14]. [1] shows that this can be formally expressed as $L(x, y, z, \theta, \phi, \lambda, t)$, and measured in $W/m^2/sr/nm/s$. With the addition of practical constraints however, the plenoptic function can be expressed more concisely as a parameterisation of 4 variables.

Pixels on camera sensors integrate the number of photons arriving at them over a finite period of time removing the temporal dimension, and each colour channel can be thought of as a monochromatic sampling of the light field, removing the spectral dimension. Additionally, and importantly, is the constraint that the radiance of light rays propagating through a vacuum do not change if samples are restricted to the convex hull of the scene, thus reducing the overall dimensionality of the plenoptic function by one parameter [18]. To illustrate this, one could think of the light rays leaving the inside of a bowl sat upright on a table. Many of those rays may only travel a small distance before being blocked by the inside of the bowl itself, meaning those rays will never be registered by any practical measurement device. As shown in figure 2.5 if we consider only the convex hull of the bowl however, we are required only to represent the value of the plenoptic function on the encapsulating surface of the object [12].

Also illustrated is a common convention for describing light rays in this 4 dimensional space called the two plane parameterisation. In this parameterisation, two parallel planes are used to fix both the position and orientation of each ray by fixing their points of intersection with two parallel planes. By convention, the plane closest to the scene it termed *u, v* and the plane closest to the camera sensor is the *s, t* plane.

This 4D realisation of the light field originated as a model of rendering 3D computer graphics, one which shifted emphasis from notions of texture and geometric primitives to modeling the behaviour of light rays permeating space. Since then however, the conceptual framework of the light field has drawn a following of researchers at an intersecting region of signal processing, computer vision and robotics [5]. This notion of light field imaging finds a foothold in this thesis project through the
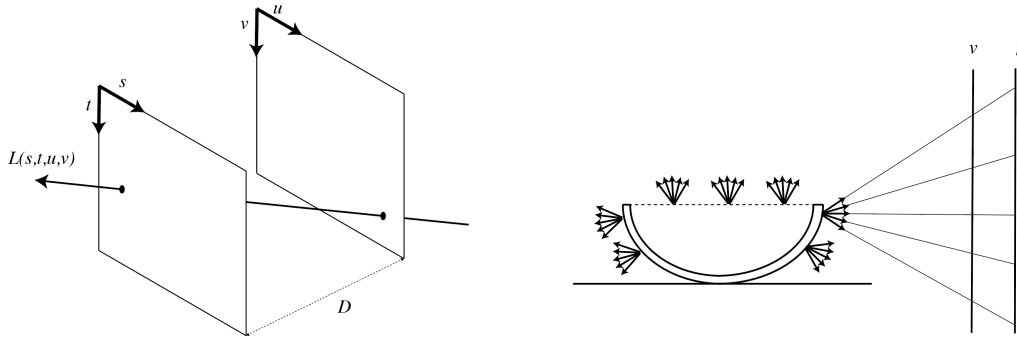
Figure 2.2: Two plane parameterisation (left): the plenoptic function can be described by the radiance along a ray passing through two parallel planes. The free space assumption (right): if we consider only the bundle of rays leaving from the convex hull of the object at a particular instance in time, in a single colour channel, we can parameterise the light rays as a function of 4 variables rather than 7.

utilisation of camera arrays, which are devices that sample multiple views of the same scene. Using a camera array is a simple method for acquiring a sparse sample of the light field, where the position of the camera determines *(s,t)* while the location of the pixel determines *(u, v)* [32]. The images captured from a camera array are mapped easily to the 4D light field, and identifying corresponding pixels across images exposes a rich tapestry of geometric information about the scene.

One way that this geometric information can be easily visualised is by taking slices of the light field image in the s,u or t, v axes as shown in figure 2.4. While the idea of taking a 2D slice from the 4D image can seem complex, the task of generating a so called 'epipolar plane image' from a camera array is deceptively simple. Images captured from camera arrays can be stacked to form a solid volume, from which 2D slices can be sampled. Each of these slices yields an image characterised by sheared straight lines, encoding information about the geometry of the scene, including depth and occlusions [3].

The geometric information encoded in a light field sampling can alternatively be visualised by processing the image into a 'focal stack'. Focal stacks closely resemble images with shallow depth of field such as those that can be captured from a commercial DSLR camera. Light field focal stacks differ from focus in the optical sense however in that they are synthetic and can be recomputed after the image has been taken, effectively allowing control over the depth of field and focal depth in post-processing. Focal stacks can be computed from camera array images by layering images over one another and taking the average value for each pixel. The result is that parts of the scene that closely overlap appear in focus while areas with poor overlap create a 'bokeh' effect. More formally, if the relative pose of each camera is known, a specific focal stack for any desired depth can be computed by projecting each image onto the desired focal plane, and computing their average [29].
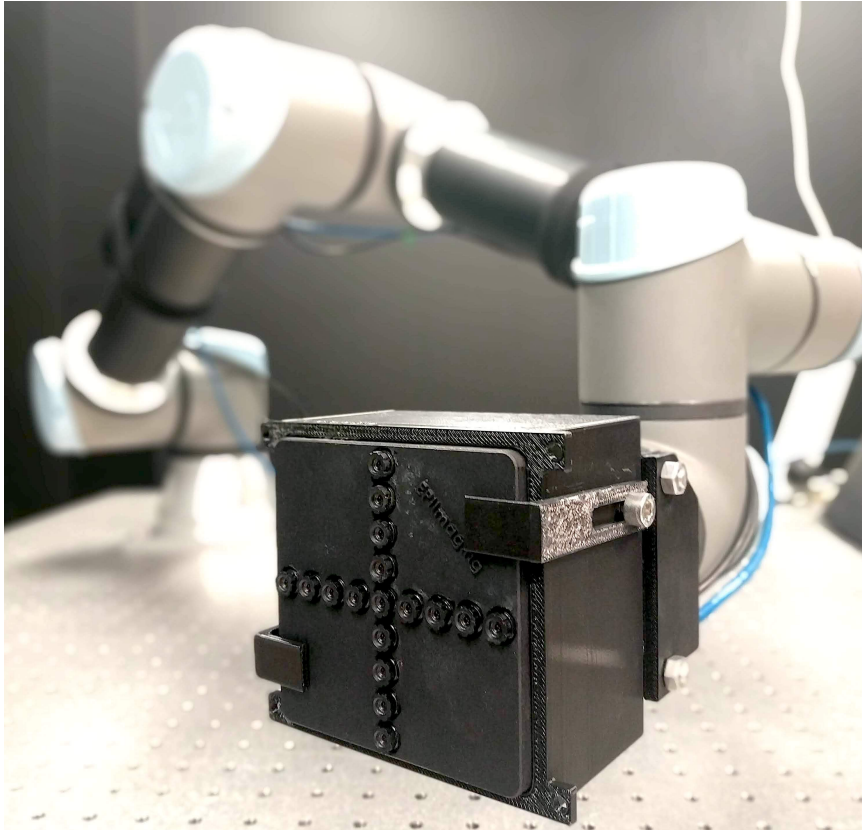
Figure 2.3: An example of a camera array mounted on a robotic arm. This camera array is configured as 17 sub-apertures arranged on a single plane in a cross-hair formation. Camera arrays sample several views of the same scene and are thus capable of acquiring a sparse sample of the light field. This is the camera that will be used throughout this thesis project.

Figure 2.4: Epipolar Plane Images (left): Shown as a slice of a volume, the images formed by dissecting the image in the s,u and t,v planes are characterised by sheared straight lines, with the grade of the slope encoding the amount of parallax experienced by a pixel at that u or v coordinate. Synthetic aperture focusing (right): taking the average of every image from the camera array yields an image where different parts are in focus depending on the alignments of the images.

These representations of the light field will play an important role in this thesis project as we experiment with different methods for feeding light field images to the machine learning pipeline. An important consideration in any machine learning algorithm is the feature space - based on what particular inputs will the algorithm be making its decision? Raw images contain millions of measurements and thus represent an incredibly high-dimensional feature space for neural networks to process. Light field images are several times larger, and thus it is important that some form of dimensionality reduction is used to ease the training process. With the goal of investigating effective methods of feeding light fields to neural networks, this thesis will explore the use of three different light field formats as the entry point to the machine learning pipeline. The first two will be the focal stack, and epipolar plane image described above, interpreting the images as a 3 dimensional volume created by stacking 2 dimensional images on top of one another. The third will treat the light field as a 4 dimensional volume, requiring a 3D signal processing pipeline to fully take advantage of the dimensionality.

2.3 Machine Learning in Computer Vision

An oft-quoted anecdote in computer vision tells of MIT researcher Seymour Papert, who in 1966 assigned a summer project that sounded simple enough, namely to construct a 'visual system' that could describe what objects it saw by name [25]. While the regimes of computer vision have evolved substantially since 1966, many of the ideas, and challenges have persisted. This is embodied in the popularity of projects such as the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [28], drawing researchers from institutions around the world. It was at the ILSVRC annual challenge where in 2012, a convolutional neural network achieved a top-5 error rate of 15.3%, outperforming all previous submissions by 10.8% [16]. Until 'AlexNet' in 2012, the competition subsisted of competitors introducing algorithms that produced marginal improvements year on year. Needless to say, an improvement of over 10% made a big splash in computer vision, drawing attention to the powerful capabilities of deep learning.

While deep neural networks for computer vision have gained massive popularity since the success of AlexNet, the history of neural architecture models begins much earlier, with the perceptron as described by Frank Rosenblatt in [26]. The fundamental building block of neural networks, the 'perceptron' is a module that accepts several inputs, and produces a single output computed as a weighted sum of each input - allowing complex functions to be approximated when several perceptrons are layered together as a 'multilayer perceptron' [22]. The process of finding the optimal set of weights that produce the desired output given a set of inputs is referred to as training, and in practice is usually found by optimising some cost function using the backpropagation algorithm [27].

Convolutional neural networks (CNN's) are similar to multilayer perceptrons, but introduce a
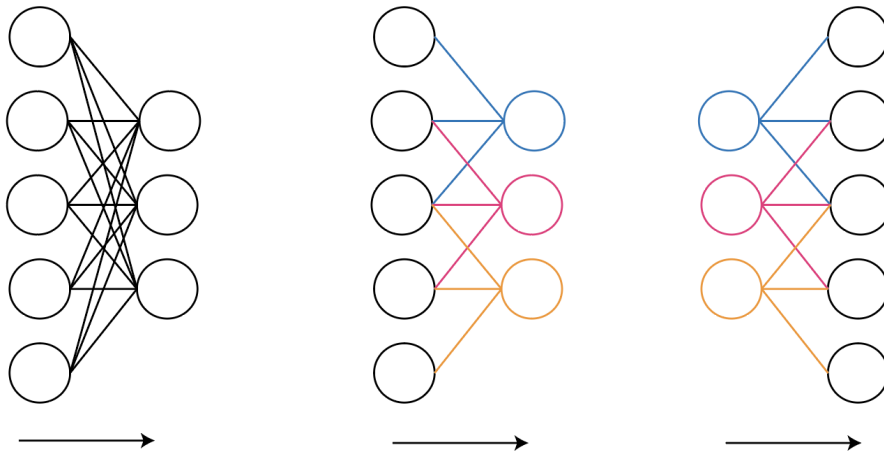
Figure 2.5: Multilayer Perceptron (left): each output from one layer is fed into the inputs of every node in the subsequent layer. Convolutional Neural Networks (middle) on the other hand have a 'receptive field', taking advantage of the spatial coherence of pixels in image data. The convolutional upsampling operation (right) is frequently used to upsample a low-dimensional feature space, to a higher dimensional one. It is often employed as a 'learned' information decompression.

spatial invariance that makes them particularly well suited for extracting high-level features from image data. Where multilayer perceptrons are densely connected, the connectivity pattern of each node in a CNN takes advantage of the hierarchical organisation of patterns in image data by using a 'receptive field'. Biologically inspired, these nodes respond to stimuli only in their receptive field, and so they typically learn to identify salient 'features' in the image - combinations of pixels that represent some kind of underlying structure. As these types of networks grow deeper, the features that they learn typically become more complex [17]. Early layers are provided with a small region of the image, typically a window between 3 and 11 pixels wide, and so will usually learn quite primitive features such as lines and corners. Deeper layers however may learn to recognise higher-level features such as eyes and mouths, and eventually even human and animal faces.

CNN's are thus popular 'feature extractors' in computer vision - their ability to learn to respond to different types of stimuli mean that they have been used as a dimensionality downsampling tool, taking the millions of dimensions present in digital images and compressing them to a feature space with a much smaller number of parameters. Closely related is the convolutional upsampling operation which performs the inverse - taking a feature space and learning to upsample that feature space into something meaningful [20]. This has given rise to a particular topology of CNN called the encoder-decoder architecture, which will be useful throughout this thesis. The encoder part of a CNN is composed of a series of convolutional down sampling operations - this can be thought of as finding an efficient compression of the information stored in the image. The decoder subsequently

uses this compressed form of the information to extract some meaningful information about it. One example where this architecture has been used is in [20], which outputs a classification for each pixel in the image. In this thesis, a fully convolutional encoder-decoder architecture will be used for a similar purpose, but rather than classifying each pixel into one of several categories, it will regress depth values for each pixel.

Chapter 3

Literature Survey

3.1 Depth Estimation and Visual Odometry

3.1.1 Geometric Approaches

Since the invention of the pinhole camera, most cameras have captured 2D representations of a 3D world, meaning information about physical structure is lost. With the addition of a second viewpoint however we can learn a little more about the shape of the scene as governed by epipolar geometry [13]. Meanwhile, a video is a sequence of images, taken in very quick succession, and so one could think of video footage as a multi-view camera where each image is separated not only spatially but also temporally. Identifying the amount of motion between two temporal frames of the video camera is the goal of visual odometry, where one intuitive approach is to observe the direction of movement of each pixel between the pair of images. In practice, features such as SURF [2] or SIFT [21] features are used to simplify the search for image correspondences. The fundamental matrix, which describes the relationship between two cameras in space can then be estimated using techniques such as the 5 or 8 point algorithm.

Doing so monocularly however is problematic because with a single camera there is no way of concretely discerning the actual magnitude of the movement based on pixel data alone, meaning some kind of scale factor needs to be estimated based on characteristics of the image [7, 24, 34, 35]. In fact, this scale ambiguity is often exploited by film makers - what appears as a sweeping shot of a vast landscape on the big screen is often modeled as a miniature film set in the studio. Because the image is monocular, there is no way to ground our measurements of scale in real world units, and so we resort to our imaginations and learned experiences to fill in the gaps. What *is* preserved in these monocular setups however is the overall structure of the scene and motion of the camera - we may not know how large the object is or how far the camera has moved, but we *can* compute the shape of the object as well as the direction of camera motion.

Using a stereo pair of cameras with a known baseline however allows the scale ambiguity issue to be resolved. In the monocular approach, the fundamental matrix must be estimated to an unknown scale factor, however given two camera views of the same scene, features can be directly triangulated in 3D space. Corresponding these 3D points between the first and second pair of images allows the translation and rotation of the camera pair to be found by optimising a cost function to most closely match the two sets of 3D points.

An alternative approach that utilises all of the pixel energy in the scene, takes advantage of the advanced geometric information available when the regular camera is swapped out for a plenoptic camera [4]. Utilsing this rich geometric information, [4] estimates a pixel-wise depth estimate using the gradient-depth constraint. The 'plenoptic flow' algorithm is an attractive approach for visual odometry as the computational complexity is constant with scene complexity, and the camera pose has a closed-form solution.

3.1.2   The Machine Learning Approach

One recent approach that has driven a large body of research is the use machine learning to perform both of these tasks, utilising convolutional neural networks to learn a non-linear mapping directly from a pair of images to their depth maps, as well as the relative pose between them [6, 8, 10, 19, 36]. Combining the spatial awareness of the convolutional down sampling operation with a neural networks ability to learn accurate approximations for complex, non linear functions, these approaches have found success in both supervised [19, 6], and unsupervised settings [8, 10, 36]. In the supervised family of algorithms, [6] and [19] take advantage of datasets such as KITTI [9], containing ground truth depth maps collected using state-of-the-art depth sensors and poses measured from inertial sensors.

Unsupervised experiments such as [8, 10, 36] on the other hand exploit the constraints imposed by epipolar geometry to learn depth either by using known camera poses or by estimating pose in addition to depth. The general pipeline utilises two CNN's, one who's purpose is to predict dense per-pixel depth maps given a single image, and the other who's job is to estimate the 6 degree-of-freedom pose from the first frame to the second given both images.

While the training pipeline is unsupervised in the sense that labelled data is not required, some form of supervision signal is nevertheless required to optimise the parameters of the pair of networks. These papers take advantage of the fact that if the physical structure of a scene is known, then a novel rendering of that scene from a different viewpoint is achievable. Thus, [8] suggested using photometric reconstruction as a supervision signal - from I_{t2}, reconstruct I_{t1}. The difference between the reconstruction and the real image can form a supervisory signal for the pair of networks.

$$L_{photometric} = \sum_{n}^{W} \sum_{m}^{H} |I_{t1}(m,n) - \hat{I_{t1}}(m,n)| \tag{3.1}$$
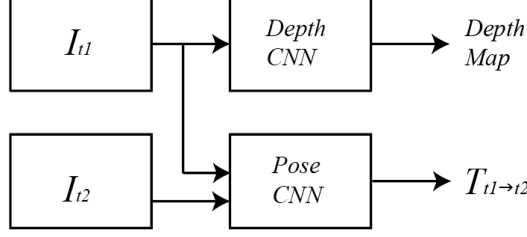
Figure 3.1: Machine learning approaches have demonstrated strong results in simultaneously estimating depth maps and relative poses between images. Typically, a pair of CNN's is used, one for each task. The depth CNN is provided with the first view at $t = 1$ and the pose CNN is provided both views.

Humans can easily imagine what a scene will look like if we move our heads slightly, but only if we know the overall shape of the object we are looking at. Similarly, given two images separated by a small interval of time I_{t1} and I_{t2}, one could synthesise what I_{t1} would look like by sampling the pixels from I_{t2}, if the relative pose $T_{t1 \rightarrow t2}$ between the two cameras and a pixel-wise depth of the scene $D_{t1}(p)$ is known. In practice, this can be done by obtaining the projected coordinates of p_{t1} on I_{t2}'s camera sensor. Assuming a pinhole model, the complete expression for p_{t1}'s projected location on the second camera, $\hat{p_{t2}}$ is

$$\hat{p_{t2}} = KT_{t1 \rightarrow t2}D(p_{t1})K^{-1}p_{t1} \tag{3.2}$$

where K represents the intrinsics matrix. The obtained coordinate $\hat{p_{t2}}$ is continuous, while pixel coordinates are discrete. A pixel value thus needs to be interpolated, keeping in mind that each step of a neural network pipeline must be differentiable to support the backpropagation of gradients.

It was suggested in [36] that adopting bilinear sampling could be used as a fully differentiable sampling mechanism. The use of bilinear interpolation as a differentiable sampling pipeline was first proposed in *Spatial Transformer Networks* [15], and adapted by [36] to perform the differentiable image warp. A bilinear sampling kernel is described by

$$V = \sum_{n}^{W} \sum_{m}^{H} U_{nm}max(0, 1 - |x - n|)max(0, 1 - |y - m|) \tag{3.3}$$

Where V is the output of the sampling kernel, U is the source image being sampled, n and m index over the columns and rows of the kernel respectively, H and W are the height and width of the sampling kernel, and x and y are the local coordinates of the sampling location. The 2×2 sampling kernel used to perform the photometric reconstruction is, in essence a weighted sum of the 4 nearest
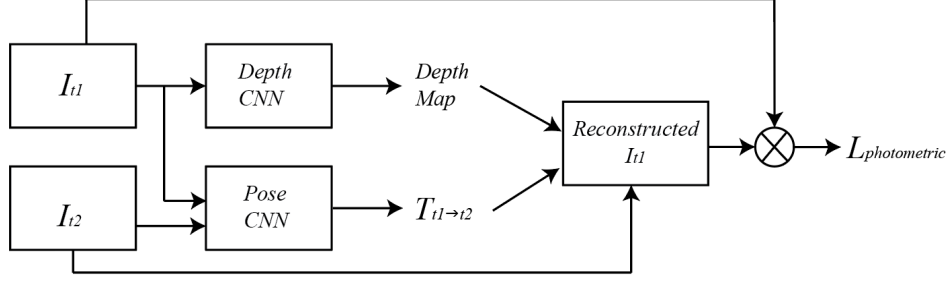
Figure 3.2: To provide a supervisory signal to the depth and pose CNN's, [8] suggested using photometric reconstruction. The loss function is formulated by taking the difference between the reconstructed image $\hat{I}_{t1}$ and the actual image I_{t1}, shown in equation 3.1.

neighbour pixels, based on its proximity to those pixels. The equation is differentiable with respect to both x and y:

$$\frac{\partial V}{\partial x} = \sum_{n}^{W} \sum_{m}^{H} U_{nm} max(0, 1 - |y - m|) \tag{3.4}$$

$$\frac{\partial V}{\partial y} = \sum_{n}^{W} \sum_{m}^{H} U_{nm} max(0, 1 - |x - n|) \tag{3.5}$$

Using this bilinear sampling kernel, we can thus sample pixels from I_{t2} to reconstruct I_{t1} in a fully differentiable manner, allowing the backpropagation of gradients through the networks. The pipeline with photometric reconstruction as the supervision signal can be illustrated as shown in figure 3.2.

In addition to the photometric reconstruction loss providing the main supervision signal to the network, [36] employs a smoothness loss to ensure that the produced depth map is globally smooth. The smoothness penalises the second order gradient of the image - i.e. the depth network is encouraged to produce a depth map characterised mostly by low-frequency components, and penalised for high-frequency components. Recognising that depth discontinuities frequently occur in parts of the image where a strong edge appears, [10] on the other hand suggested the use of an edge-aware smoothness loss that also penalises large gradients in the depth map ∂d, but lowers the weight of the loss in regions where the image gradient ∂I is large.

$$L_{smooth} = \sum_{n}^{W} \sum_{m}^{H} |\partial_x d_{nm} e^{-|\delta_x I_{nm}|}| + \partial_y d_{nm} e^{-|\delta_y I_{nm}|}| \tag{3.6}$$

There have been several iterations [10, 11, 33] of this pipeline, most of which have focussed on introducing novel loss functions to improve the quality of the depth and pose estimates. To the best

of our knowledge however, this pipeline has only ever been applied to monocular and stereo camera setups. While monocular and stereo cameras are ubiquitous in modern digital devices, this thesis will investigate the potential for the use of this pipeline to achieve improved results using a broader, more generalised family of imaging devices.

It is estimated that the biological eye has independently evolved no fewer than 40 times in the animal kingdom producing a huge variety of visual configurations [23]. The evolution of a systems eye is evidently correlated to the tasks that the system has to solve. Breaking free of the pinhole principle from which human eyes have developed and which commercial cameras have adopted has wide implications in the field of machine vision. However, interpreting the ray directions and geometry for new cameras may not always be apparent to the users, giving rise to the data-driven approach. Working on the principle that machine vision does not necessitate mimicking the human eye to capture imagery, this thesis investigates the capabilities of machine learning for querying depth and visual odometry using one such novel imaging device.

3.2 Convolutional Neural Networks and Light Field Images

In order to craft such a machine learning pipeline that complements the use of camera arrays, an investigation into input methods for 4D light fields into convolutional neural networks is due. It is recognised in [31] that while epipolar plane images (EPI) explicitly encode depth information, directly extracting such information without significant post-processing refinement and computational cost is difficult. [31] demonstrates strong results in using a CNN to interpret the gradient-depth relationship between an EPI and the images corresponding depth map. An EPI is extracted in both the horizontal and vertical direction for every row and column of the image in the u, v plane, and subsequently fed to the network as two long, cubic volumes to obtain their results.

The rich textural data available in a 4D lightfield was purposed by [30] for the task of material recognition using a deep CNN. In addition to reporting significantly improved material-recognition results (from 70% to 77% accuracy), [30] proposes and compares a number of strategies for training on 4D images. One method that achieves strong results uses an angular filter, taking advantage of the 'angular resolution' that is gained by using a light field image over 2D images. The 4D light field image is first reshaped to form what [30] calls a 'remap' image. A traditional 2D image is formed by discrete pixels; a remapped 4D image on the other hand is formed of blocks of pixels, each block of size $h_a \times w_a$, formed by taking one pixel from each camera in the array. The result is a 2D image from which a 2D convolutional filter is able to learn features that indicate texture and parallax. While the remapping method achieved the best results, unfortunately such an arrangement of the image data does not make sense in the case of the camera array being used for this project, as the cameras are arranged in a crosshair formation - 8 vertical cameras, 8 horizontal and 1 center image. The remapped image using this camera would comprise 17×17 blocks of mostly black pixels,

producing a very inefficient representation of the data.

Another method proposed in [30], albeit one that demonstrated lower accuracy than the angular filter method, was to concatenate images along their RGB channels prior to being fed to the network. The dimensionality of the image is quickly downsampled, and is thus a more computationally and memory efficient method than using angular filters, whilst still demonstrating improvement over 2D images. A potential improvement on this strategy that may prove useful in this work would be to use a dilated convolution, effectively opening the receptive field to a much larger area of the image without increasing computational complexity. In another potential improvement, convolutional filters that sample in 3 dimensional space may be able to utilise the 4 dimensional signal structures in the light field more effectively than a 2 dimensional filter.

Chapter 4

Progress Report

Chapter 5

Updated Research Proposal

Bibliography

[1] Edward H. Adelson and James R. Bergen. The plenoptic function and the elements of early vision. In *Computational Models of Visual Processing*, pages 3–20. MIT Press, 1991.

[2] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008.

[3] Robert C. Bolles and H. Harlyn Baker. Readings in computer vision: Issues, problems, principles, and paradigms. chapter Epipolar-plane Image Analysis: A Technique for Analyzing Motion Sequences, pages 26–36. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

[4] D. G. Dansereau, I. Mahon, O. Pizarro, and S. B. Williams. Plenoptic flow: Closed-form visual odometry for light field cameras. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4455–4462, Sep. 2011.

[5] Donald Dansereau. *Plenoptic Signal Processing for Robust Vision in Field Robotics*. PhD thesis, Australian Centre for Field Robotics, 2014.

[6] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *CoRR*, abs/1406.2283, 2014.

[7] Paul Verlaine Gakne and Kyle O'Keefe. Tackling the scale factor issue in a monocular visual odometry using a 3D city model. 2018.

[8] Ravi Garg, Vijay Kumar B. G, and Ian D. Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. *CoRR*, abs/1603.04992, 2016.

[9] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research (IJRR)*, 2013.

[10] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency. *CoRR*, abs/1609.03677, 2016.

[11] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. *CoRR*, abs/1806.01260, 2018.

[12] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 43–54, New York, NY, USA, 1996. ACM.

[13] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, 2004.

[14] Ivo Ihrke, John Restrepo, and Loïs Mignard-Debise. Principles of light field imaging: Briefly revisiting 25 years of research. *IEEE Signal Processing Magazine*, 33:59–69, 09 2016.

[15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and koray kavukcuoglu. Spatial transformer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2017–2025. Curran Associates, Inc., 2015.

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[17] Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[18] Mark Levoy and Pat Hanrahan. Light field rendering. In *Proc. ACM SIGGRAPH*, 1995.

[19] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian D. Reid. Learning depth from single monocular images using deep convolutional neural fields. *CoRR*, abs/1502.07411, 2015.

[20] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014.

[21] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, November 2004.

[22] Marvin Minsky and Seymour Papert. *Perceptrons: An Introduction to Computational Geometry.* MIT Press, Cambridge, MA, USA, 1969.

[23] Jan Neumann, Cornelia Fermüller, and Yiannis Aloimonos. Polydioptric cameras: New eyes for structure from motion. In Luc Van Gool, editor, *Pattern Recognition*, pages 618–625, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[24] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages I–652–I–659 Vol.1, Los Alamitos, CA, USA, jul 2004. IEEE Computer Society.

[25] Seymour Papert. The summer vision project. 10 1966.

[26] Frank Rosenblatt. *Principles of Neurodynamics*. Spartan Books, 1959.

[27] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[29] Vaibhav Vaish, B. Wilburn, N. Joshi, and M. Levoy. Using plane + parallax for calibrating dense camera arrays. volume 1, pages I–2, 01 2004.

[30] Ting-Chun Wang, Jun-Yan Zhu, Hiroaki Ebi, Manmohan Chandraker, Alexei A. Efros, and Ravi Ramamoorthi. A 4d light-field dataset and CNN architectures for material recognition. *CoRR*, abs/1608.06985, 2016.

[31] Xing Sun, Z. Xu, Nan Meng, E. Y. Lam, and H. K. . So. Data-driven light field depth estimation using deep convolutional neural networks. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 367–374, July 2016.

[32] Li Yao, Yunjian Liu, and Weixin Xu. Real-time virtual view synthesis using light field. *EURASIP Journal on Image and Video Processing*, 2016(1):25, Sep 2016.

[33] Huangying Zhan, Ravi Garg, Chamara Saroj Weerasekera, Kejie Li, Harsh Agarwal, and Ian D. Reid. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. *CoRR*, abs/1803.03893, 2018.

[34] Dingfu Zhou, Y. Dai, and Hongdong Li. Reliable scale estimation and correction for monocular visual odometry. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 490–495, June 2016.

[35] Dingfu Zhou, Yuchao Dai, and Hongdong Li. Ground plane based absolute scale estimation for monocular visual odometry. *CoRR*, abs/1903.00912, 2019.

[36] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017.